

Building Transparent and Efficient Community Administration: Agile Development of a Neighborhood Information System at Kertamukti Sakti Residence

¹**Reza Riyaldi Irawan***, ²**Dendy K Pramudito**, ³**Muhtajuddin Danny**

^{1,2,3}Program Studi Teknik Informatika, Fakultas Teknik, Universitas Pelita Bangsa, Cikarang, Jawa Barat, 17540, Indonesia

*e-mail: papers.upb@gmail.com

(*received*: 8 October 2025, *revised*: 27 October 2025, *accepted*: 28 October 2025)

Abstract

Community management in residential areas often relies on manual paper-based administration, leading to inefficiency, unclear financial records, data loss, and limited transparency, which undermine good governance and residents' trust. This study aims to develop a web-based neighborhood (RT/RW) management information system to improve administrative effectiveness, financial transparency, and service quality. The system was built using CodeIgniter, PHP, MySQL, Bootstrap, and jQuery, applying the Agile development method to ensure flexibility and iterative improvement through continuous feedback between the developers and the community. The development process consisted of planning, design, coding, testing, and release stages, with flowcharts and wireframes supporting interface design and black box testing used for functional validation. The system was evaluated using a user-centered usability assessment (System Usability Scale – SUS), obtaining an average score of 82.5, which falls under the Excellent category. In addition, the financial reporting process time was reduced from three days to one hour, and data entry errors decreased by 90%, proving that the system significantly improves operational efficiency and transparency compared to manual methods. In conclusion, the combination of Agile methodology and lightweight frameworks such as CodeIgniter successfully delivers a responsive, transparent, and user-oriented information system that enhances trust and collaboration within the community. Future development will focus on integrating QRIS, e-wallets, and bank transfers to further streamline financial transactions and support sustainable digital transformation in community management.

Keywords: agile development, codeigniter, community administration, management information system, usability evaluation

1 Introduction

In this era of rapid digitalization, the need for efficiency and transparency in community administration continues to grow, including within residential neighborhoods. Many neighborhood community (RT) caretakers still rely on manual methods to record resident data, collect dues, and disseminate information. This conventional approach often leads to delays, recording errors, and a lack of transparency in administrative processes. One such example is the Kertamukti Sakti Residence neighborhood, which faces these challenges [1].

The main obstacle lies in the limited time and resources of RT administrators, who often have full-time jobs outside their community duties. As a result, administrative services for residents are frequently delayed and disorganized [2]. Therefore, a digital system is needed to assist RT administrators in managing resident data and financial transactions more efficiently and systematically.

A web-based system is considered the most suitable solution because it allows real-time access, centralized data management, and easy communication between administrators and residents regardless of time or location. Compared to mobile or desktop-based applications, web-based platforms offer greater accessibility and lower maintenance costs, which are crucial for small-scale community environments such as neighborhood associations. Additionally, they allow seamless data sharing, transparency in financial reports, and faster dissemination of information.

This study develops a web-based RT Management Information System using the CodeIgniter 4 framework, supported by MySQL, PHP, Bootstrap, and jQuery. CodeIgniter was selected because of its lightweight, modular structure and clear MVC (Model-View-Controller) architecture, enabling rapid, structured, and secure development. To ensure flexibility and user-oriented improvement, the Agile development approach was adopted, allowing continuous refinement based on user feedback. The resulting system aims to accelerate administrative processes, improve communication among residents, and promote transparency in financial management. This research provides practical implications for assisting RT administrators in performing their daily duties more efficiently and may serve as a reference for other neighborhood communities [4].

This research aims to achieve three main objectives. First, to design and develop a web-based neighborhood community management information system that integrates resident data management, contribution tracking, and information dissemination into a single unified platform. Second, to implement the Agile development methodology throughout the system development process to ensure flexibility and continuous adaptation based on user feedback and evolving community needs. Third, to evaluate the effectiveness and usability of the developed system in improving administrative efficiency, financial transparency, and overall service quality for both administrators and residents within the neighborhood community environment.

2 Literatur Review

Discusses the Financial Information System in discussing the improvement of effectiveness and efficiency in recording and reporting finance which were previously done manually using auxiliary books and bookkeeping[5]. The data collection method in this study uses observation, interviews, and literature studies, while the system development model used is the waterfall model. The results of this study are a financial information system that makes it easier for RT administrators to record transactions and manage cash flow and assets, and create financial reports so that the financial management process becomes more orderly, accurate, and integrated.

The system design of Web-Based Population Information System in the RT/RW Environment discusses the development of a system to simplify the process of data collection and management of population data at the RT/RW level. The main problem faced is that data management is still manual using physical books and archives, so that errors and data loss often occur when changing administrators. This study uses the Rational Unified Process (RUP) method with Unified Modeling Language (UML) modeling, and is implemented using PHP and the Laravel framework. The results of the study are a web-based population information system that can be accessed by admins, RTs, RWs, and residents to manage KK (Family Card) data, KTP (National Identity Card) data, and submit cover letters. This system increases efficiency and accuracy in population administration [6].

The Implementation of Agile Methods in the Design of a Village Tourism Information System discusses the development of a web-based tourism information system to facilitate the community and tourists in accessing tourism information in East Biluhu Village. The method used is Agile with Unified Modeling Language (UML) modeling and testing using Black Box Testing. The system was designed using PHP and MySQL. The results of this study are a tourism information system that helps administrators manage tourism data and makes it easier for tourists to obtain information quickly and efficiently [7].

The development of a web-based neighborhood community management information system discusses the creation of a system to facilitate administrators in managing resident data and recording neighborhood community activities such as meetings and mutual cooperation. Waterfall is one of the model that have been used, consisting of: need analysis, system's design (using UML), coding (by using java language and MySQL as a database) and testing through User Acceptance Test (UAT). The result of this research is that the management information system for a web-based neighborhood community which can combine resident's data with the activities in the scope of neighborhood community will improve efficiency, transparency and easy to receive information by resident and community caretakers. The trials gave result 89% level of satisfaction which suggests that the system is being well used [8].

The user interface of the proposed system to minimize social problems discusses the development of an information system aimed at assisting neighborhood community caretakers in

managing the resident data and resolving social problems in the community. The raised issue is the lack of an effective data management and citizen communication system, where the information and services are often hampered. This study uses the Scrum method as part of an agile approach and designed using PHP, JavaScript, and MySQL, with a Figma-based interface design. The results of this study are a web-based community management information system that has features such as area maps, discussion and complaint forums, activity calendars, population data, security information, culture, and social assistance which is expected to increase transparency, accelerate services, and assist the government in monitoring community welfare from the RT level [9].

Contradictory to prior studies, this study presents a distinct perspective which has developed an Integrated Information System for the Neighborhood Association, accessible via the website of the Kertamukti Sakti Residence Housing Complex. This system focuses on managing resident administration and membership fees and providing information in a digital and integrated format. Given that time is limited for RT caretakers, this system is designed to enhance data management, improve service delivery to residents, and increase overall efficiency. This study combines all three elements into a unified model, unlike earlier research that primarily focused on financial aspects as the sole funding source or on population and resident communication. The author employed Agile Development in creating the system due to its adaptability and capability to address new client needs. The system was constructed using the CodeIgniter 4 framework, which is compatible with MySQL, PHP, Bootstrap, and jQuery [4][10].

3 Research Method

This study method includes data collection, analysis, system planning, system development and testing used in the implementation of a web-based RT Management Information System using the CodeIgniter framework in RT 002/RW 010, Kertamukti Sakti Residence as shown in Figure 1. Data were collected through direct observation to understand the flow of resident data collection and contribution management mechanisms, interviews with the head of neighborhood community, treasurer, and secretary to explore system needs, and literature studies from various theoretical and research sources related to RT management.

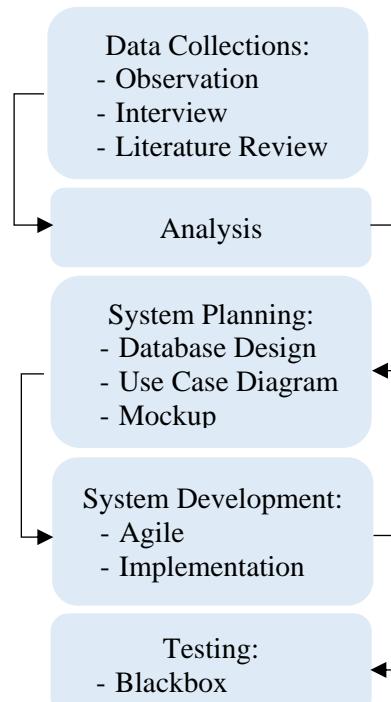


Figure 1 Research method stages

There are seven stages in the agile development method where every members must have some plans to achieve the goals and the needs of the system development, design to create a system interface and architecture design, develop to carry out the development or coding process, test to ensure the system runs according to its function, deploy to apply the system to the user environment, review to evaluate the results and receive feedback, and launch as the system launch stage so that it can be fully used by users[14][15] while the system testing is implementing the black box testing method which testing the function and the output of the system without seeing the program code directly [10] [17].

3.1 Data Collection

Data collection as an initial step for system development is very important in a research because it provides a foundation for system development [11]. This step is considered as type of method to obtain the data should be done cautiously and thoroughly, thus, the results obtained are meaningful and match with the system plan for the based neighborhood community management information system . Three approaches were used to collect and retrieve data during this study.

- a) Observation: During the observation process, it was discovered that the neighborhood community still carries out many manual processes, from calculating monthly dues to registering residents to paying security personnel and other expenses. This process is undoubtedly very cumbersome for neighborhood community caretakers.
- b) Interview: The main aim of these sessions was to find out the perspectives and direct experiences of the parties involved in the neighborhood management system as well as ways to resolve these problems. Those sessions were conducted in two stages, where the first stage was carried out with the head of the neighborhood community and the second one was carried out with the resident as the party who is part of the neighborhood community member.
- c) Literature Review: Collecting data through literature review and sales information system references in books, scientific journals, articles, libraries, or online databases. This is done to identify the basic concepts needed to build a sales information system, such as sales data models, sales business processes, and the technology used. The theory discovered was then used to create a framework or structure for the neighborhood community management information system. By conducting in-depth literature research, the author gained the understanding and knowledge needed to create an effective and innovative sales information system

3.2 Analysis Stage

This activity includes determining the supporting software and system architecture, which is illustrated through several models such as use case diagrams, class diagrams, and activity diagrams. The software used in the development process includes Visual Studio Code as an editor for writing program code and XAMPP as a local web server that functions to run applications and manage the system database.

3.3 System Planing

The aim of this system design is easiness of system development and serve as functional specification documentation.

- a) Database design: Database systems manage, store and retrieve large amount of data efficiently and securely [18]. It is a centralized and organized approach designed to eliminate the problems that plagued older file-based systems of data management, such as data redundancy, inconsistency, poor security, and difficulty in accessing information [19]. And [20] describes that a central part of a database system is the Database Management System (DBMS) which is used by both people and application programs to access databases. A database also enables multiple users and applications to access and manipulate data simultaneously without conflict, supports faster data access for ad-hoc queries, and offers built-in backup and recovery facilities to prevent data loss as shown in Figure 2. Ultimately, by turning raw data into accessible, quality information, it empowers better decision-making and increases overall productivity for businesses and end-users.

b) Use case diagram: The system development process, known as the use case diagram, documents and implements system specifications using a graphical language. Use case diagram is a writing standard or template that addresses business classes and processes in a specific language. Currently, use case diagrams are a standard language that is often used to design a system so that it is easy for ordinary people to understand. Use case diagram functions to make it easier for system designers to prepare the system to be created.

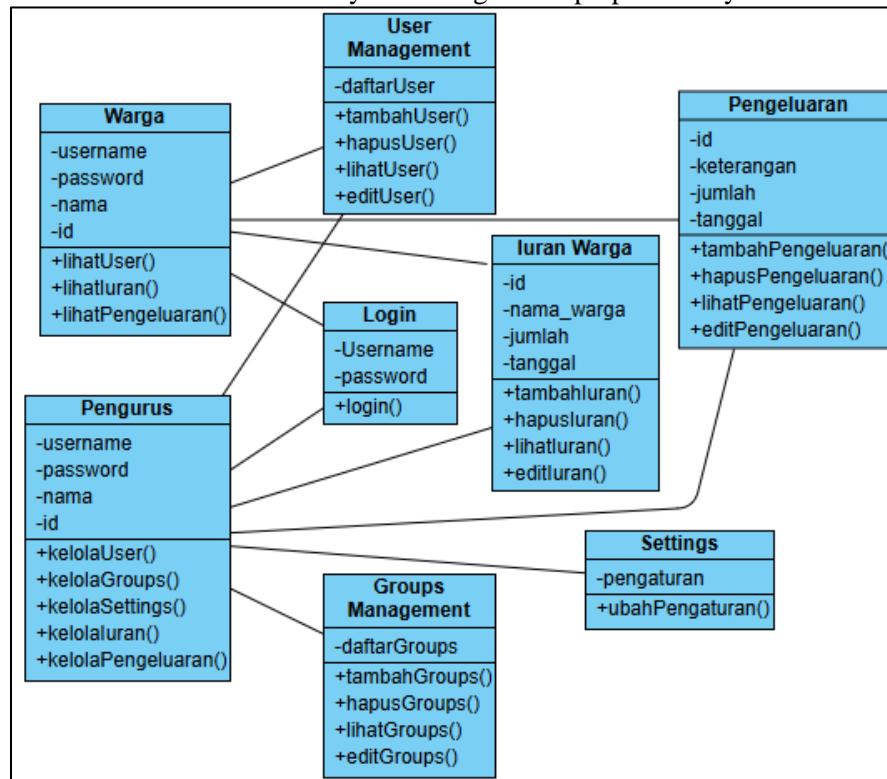


Figure 2 Database design

c) The use case diagram design as shown on Fig 3 is the neighborhood community management information system, where the administrators are the head of neighborhood community and the accountant who will manage the website content that contains of five main features of the neighborhood community management information system.

- Administrator login menu is a web page where the accountant will focus on managing contributions and expenditures, while the neighborhood community will manage resident data. Residents can log in to the website, but they can only view its contents.
- User management is controlled by the administrator where the features that can be viewed are user management (resident data), resident contributions, and expenditures. An explanation of each use case's contents follows. In addition, the head of neighborhood community can add residents, modify resident information, manage resident accounts and view detailed data for a given resident.
- Setting management is menu for administrator to ensure that data is accurate and maintains systematically. Neighbourhood community head is the user manager. The head of neighborhood community shall be able to add resident data, update resident details, manage residents' accounts and get detail information of the resident. All this processes of configuration are directly made by the head of neighborhood community to guarantee the correctness and coherence of data attribution in the system.

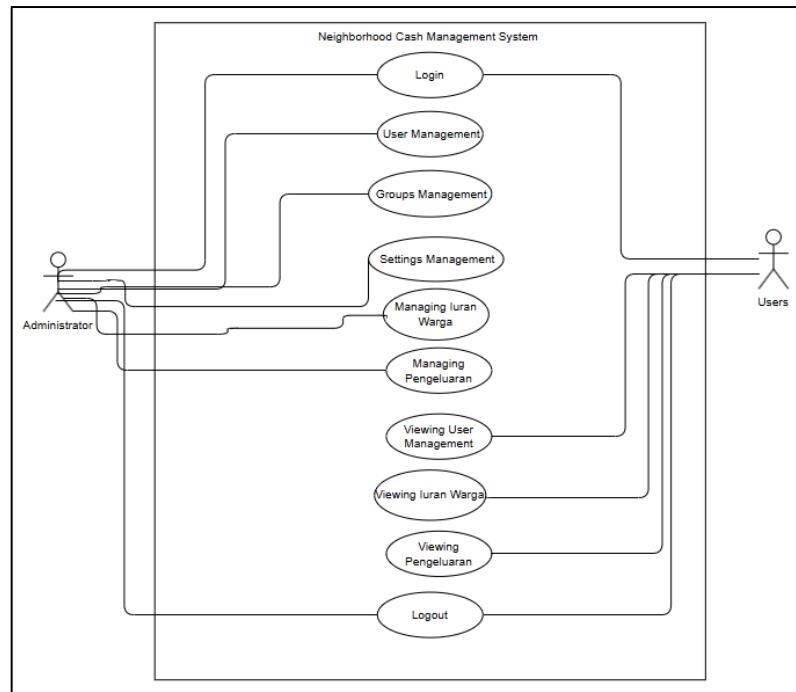


Figure 3 Use case diagram

- iv. Monthly fee menu is an important one that configured by the accountant to handle fee that gathered from the residents. In this menu, accountant can create new contributions; select the type of contribution they wish to look at, see a detailed view for that contact and remove the contribution.
- d) Mockup is an illustration of how a design concept will look when applied to a real object. The mockup is part of a preview of a "flat" design concept that is given visual effects so that the results look similar [10].
 - i. Dashboard mockup shows resident info such as family identity card, national identity number, username, full name, occupation, gender, marital status, address number, mobile phone numbers, car plates number, name of family members, and registration date as shown in Figure 4. Administrators can also edit the information in case of change or update. This function simplifies the management of the resident data and verification making information more accurate and up to date as necessary.

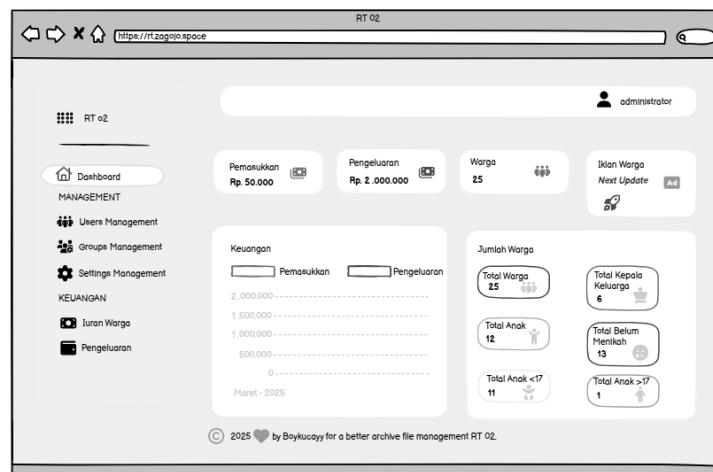


Figure 4 Dashboard mockup

ii. Figure 5 is a user management mockup to manage various groups within the system, containing a list of the names of previously created groups, includes features for adding new groups, editing existing, and viewing detailed information about each group. This feature makes group management more structured and flexible, making it easier for users to organize and group members according to their roles or community needs.

No	Username	No Rumah	Name Lengkap	No HP	Jabatan	Joined
1	G2/16	G2 No 16	Wohdin	08960840*****	Warga	22.32, 05 Mar 2025
2	G3/18	G3 No 18	Rohad Jeneiri	-	Sekretris	20.44, 05 Mar 2025
3	G3/24	G3 No 24	Yadi Mulyadi	08622117*****	Warga	18.60, 06 Mar 2025
4	Teguh	G5 No 5	Teguh Nugroho	08661141*****	Superadmin	00.62, 01 Mar 2025
5	G5/19	G5 No 19	Sri Condi Cahyadi	08578162*****	Ketua RT	10.26, 20 May 2024

Figure 5 User management mockup

iii. Setting management mockup resident data such as, housing number, national identification card number, user ID birth date, occupation, gender, marital status, car plate numbers, family members, mobile phone numbers, and the date of registration. The design has a feature that lets administrator see residents' details and make edits, should there be any changes on it as shown on Figure 6. This functionality consolidates resident data in an efficient and easy-to-use system, allowing for accurate information that meets the needs of neighborhood community.

Figure 6 Setting management mockup

iv. Monthly fee mockup is also created to make it easier for admins of neighborhood communities to take charge, by recording all possible expenses. Figure 7 contains sections to add new expenses, view expense details, edit the data and delete expenses if needed. This mockup also offers a list of entered expenses that allow for a quick and effective overview of or cash flow management.

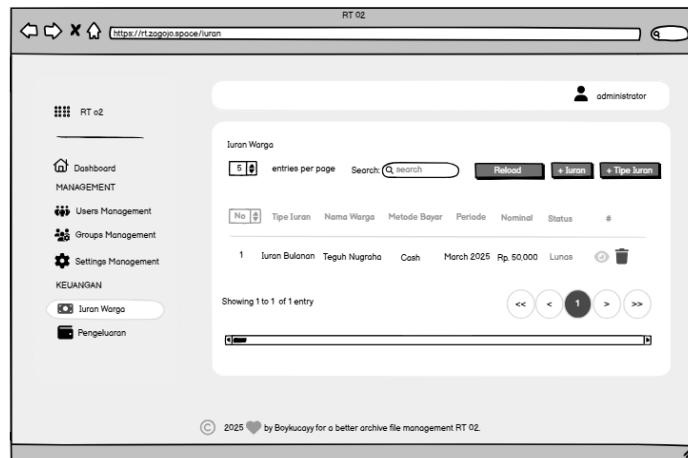


Figure 7 Monthly fee mockup

3.4 System Development

Agile development method is a software development method based on the principles of rapid adaptation to existing changes and intensive interaction with clients during the system design process. This method was utilized as system development because of its flexibility and adaptability to changing user needs. [11][12][13] describe the collaboration between developers and neighborhood community through adaptive communication and rapid feedback which can deliver more effectively, efficiently, and transparently to support administrative and financial management in the neighborhood community environment with stages of the agile development method as shown in Figure 8.

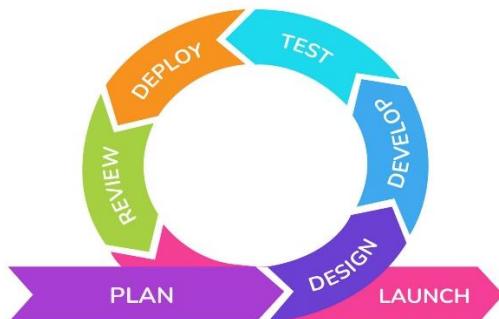


Figure 8 Agile development stages

The following explanation of agile development stages are:

1. Planning is an important stage in the Agile methodology which allows developers to understand the goals of the project, determine requirements, and plan the actions to be taken to achieve the desired results. During the planning stage, developers work closely with stakeholders to determine which features are most important to develop and identify what is most important.
2. The design stage is when creating a detailed product design. Although this stage can be carried out simultaneously with development, this design stage includes visual aspects, the user interface, and the overall structure of the product.
3. Development is a stage where developers begin to execute plans in writing (on a dry-erase board or whiteboard) is when it all becomes development in Agile. Hard (and soft) coded into the product are features that were deemed most important from a planning standpoint. Developers can make estimates each new product replica's development, and these product replicas can be reviewed by stakeholders since development is done in relatively short bursts called sprints.
4. Testing As for Validation, to guarantee that the implemented features work properly and do not harbor any bugs, the testing phase is an important step in Agile methodology. Testing is

accomplished during post development. Testing contains Functional, Integration and Performance tests etc.

5. Deployment of the product that has been developed is implemented or delivered to the production environment or user at the delivery stage. Delivery is done repeatedly in the Agile methodology. From the start, this method allows developers to collect user feedback and make changes if necessary.
6. Review stage or commonly called retrospective demo is a crucial step in Agile methodology because it ensures that the product develops according to the wishes and expectations of the users.
7. Launch stage happened when the product is finally ready to be released to end users after going through various stages of development, testing, and improvement, the launch stage is considered to have reached sufficient quality for use by consumers.

3.5 Testing Stage

System testing involves conducting tests on a complete integrated software system with the purpose of ensuring that it meets specified requirements (functional and non-functional). This is the round of testing that's all about seeing how it works between all the parts, from hardware to software, in a world like a real production world. Testing activities are also needed to ensure that the system being developed can function as desired by the user, where testers may find functional bugs that are not found during other testing processes such as unit testing or integration testing, so that this can reduce operational risks and can foster trust and confidence in the results [9]. This study used a black-box testing approach that offers an objective, user-centric assessment of software operation from an external viewpoint, concentrating on system inputs and outputs without necessitating familiarity with the source code. This method makes testing easier for big systems, can be done early in development, and is good at finding defects, usability problems, and security holes, which makes it easier to connect with other systems [10]. Table 1 is showing 8 of 23 testing scenarios in this study and the result of those scenarios are valid which mean the system function is working properly based on the planning, design, include the development process.

Table 1 List of testing scenarios

No.	Testing Scenarios	Expected Result	Final Result
1	Wrong username credential	System will reject and shows pop-up message “Please ensure your username or password is correct”.	Users cannot login with wrong username
2	Wrong password credential	System will reject and shows pop-up message “Please ensure your username or password is correct”.	Users cannot login with wrong password.
3	Correct username and password	System will authorize and user will be redirected to dashboard menu.	User can login with correct username and password.
4	Click the Dashboard Menu	Quick shortcut to show the dashboard menu	System successfully show the dashboard menu
5	Click the User Management Menu	Quick shortcut to show the user management menu	System successfully show the user management menu.
6	Click the Group Management Menu	Quick shortcut to show the group management menu	System successfully show the group management menu.
7	Click the Setting Management Menu	Quick shortcut to show the setting management menu	System successfully show the setting management menu.
8	Click the Monthly Fee Menu	Quick shortcut to show the monthly fee menu	System successfully show the monthly management menu.

3.6 Usability Evaluation

To assess the system's effectiveness and user acceptance, this study employed the System Usability Scale (SUS) method as a standardized usability evaluation tool. SUS is a reliable and widely-used questionnaire consisting of 10 statements designed to measure users' subjective assessment of system usability, including ease of use, learnability, and overall user satisfaction [22].

The usability evaluation was conducted after the system deployment and involved 20 respondents comprising RT administrators (n=3), treasurers (n=2), and active residents (n=15) who had used the system for at least two weeks. Each respondent was asked to rate 10 statements using a 5-point Likert scale ranging from 1 (Strongly Disagree) to 5 (Strongly Agree). The statements covered various aspects of usability, such as willingness to use the system frequently, system complexity, ease of use, need for technical support, integration of system functions, consistency, learnability, system confidence, required prior knowledge and overall user experience

The SUS score was calculated using the standard formula where individual responses are converted to a score ranging from 0 to 100. According to the SUS interpretation scale, scores below 68 are considered below average, scores between 68-80.3 are above average, and scores above 80.3 are categorized as "Excellent" [22]. This evaluation method was chosen because it provides a quick, reliable, and cost-effective way to measure user satisfaction and system usability from the end-user perspective.

4 Result and Discussion

4.1 Result

Based on 23 scenarios of black box testing, the testing result found that the proposed system was in accordance with the design planning. The system can overcome problems at neighborhood community management information system.

- Dashboard menu presented the summary information of total income, total expense and the number of registered users in real time as shown in Figure 9. Such information is displayed in an easy-to-understand manner, such as summary diagrams and axisymmetric diagrams to enable users to monitor a cash flow with a simple operation. Additionally, the dashboard page also links to a few other key areas of the software including the User Management, giving Super-Admins access to user data where new accounts can be added or user permissions edited.

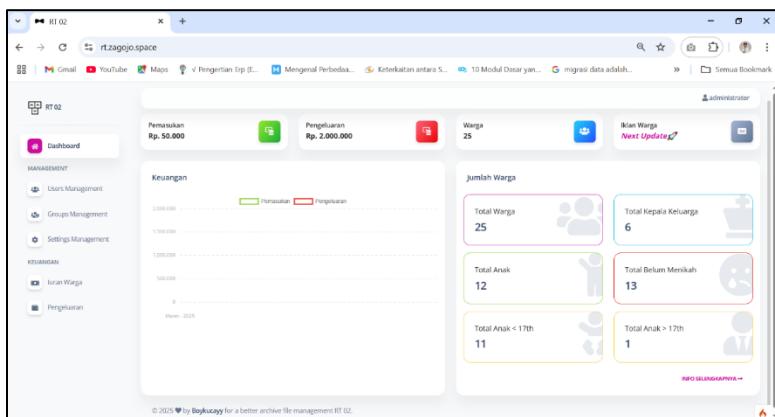


Figure 9 Dashboard menu

- User management menu as shown in Figure 10 shows a page used to manage the data of all neighborhood community members registered in the neighborhood community management information system. This menu displays a list of users in a table, including information such as family identity card, national identity number, username, full name, occupation, gender, marital status, address number, mobile phone numbers, car plates number, name of family

members, and joining date. Therefore, the administrator can easily see the detailed information and make some edits if necessary.

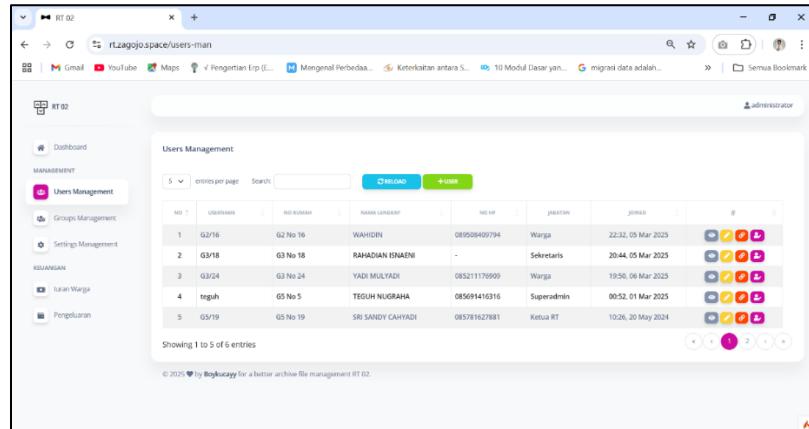


Figure 10 Users management menu

c. Figure 11 shows the settings management menu where it is a dedicated page for performing overall system settings within the RT Management Information System. Through this page, users with Super-Admin privileges can perform initial configuration and manage basic information related to the RT environment.

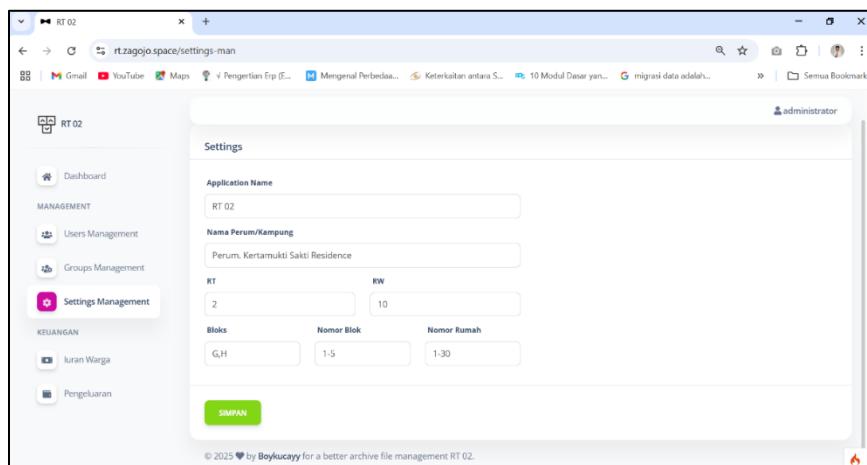


Figure 11 Settings management menu

d. Figure 12 shows the resident contribution menu is a page that displays a list of residents who have paid monthly or other contributions in the neighborhood community management information system. The information on this page helps the head of neighborhood community and his team to monitor and manage contribution data in a structured and transparent manner.

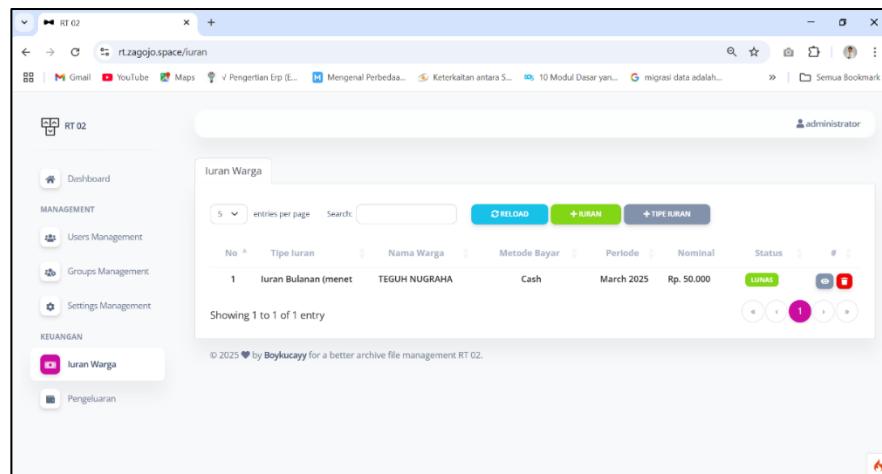


Figure 12 Monthly fee menu

e. Usability Evaluation Results

Following the black box functional testing, a usability evaluation was conducted using the System Usability Scale (SUS) to assess user acceptance and satisfaction with the developed system. The evaluation involved 20 participants consisting of RT administrators (n=3), treasurers (n=2), and active residents (n=15) who had been using the system for a minimum of two weeks.

The SUS assessment yielded an average score of 82.5 out of 100, which falls into the "Excellent" category according to the standard SUS interpretation scale. This high score indicates strong user acceptance and demonstrates that the system successfully meets usability requirements from the users' perspective. The detailed results show that 85% of respondents indicated they would like to use the system frequently, 90% found the system easy to use without requiring technical assistance, 80% felt the system functions were well integrated, 95% felt confident using the system, and 75% stated that most people would learn to use the system quickly.

Additionally, the system demonstrated significant improvements in operational efficiency compared to the previous manual methods. The time required to generate monthly financial reports was reduced from three days to approximately one hour, representing a 96% reduction in processing time. Data entry errors decreased by 90%, from an average of 10 errors per month to only 1 error per month. Furthermore, the average response time for resident inquiries improved from 2-3 days to less than 24 hours due to the centralized and easily accessible information system.

These quantitative results validate that the web-based RT management information system not only functions correctly from a technical perspective but also delivers substantial practical benefits in terms of efficiency, accuracy, and user satisfaction.

4.2 Discussion

Currently the administrative activity in neighbourhood community services at Kertamukti Sakti Residence still in manually for recording of resident contribution, population management and information dissemination which results in low efficiency and high error rate. Another outcome is delayed delivery of information and poor resident satisfaction with local community services.

To improve the satisfaction and involvement of citizens, quick and accurate service is important. By the neighborhood community management information system, administrator can communicate in a more timely manner, dues payment can be paid more accurate, and to meet the needs of residents. Such a system allows the community's resident administrators to maintain good relations with residents.

Apart from correctness and availability, transparency is another important characteristic that information systems can promote. All residents visit the Add Contributions, Financial Reports and

Neighborhood Community Activity Agendas publicly Discuss options with all community members, no privatized information etc. This builds the confidence of our neighbors in community leaders and paves an orderly and open society.

When the system was being developed, third-party correspondence with residents and neighborhood community on administrative matters were taken into account as much as possible from the development phase. Rapid iteration and repeated evaluation provided developers with an intuition of system performance, enabling them to shape it to real-world requirements. A few new features such as attending cleaning fee, online submission the letter's list and resident data carpeted floor by block before housing were added during the development. Agile development methodologies can allow such adaptations with less impact to the larger system development.

Agile development processes made for a close collaboration between the developing team, administrator of neighborhood communities and residents' representatives. As a result of that partnership, the neighborhood management perspective shaped the system and has developed an efficient information management system to aid administration and service delivery to residents. With the library of CodeIgniter, so it can be built more efficiently neighbourhood community information system-based management Kertamukti Sakti Housing Complex Residency. CodeIgniter offers a simple and light-weight MVC (Model, View, Controller) architecture for feature development, allowing developers to more efficiently develop resident data management functions and record contributions as well as activity reports.

5 Conclusion

The development of a web-based Neighborhood Community Management Information System using the CodeIgniter framework and Agile development methodology at Kertamukti Sakti Residence successfully simplified contribution recording, population data processing, and information management. The system ensures transparency and improves the organization of community administration by providing centralized and easily accessible records. Based on the evaluation results, the system achieved an average System Usability Scale (SUS) score of 82.5, categorized as "Excellent," indicating strong user acceptance and satisfaction. Quantitative measurements demonstrate significant improvements in operational efficiency: the time required to generate financial reports was reduced from three days to one hour (96% reduction), while data entry errors decreased by 90%. These findings provide concrete evidence that the system significantly increases efficiency, accuracy, and residents' trust in community management.

6 Limitation and Future Research

Despite these positive results, this study has several limitations. First, the evaluation was conducted only within a single neighborhood community (RT 002/RW 010, Kertamukti Sakti Residence), which may limit the generalizability of findings to other community contexts with different characteristics or needs. Second, the usability assessment was performed shortly after system deployment (two weeks), and long-term user satisfaction, system sustainability, and adoption patterns require further longitudinal investigation. Third, the current system still relies on manual payment recording and verification, which could benefit from automation to further reduce administrative burden.

Future work will address these limitations by expanding the system's implementation to multiple neighborhoods with diverse demographics to assess scalability and adaptability. Long-term studies will be conducted to evaluate sustained impact on community management practices. Additionally, the system will be enhanced through integration with digital payment platforms such as QRIS, e-wallets, and direct bank transfers to streamline financial transactions. Other planned features include automated payment reminders, data analytics dashboards for trend analysis, mobile application support for improved accessibility, and notification systems for real-time community announcements. These enhancements aim to further improve user experience, operational efficiency, and support sustainable digital transformation in community management practices.

Reference

- [1] A. Peryanto, D. Susanto, Y. F. Widodo, and A. Y. Aditya, "Digitalisasi Sistem Keuangan dan Informasi berbasis *Android* di RT 05 Karangsari, Sendangtirto, Berbah, Sleman," *Jurnal Pengabdian Masyarakat - PIMAS*, Vol. 4, No. 1, pp. 38–45, 2025, DOI: <https://doi.org/10.35960/pimas.v4i1.1796>.
- [2] V. Miftahuljannah and A. Suharso, "Pengimplementasian Berbagai *WEB* berdasarkan Kebutuhan Pengguna dengan menggunakan Metode *Systematic Literature Review*," *INFOTECH journal*, Vol. 9, No. 2, pp. 401–405, Aug. 2023, DOI: <https://doi.org/10.31949/infotech.v9i2.6341>.
- [3] M. Ikhsan, Helmina, Z. Akbar, R. Dani, and O. Ediansa, "Sosialisasi dan Pelatihan *Framework Codeigniter* untuk Mahasiswa Universitas Muhammadiyah Jambi," *ASPIRASI: Publikasi Hasil Pengabdian dan Kegiatan Masyarakat*, Vol. 2, No. 1, pp. 70–76, Dec. 2023, DOI: <https://doi.org/10.61132/aspirasi.v2i1.138>.
- [4] I. Larasati, A. N. Yusril, and P. Al Zukri, "Systematic Literature Review Analisis Metode *Agile* dalam Pengembangan Aplikasi *Mobile*," *SISTEMASI*, Vol. 10, No. 2, p. 369, May 2021, DOI: <https://doi.org/10.32520/stmsi.v10i2.1237>.
- [5] A. O. Pratiwi, M. R. Qisthiano, and N. A. Aulia, "Web Game Implementasi Metode *Agile* dalam Pembuatan Sistem Game Edukatif berbasis *Web* untuk Pembelajaran Bahasa Inggris Tingkat Sekolah Dasar," *Jurnal Komtika (Komputasi dan Informatika)*, Vol. 9, No. 1, pp. 100–111, May 2025, DOI: <https://doi.org/10.31603/komtika.v9i1.13495>.
- [6] R. F. I. Haq, R. Pandiya, and R. Setyadi, "Rancang Bangun Sistem Informasi Keuangan Tingkat RT menggunakan Metode *Agile*," *JATI (Jurnal Mahasiswa Teknik Informatika)*, Vol. 8, No. 1, pp. 48–56, Feb. 2024, DOI: <https://doi.org/10.36040/jati.v8i1.8558>.
- [7] D. Kurniadi, Y. Septiana, A. R. Ningsih, and H. Suhendar, "Perancangan Sistem Informasi Kependudukan di Lingkungan Rukun Tetangga atau Rukun Warga berbasis *Web*," *Jurnal Algoritma*, Vol. 18, No. 2, pp. 385–395, Jan. 2022, DOI: <https://doi.org/10.33364/algoritma.v.18-2.986>.
- [8] A. R. Ismail, Y. Yunus, B. Senung, and A. M. M. Pratama, "Implementasi Metode *Agile* pada Perancangan Sistem Informasi Wisata Desa," *Journal of Information System Research (JOSH)*, Vol. 5, No. 4, pp. 1455–1463, Jul. 2024, DOI: <https://doi.org/10.47065/josh.v5i4.5676>.
- [9] F. A. Fauzan, A. Faroqi, and S. Mukaromah, "Analisis Kepuasan Wisatawan Kota Surabaya dengan Model *End User Computing Satisfaction* (Studi Kasus: Tiket Wisata Surabaya)," *Saturnus : Jurnal Teknologi dan Sistem Informasi*, Vol. 3, No. 1, pp. 45–58, Jan. 2025, DOI: <https://doi.org/10.61132/saturnus.v3i1.638>.
- [10] A. Indriani, D. K. Pramudito, and Suherman, "An Agile Development Method of Employee Co-op Shop Web-based Sales Information System ", *jidt*, Vol. 6, No. 2, pp. 123–132, Jun. 2024. DOI: <https://doi.org/10.60083/jidt.v6i2.538>.
- [11] M. Rizal, A. Arifin, and A. Bahtiar, "Sistem Informasi Manajemen pada Pusat Penelitian dan Pengabdian Masyarakat (P3M) Universitas Dipa Makassar Berbasis *Web*," *Jurnal Minfo Polgan*, Vol. 13, No. 1, pp. 219–224, Feb. 2024, DOI: <https://doi.org/10.33395/jmp.v13i1.13537>.
- [12] A. Kurniawan and Fauziah, "SIMPEL (Sistem Informasi Manajemen Pelatihan) Internal BRI menggunakan Metode *Agile* dengan Model *Extreme Programming* dan *Algoritma Brute Force*," *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, Vol. 7, No. 2, pp. 270–279, Apr. 2023, DOI: <https://doi.org/10.35870/jtik.v7i2.754>.
- [13] S. M. Vergantana and I W. Y. Pradnya, "Perancangan Sistem Informasi Inventori Obat E-Farms berbasis *Website* dengan Metode *Agile*," *Journal Pharmactive*, Vol. 3, No. 2, Oct. 2024, DOI: <https://doi.org/10.64036/pharmactive.v3i2.61>.
- [14] F. D. Amalia and A. B. Kusdinar, "Penerapan Algoritma Dijkstra untuk menemukan Tempat Ibadah Terdekat di Kota Sukabumi menggunakan Metode *Agile*," *Joutica*, Vol. 9, No. 2, pp. 196–208, Sep. 2024, DOI: <https://doi.org/10.30736/informatika.v9i2.1311>.

- [15] L. W. Silvana, W. Agustiono, and F. A. Mufarroha, "Implementasi Payment Gateway Pada Marketplace Digital Product Berbasis Website Dengan Menggunakan Metode Agile," *Journal of Scientech Research and Development*, vol. 6, no. 2, pp. 451–464, Dec. 2024, doi: <https://doi.org/10.56670/jsrd.v6i2.523>.
- [16] T. A. Pertiwi, N. Try Luchia, P. Sinta, A. Dahlia, I. Rachmat Fachrezi, and R. Aprinastya, "Perancangan dan Implementasi Sistem Informasi Absensi berbasis *Web* menggunakan Metode *Agile Software Development*," *Jurnal Testing dan Implementasi Sistem Informasi*, Vol. 1, No. 1, pp. 53–66, Mar. 2023, Doi: <https://doi.org/10.55583/jtisi.v1i1.325>.
- [17] Uminingsih, M. N. Ichsanudin, M. Yusuf, and S. Suraya, "Pengujian Fungsional Perangkat Lunak Sistem Informasi Perpustakaan dengan Metode *Black Box Testing* Bagi Pemula," *STORAGE: Jurnal Ilmiah Teknik dan Ilmu Komputer*, Vol. 1, No. 2, pp. 1–8, May 2022, DOI: <https://doi.org/10.55123/storage.v1i2.270>.
- [18] A. Dillah, G. F. Nama, D. Budiyanto, and M. A. Muhammad, "Rancang Bangun Aplikasi Monitoring Operasi P2tl Pengukuran Tidak Langsung 2 Phasa di PT. PLN (Persero) Unit Pelaksana Pelayanan Pelanggan (Up3) Metro," *Jurnal Informatika dan Teknik Elektro Terapan*, Vol. 12, No. 3, Aug. 2024, DOI: <https://doi.org/10.23960/jitet.v12i3.4458>.
- [19] M. R. Wayahdi and F. Ruziq, "Pemodelan Sistem Penerimaan Anggota Baru dengan *Unified Modeling Language (UML)* (Studi Kasus: Programmer Association of Battuta)," *Jurnal Minfo Polgan*, Vol. 12, No. 1, pp. 1514–1521, Aug. 2023, DOI: <https://doi.org/10.33395/jmp.v12i1.12870>.
- [20] A. T. Hidayati, A. E. Widyantoro, and H. J. Ramadhami, "Perancangan Sistem Informasi Wirausaha Mahasiswa (Siwirma) berbasis *Web* dengan *Unified Modelling Languange (UML)*," *Jurnal Penelitian Rumpun Ilmu Teknik*, Vol. 2, No. 4, pp. 86–107, Nov. 2023, DOI: <https://doi.org/10.55606/juprit.v2i4.2906>.
- [21] R. Auliazmi, G. Rudiyanto, and R. D. W. Utomo, "Kajian Estetika *Visual Interface* dan *User Experience* pada Aplikasi Ruangguru *Aesthetic Studies of Visual Interface and User Experience of The Ruangguru Application*," *Jurnal Seni dan Reka Rancang: Jurnal Ilmiah Magister Desain*, Vol. 4, No. 1, pp. 21–36, Aug. 2021, DOI: <https://doi.org/10.25105/jsrr.v4i1.9968>.
- [22] A. Bangor, P. T. Kortum, and J. T. Miller, "An Empirical Evaluation of the System Usability Scale," *International Journal of Human-Computer Interaction*, Vol. 24, No. 6, pp. 574–594, Jul. 2008, DOI: <https://doi.org/10.1080/10447310802205776>