

# An efficient and interactive android-based neighborhood management

Haris Junianto <sup>a,1,\*</sup>, Dhanar Intan Surya Saputra <sup>a,2</sup>, Rujianto Eko Saputro <sup>a,3</sup>

<sup>a</sup> Universitas Amikom Purwokerto, Watumas, Purwanegara, Kec. Purwokerto Utara, Kabupaten Banyumas, Jawa Tengah, Indonesia

<sup>1</sup> [hariescuy@gmail.com](mailto:hariescuy@gmail.com); <sup>2</sup> [dhanarsaputra@amikompurwokerto.ac.id](mailto:dhanarsaputra@amikompurwokerto.ac.id); <sup>3</sup> [rujianto@amikompurwokerto.ac.id](mailto:rujianto@amikompurwokerto.ac.id)

\* corresponding author

## ARTICLE INFO

### Article history

Received January 8, 2024

Revised April 18, 2024

Accepted July 3, 2024

### Keywords

Android-based residential

Management application

System information

Agile approach

Real-time administrative

## ABSTRACT

This research aims to design and develop an Android-based RT management information system application prototype with an Agile approach to assist RT administrators in administrating information in real-time. The Agile approach was chosen to increase flexibility and responsiveness in application development, allowing adaptation to potential user needs and changes that may occur during the development process. Hopefully, this application can improve service quality and transparency of governance at the RT level while making it easier for residents to access information and interact with RT administrators. The application development process uses an Agile approach, which involves the development team in an iterative cycle to meet user needs. The cycle in the Agile method applied in this research consists of stages: planning & analysis, design, implementation & testing, and deployment (presentation to users). If the user accepts the results, the process is terminated; however, the cycle will be repeated if repairs or updates are still needed. The user interface (UI) testing results show a score of 299 out of a total score of 360, while user acceptance testing (UAT) gets a score of 255 out of 300. When combined, UI and UAT achieve a result of 84%. The percentage value is obtained from 41.50% for User Interface Testing (UI) and 42.50% for User Acceptance Testing (UAT); this shows good application performance in data management, which represents the achievement of research objectives with an application that can manage citizen data, RT finances, event scheduling, and RT news. The Agile approach used in application development provides the flexibility necessary to adapt to changing user needs, offering solutions to the challenges faced by neighborhood association administrators in carrying out their duties. This aligns with Agile principles, emphasizing user collaboration and responsiveness to change.

This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



## 1. Introduction

Neighborhood associations are one of the community organizations in Indonesia that play a crucial role in providing administrative and social services to residents at the grassroots level. These neighborhood associations are also responsible for managing population data, finances, events, and information related to the welfare of the residents. However, in practice, many neighborhood associations still need to work on performing these tasks, such as having limited human resources, facilities, and infrastructure and a lack of transparency and accountability [1]. Often, this situation has the potential to trigger conflict between the Rukun Tetangga administrators and residents. Apart from that, this also hinders the running of government administration processes because the system is still conventional, which requires quite a long time for administrative processing [2]. Therefore, there is a need for a management information system that can assist Neighborhood Association officials in enhancing their performance and the quality of their services [3]. One solution can be developing an Android-based Neighborhood Association Management Information System application.

Neighborhood Association administrators can utilize this application for real-time administrative processes and information dissemination. Residents can also access it to obtain information and interact with Neighborhood Association administrators. This application can also provide convenience and comfort to users since it can be accessed through smartphones, which are commonly owned by the community [4]. This research employs the Agile approach to develop this application, a software development methodology based on iterative, responsive, and repetitive work processes that adapt to changes [5]. The Agile approach was chosen because it is considered suitable for the characteristics of the neighborhood association application development project, which involves dynamic and complex user requirements and requires intensive collaboration between the development team and users [6]. By employing the Agile approach, the resulting application is expected to optimally fulfill the users' needs and expectations.

This research aims to present an application prototype to increase efficiency, improve the quality of neighborhood management services in neighborhood administration, and provide real-time information regarding residents' activities [7], [8]. So, it is hoped that it can simplify the service process, increase citizen participation in their respective environments, and bring changes to neighborhood management.

## 2. Method

Research methodology is a process for identifying the root problems and finding solutions [9]. Flexibility is required to ensure optimal governance of government information systems due to the involvement of various public interests. Agile methods are more relevant than other methods, especially in developing government administration applications, such as within the scope of Rukun Tetangga management [10]. Agile is considered the most suitable because it has the advantage of being exceptionally responsive to dynamic and complex changes in user needs [11]. This allows developers to respond more quickly to changing user needs and applications to continue to develop according to the demands of an ever-changing environment [12]. Therefore, in this research, the author chose the Agile method to integrate the principles of flexibility, collaboration, and responsiveness in developing an effective and efficient government information system application [13]. One of the approaches that facilitates rapid software development is Agile Development as show in Fig. 1 [14].

The stages of the Agile method used in this research are as follows:

- Planning

The planning phase in designing the SinaRT application involves identifying user and stakeholder needs through random online surveys conducted among RT residents. The survey results will guide user preferences, needs, and expected features as the foundation for designing a system that aligns with their expectations [15].

- Analysis

In the analysis phase, the researcher will analyze the functional and non-functional requirements of the system [16]. This involves creating user stories and project backlogs encompassing user needs and identifying tasks to accomplish. Through meticulous analysis, the researcher will ensure that the system's design can meet the expectations of users and stakeholders when managing the SinaRT application.

- Design

In the SinaRT application design phase, the researcher will design the system's architecture, user interface, and database structure [17]. This design will ensure that the system has a strong foundation, an intuitive interface, and efficient data storage [18]. With a focus on good design, the team will ensure that the application can provide an optimal user experience and effectively support RT management tasks.

- Implementation

In the implementation phase of the SinaRT application design, the researcher will begin to implement source code, integrate modules, and conduct unit testing. The researcher will translate the system design into actual code, ensure that various components work seamlessly, and test

code units to ensure optimal quality and functionality. The implementation phase is crucial in bringing the System Information Management RT application into reality.

- Testing

In the testing phase of the SinaRT application design, the researcher will meticulously test the system's functionality, performance, and usability. The researcher will evaluate every feature and function based on predefined specifications, identify functional issues or bugs, and create test scenarios covering various user situations. The results of this phase will ensure that the system is ready for practical use by users, with optimal performance and ease of service.

- Deployment

In the Deployment phase of the SinaRT application design, the application is ready for implementation in the RT environment. System installation on servers will be carried out, users will receive necessary training, and the system's results will be evaluated. This phase marks the application's launch into actual operations and ensures RT can effectively use the application. However, this research only represents a prototype of the actual implementation.



**Fig. 1.** Agile Method

### 3. Results and Discussion

#### 3.1. Requirements Identification

Requirements identification is the initial step in the design of the RT Management Information System application [19]. This phase aims to determine and document the needs desired by users and stakeholders related to the application's development. From the data obtained, two types of requirements can be classified, namely functional and non-functional requirements, as follows:

- Functional Requirements

Functional requirements in the design of the SinaRT application include managing resident data, managing RT finances, scheduling residents' activities, and recording RT-owned inventories.

- Non-Functional Requirements

Non-functional requirements in the design of the SinaRT application encompass easy access through all residents' smartphones, real-time access, and optimal performance

#### 3.2. Requirement Analysis

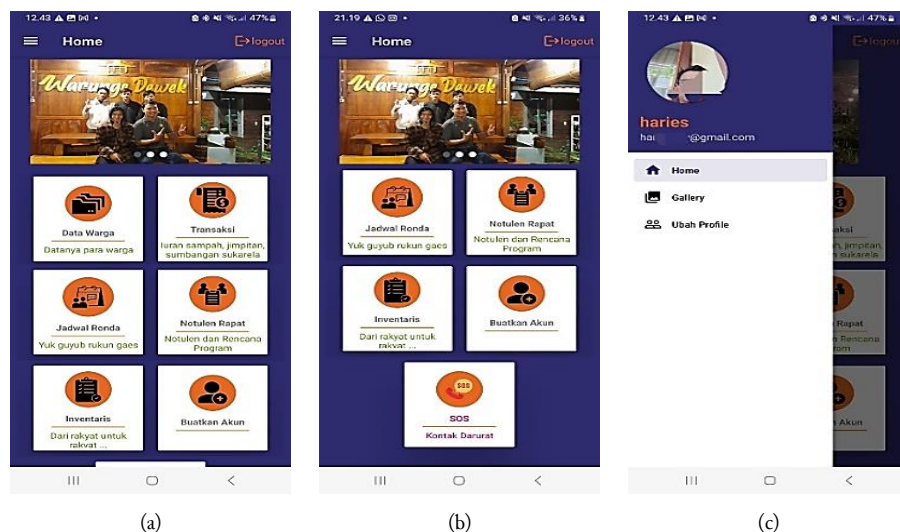
After the researcher had conducted requirement identification, the results of the requirement identification for designing the SinaRT application were obtained as show in Table 1.

**Table.1** Needs Analysis

Role	Menu Access requirement	User Type
Chairman of the neighborhood & Treasurer	Login	Admin
	Input, Update, and Delete citizen data	
	View Citizen Data	
	Input, Update, Delete Incoming & Outgoing Transactions	
	View Transactions	
	Input, Update, and Delete patrol schedule	
	See Patrol Schedule	
	Input, Update, Delete Meeting Minutes	
	Input, Update, Delete Inventory	
	View Inventory	
	Edit Account Profile	
	SOS (Emergency Contact)	
	Create an Account	
Public	Logout	User
	Login	
	View Citizen Data	
	View Transactions	
	See Patrol Schedule	
	View Inventory	
	SOS (Emergency Contact)	
	Edit Account Profile	
	Logout	
	Logout	

### 3.3. User Interface Design

The user interface design in the SinaRT application includes intuitive UI elements, such as navigation buttons, quickly scrollable lists, a calendar for scheduling activities, and action menus that facilitate access to various features. Furthermore, the researchers carefully selected a color palette that was easy for the user to see and organized the information layout for easy retrieval. With an attractive visual design and easily accessible functionality, the user interface design in the SinaRT application can enhance efficiency and user experience [20]. The Fig. 2 is the user interface design appearance for each respective menu.



**Fig. 2.** (a) Design the home menu before scrolling, (b) Design the home menu before scrolling, (c) Navbar design

### 3.4. Design System

The design of the SinaRT application system incorporates various access restrictions based on user account types to ensure data security and privacy. Through this design, only admin accounts have full authorization to open, manage, create, update, and delete data in various features such as the "input data warga menu, input jadwal ronda, input transaksi, input inventaris," and the "menu buat akun for creating new user accounts." This ensures that higher-level access is only granted to users with admin roles. On the other hand, users with different types of accounts have more limited access restrictions. They can only view resident data, transaction history, and inventory, change their profiles, and view patrol schedules. This is intended to maintain data security and privacy and prevent users with more restricted roles from making changes or accessing data they shouldn't have access to [21]. This design creates a good level of control. It secures data by ensuring users only have access to information and features appropriate to their roles and responsibilities within the SinaRT application. Use case diagram show as Fig. 3.

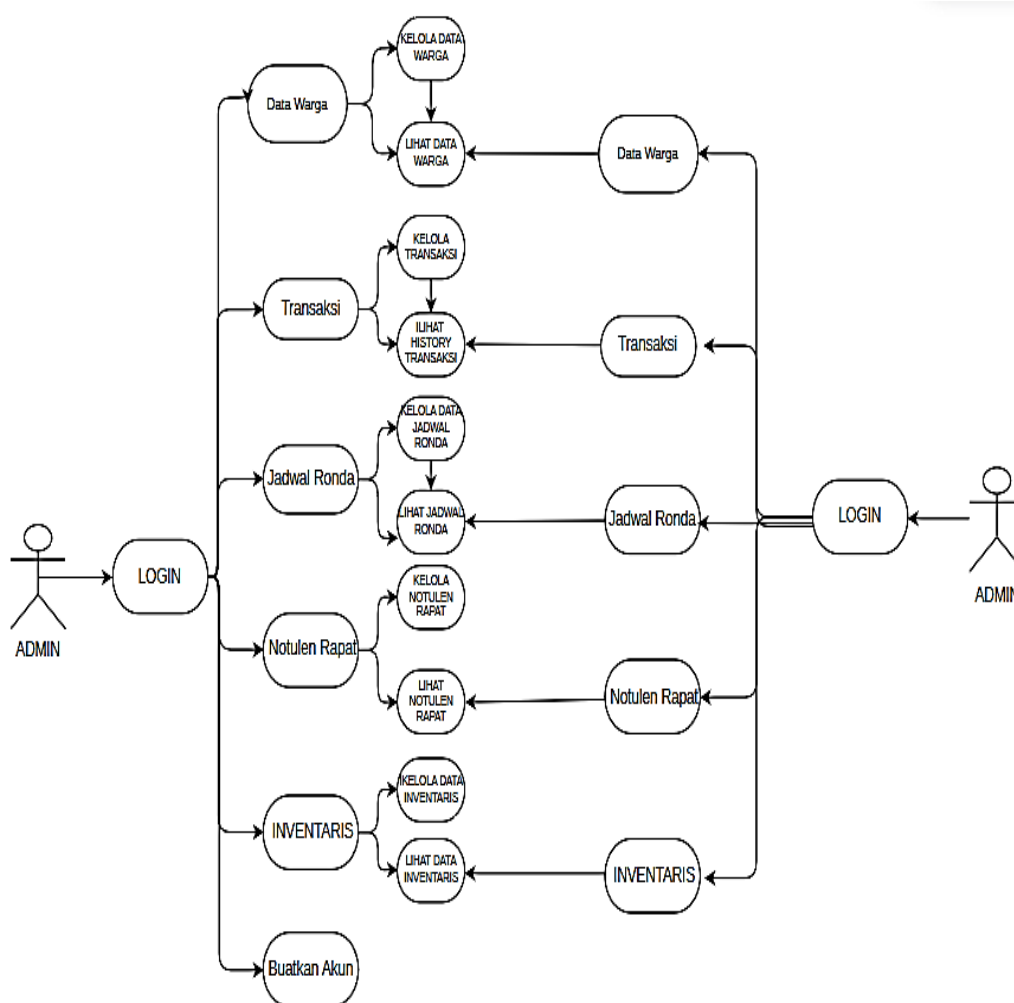


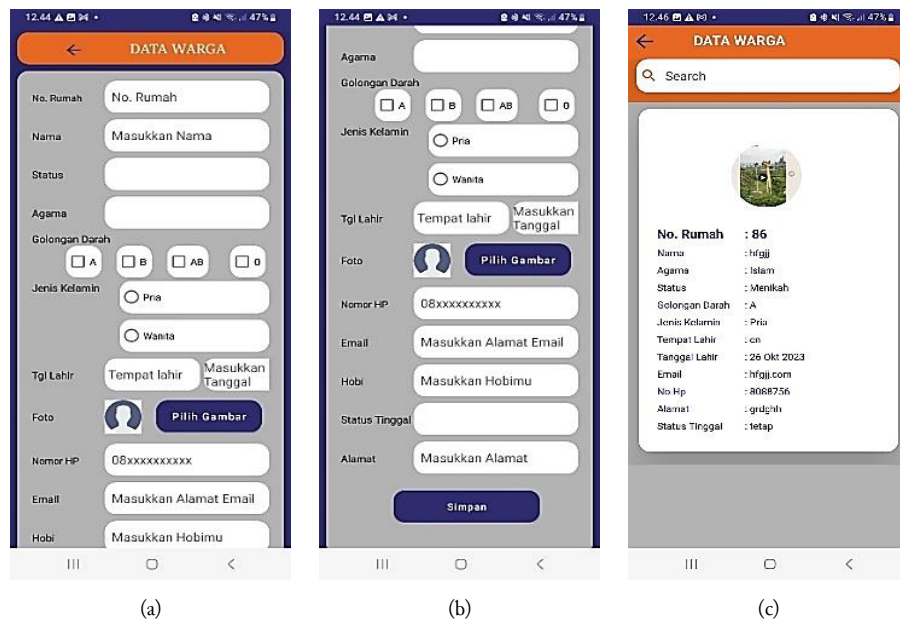
Fig. 3. Use a case diagram

### 3.5. Features in the SinaRT application

Within the SinaRT application, several menu features are designed to focus on data management and communication in the context of information management within a specific community or neighborhood. Below is a brief explanation of each menu feature found within the SinaRT application.

- Data Warga (citizen data menu) as show in Fig. 4

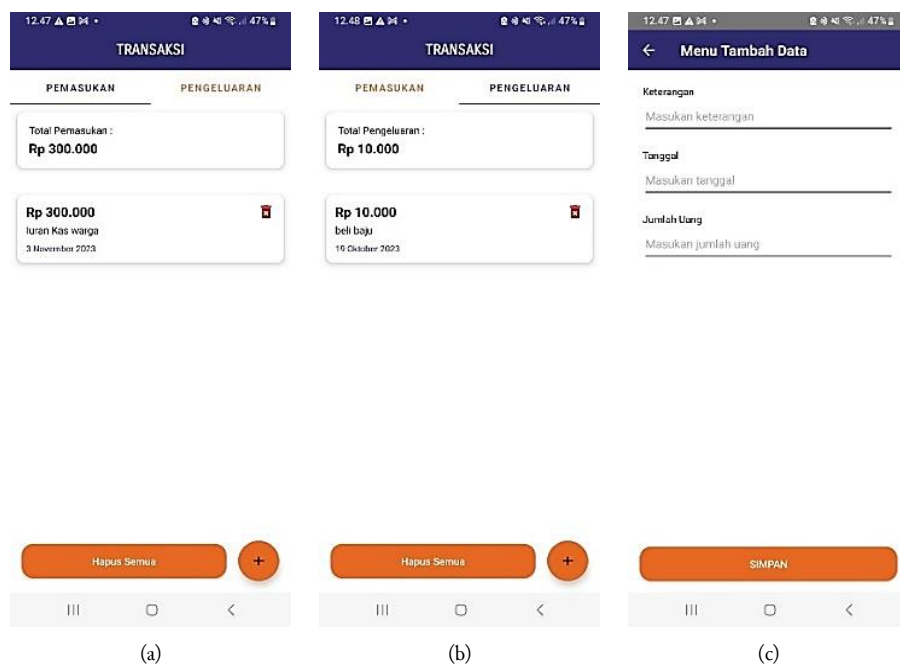
This feature allows users to manage the identification data of neighborhood residents. Within it, users can record and access personal data, addresses, contact numbers, and other essential information about each resident. This facilitates resident data management and enhances internal communication.



**Fig. 4.** (a) Input citizen data display menu before scrolling, (b) Input citizen data display menu after scrolling, (c) View Citizen Data

- Transaksi (Transaction menu) as show in [Fig. 5](#)

This feature enables the recording and tracking financial transactions, such as dues payments or other economic activities involving the Neighborhood Fund. It helps ensure transparency and accountability in community financial management.



**Fig. 5.** (a) Exit transaction menu display, (b) Incoming transaction menu display, (c) display adds transaction notes

- Jadwal Ronda (patrol schedule menu) as show in [Fig. 6](#).

This feature creates and shares patrol or security schedules by security personnel or neighborhood administrators. Users can view plans, identify individuals on duty, and receive notifications related to patrol schedules.

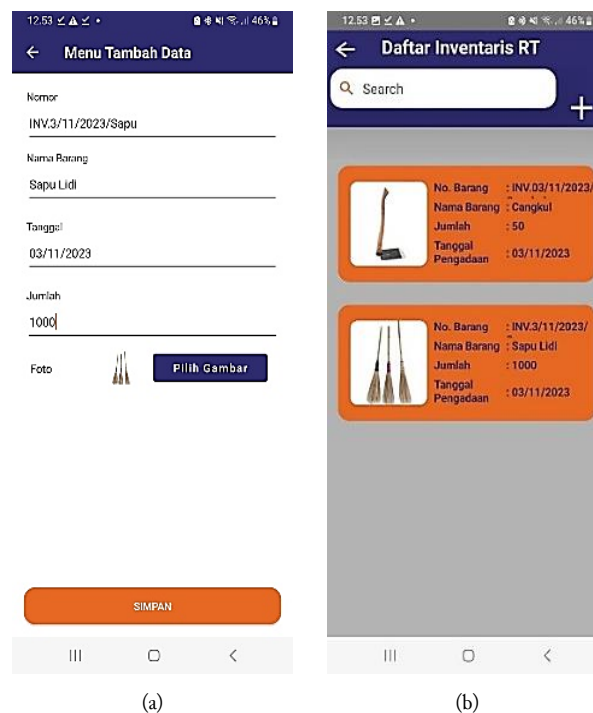




**Fig. 6.** (a) Input Patrol Schedule, (b) display Patrol Schedule Input

- Inventaris (inventory menu) as show in [Fig. 7](#)

This feature allows users to record and manage the community's inventory or belongings, such as fire extinguishers, cleaning equipment, or other common facilities. It aids in inventory maintenance and simplifies item retrieval when needed.



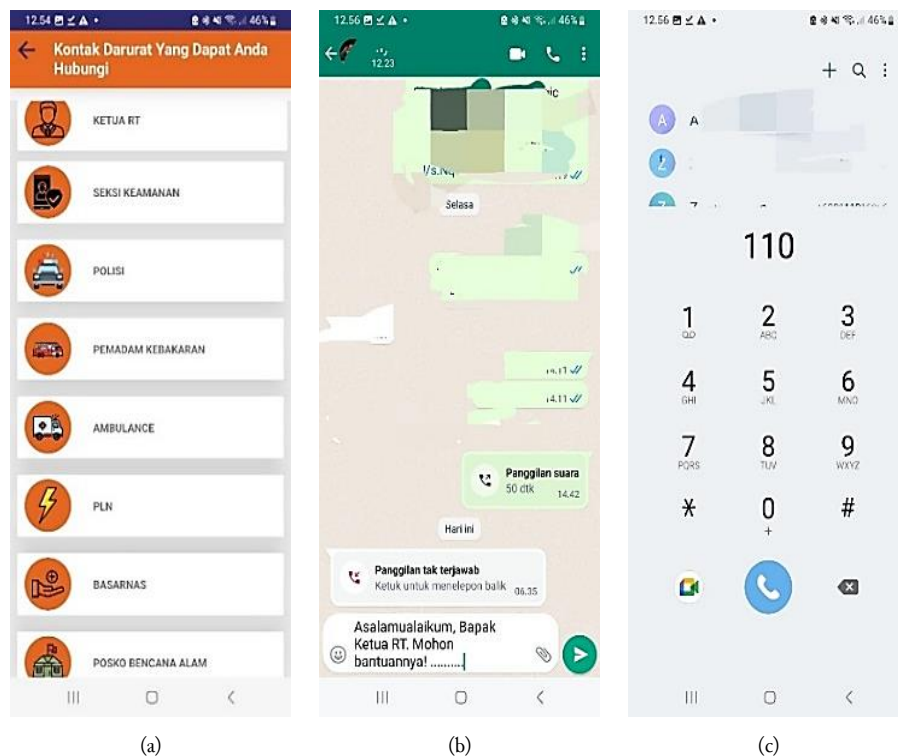
**Fig. 7.** (a) main page of Notulen Menu, (b) Notulen Input Menu Display

- Notulen (meeting minutes menu)

This feature helps record and share minutes of community meetings or events. It helps maintain vital records of each session, document decisions, and store them for future reference.

- SOS (emergency contact menu) as show in Fig. 8

This feature is designed for emergencies. Users can quickly access a list of emergency contacts or emergency services, such as police phone numbers, hospitals, fire departments, or other crucial contacts when needed.



**Fig. 8.** (a) emergency call menu list display, (b) Display when a contact is selected, and the number is connected to the WhatsApp application, (c) Display when a contact is selected, and the number only accepts phone calls

### 3.6. Application Testing

Researchers employed the black-box testing method to test the SinaRT application. Before conducting the tests, researchers collected data through observations and user questionnaire surveys to measure user responses and preferences. The testing of the SinaRT application involves multiple stages, including user interface testing to ensure an intuitive user interface and user acceptance testing to gauge users' readiness to accept the application. The collected data will be processed using SPSS version 23 to assess the validity and reliability of the test results, ensuring that the SinaRT application meets the expected quality standards. The following are the stages in testing the SinaRT application.

- Data Collection
  - Observation  
Researchers observed the SinaRT application's interface and related functions during this stage
  - Questionnaire Survey  
In this stage, the researchers used a questionnaire survey to gather data from the testing results. Before conducting this stage, the researchers made the following preparations.
  - Questionnaire Development  
In this phase, the researchers compiled a list of questions as listed in the questionnaire [Table. 2](#) that had been successfully prepared.



**Table.2** Questionnaire List

Code	Type of testing	list of questions
X1.1	User Interface Testing	Did you find this application's user interface (UI) easy to use?
X1.2		Are the UI elements, such as buttons and menus, responsive to your touch or interaction?
X1.3		Are the colors used in the application's user interface easy to read and pleasing to the eye?
X1.4		Is this application's user interface intuitive, allowing you to understand how to use it quickly?
X1.5		Are you satisfied with how information, such as text and images, is presented in this application?
X1.6		Do you find the appearance of the application's interface in line with your visual preferences?
X2.1	User Acceptance Testing (UAT)	Did you find the application easy to use and suitable for your needs?
X2.2		Does this application meet your needs well, or do you feel there are some missing or necessary features?
X2.3		What is your satisfaction level with how information is presented in the application, such as text, images, and graphics?
X2.4		What is your overall satisfaction level with the experience of using this application?
X2.5		Would you recommend this application to friends or colleagues based on your user experience?

Table 3 is a Likert scale table for each respondent's answers:

**Table.3** Likert Scale

Item	value
Strongly agree	1
Agree	2
Enough	3
Disagree	4
Strongly Disagree	5

#### - Distributing Questionnaires

In this stage, the researcher identified respondents who were asked to try the SinaRT application before filling out the online questionnaire using Google Forms. The researcher involved 12 respondents with diverse backgrounds. Table 4 are the questionnaire results from 12 respondents.

**Table.4** User Interface Testing Questionnaire Results

X1.1	X1.2	X1.3	X1.4	X1.5	X1.6	TOTAL
5	5	5	5	5	5	30
3	5	3	4	3	3	21
5	5	5	5	5	5	30
4	4	3	5	4	3	23
4	4	4	4	3	4	23
4	4	4	4	4	4	24

X1.1	X1.2	X1.3	X1.4	X1.5	X1.6	TOTAL
4	5	4	4	5	4	26
4	4	4	4	4	4	24
5	4	5	5	4	5	28
4	3	5	3	4	3	22
4	3	2	4	3	2	18
5	5	5	5	5	5	30

User Acceptance Testing (UAT) Questionnaire Results as show in [Table 5](#).

**Table.5** User Acceptance Testing (UAT) Questionnaire Results

X2.1	X2.2	X2.3	X2.4	X2.5	TOTAL
5	5	5	5	5	25
5	4	3	4	5	21
5	5	5	5	5	25
4	3	3	4	4	18
5	5	3	4	5	22
4	3	4	4	4	19
4	4	5	4	5	22
4	3	4	4	4	19
5	2	5	5	5	22
5	4	5	3	5	22
2	3	3	2	5	15
5	5	5	5	5	25

- Data Preprocessing

In data processing, researchers test the validity and reliability of the obtained data to ensure its accuracy and dependability

- Validity testing

Validity testing is a process or method used to assess the extent to which an instrument or measurement tool can measure what it is intended to measure [22]–[24]. Validity testing results provide the foundation for considerations in decision-making and concluding the analyzed data [25]. The results of data processing that have undergone validity testing are show in [Table 6](#).

**Table.6** of User Interface Testing Validity Test Results

Code	Total X1	The level of significance (5%)	Description
X1.1	.830**	0,602	Valid
X1.2	.685*		Valid
X1.3	.817**		Valid
X1.4	.689*		Valid
X1.5	.851**		Valid
X1.6	.953**		Valid
TOTAL_X1	1		Valid

[Table.7](#) show of User Acceptance Testing (UAT) Validity Test Results

**Table.7** of User Acceptance Testing (UAT) Validity Test Results

code	Total X2	The level of significance (5%)	Description
X2.1	.845**	0,602	Valid
X2.2	.703*		Valid
X2.3	.709**		Valid
X2.4	.779**		Valid
X2.5	0,506	0,602	Valid
TOTAL_x2	1		Valid

- Reliability Test

The reliability testing of the sinaRT application testing questionnaire data using Cronbach's Alpha value is a crucial step in ensuring the reliability and consistency of the measurement instrument [26]. The Cronbach's Alpha value is a statistical measure used to assess the internal surface of the questionnaire or measurement instrument used in research or testing. In the intelligent application testing context, the threshold value for validity is set at 0.60. This means that if the Cronbach's Alpha value generated in the questionnaire testing exceeds 0.60, the questionnaire is considered valid, indicating that the instrument is reliable in measuring the desired constructs. Conversely, if the Cronbach's Alpha value is less than 0.60, the questionnaire is deemed invalid [27], and the measurement instrument may need to be revised or improved. Here are the results of the reliability testing of sinaRT application testing data.

**Table.8** of User Interface Testing Case Processing Summary

	N	%
Cases Valid	12	100,0
Excluded	0	0,0
Total	12	100,0

**Table.9** Reliability Statistics

Cronbach's Alpha	N of Items
0,888	6

**Table.10** of User Acceptance Testing (UAT) Case Processing Summary

	N	%
Cases Valid	12	100,0
Excluded	0	0,0
Total	12	100,0

**Table.11** Reliability Statistics

Cronbach's Alpha	N of Items
0,753	5

The test results indicate that the data reliability testing results for User Interface Testing have a Cronbach's Alpha value of 0.888, and the User Acceptance Testing (UAT) results have a value of

0.753. These values exceed the established threshold of 0.60, as determined by the researcher. Therefore, based on these results, the questionnaire used in the sinaRT application testing has a good level of validity, as both values exceed the set threshold

- Data Processing Result as show in Fig. 9

The test results involved 12 random respondents with various backgrounds, comprising six questions about UI and five about UAT. User interface (UI) testing scored 299 out of a total score of 360, while user acceptance testing (UAT) scored 255 out of 300. With a combined total score of UI and UAT reaching 84%, with UI reaching 41.50% and UAT 42.50%, indicating that the application performance is quite good in data management. as shown in the image below.

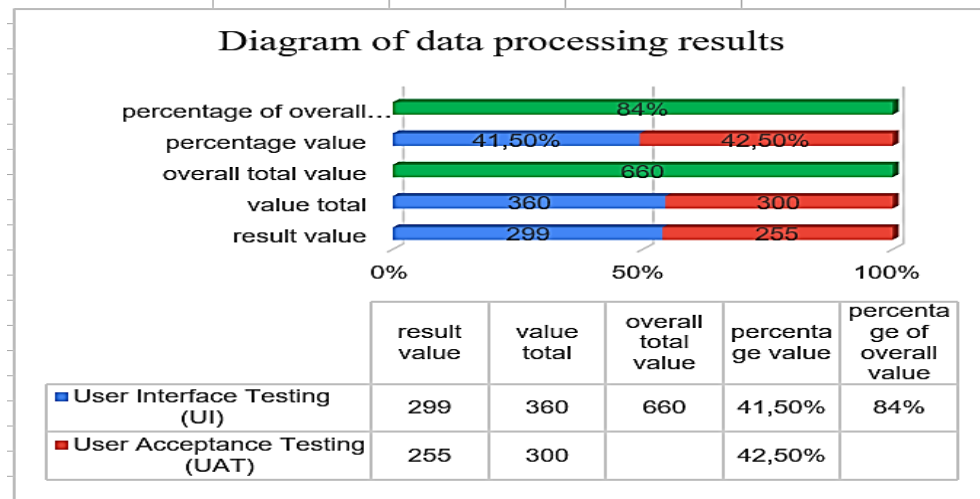


Fig. 9. Diagram of Data Processing Result

#### 4. Conclusion

In this research, the researcher successfully designed a prototype of the Residential Association Management Information System application (SinaRT) by applying the Agile methodology. Although this is only a prototype, the Agile methodology is highly suitable for applications designed to be dynamic and responsive to user needs [28], [29], particularly in a Residential Association environment. Implementing the Agile methodology in developing the SinaRT prototype allows the researcher to be more flexible in addressing changes in requirements that may arise from Residential Association users. Residential Association environments often face complex and diverse challenges and changing needs over time. The Agile methodology enables the application to adapt to these challenges and meet user requirements quickly [30]. The SinaRT application can be an effective solution for managing administrative tasks at the Residential Association level. By reducing convoluted bureaucracy, this application can enhance efficiency and simplify administrative tasks related to Residential Association management. This will significantly benefit users in the Residential Association community while improving the quality of administrative services from the test results involving 12 random respondents with various backgrounds, consisting of 6 questions about UI and five about UAT. User interface (UI) testing scored 299 out of a total score of 360, while user acceptance testing (UAT) scored 255 out of 300. With a combined total score of UI and UAT reaching 84%, UI reaching 41.50%, and UAT 42.50%, the application performance is quite good in data management. This reinforces that Agile methodology is suitable for developing prototype applications such as SinaRT, specifically designed to address user dynamics and needs in Housing Associations. With the continued development of applications like this, they are hoped to have a significant position.

#### References

- [1] E. Widiyanto and D. Kurniadi, "Design and Construction of a Web-Based RT/RW Financial Management Information System," *J. Algoritma*, vol. 18, no. 1, pp. 246–253, Nov. 2021, doi: [10.33364/algoritma/v.18-1.838](https://doi.org/10.33364/algoritma/v.18-1.838).

- 
- [2] W. Jin, "Challenges and Innovative Countermeasures Faced by Public Administration in the Context of Big Data and Internet of Things," *Math. Probl. Eng.*, vol. 2022, no. 1, pp. 1–10, Jul. 2022, doi: [10.1155/2022/8949365](https://doi.org/10.1155/2022/8949365).
- [3] R. I. Awulor, R. Obi-Mallam, and N. M. Chukwu, "Enhancing organisational decision-making through management information system," *J. Glob. Soc. Sci.*, vol. 3, no. 11, pp. 115–133, Sep. 2022, doi: [10.31039/jgss.v3i11.71](https://doi.org/10.31039/jgss.v3i11.71).
- [4] R. T. Aldisa and P. A. Soleha, "Neighborhood Service Administration System With PIECES Framework For Android-Based User Satisfaction," *J. Sist. Komput. dan Inform.*, vol. 4, no. 2, p. 288, Dec. 2022, doi: [10.30865/json.v4i2.5288](https://doi.org/10.30865/json.v4i2.5288).
- [5] S. Alsaqqa, S. Sawalha, and H. Abdel-Nabi, "Agile Software Development: Methodologies and Trends," *Int. J. Interact. Mob. Technol.*, vol. 14, no. 11, p. 246, Jul. 2020, doi: [10.3991/ijim.v14i11.13269](https://doi.org/10.3991/ijim.v14i11.13269).
- [6] J. Leong, K. May Yee, O. Baitsegi, L. Palanisamy, and R. K. Ramasamy, "Hybrid Project Management between Traditional Software Development Lifecycle and Agile Based Product Development for Future Sustainability," *Sustainability*, vol. 15, no. 2, p. 1121, Jan. 2023, doi: [10.3390/su15021121](https://doi.org/10.3390/su15021121).
- [7] R. Miranda *et al.*, "Revolutionising the Quality of Life: The Role of Real-Time Sensing in Smart Cities," *Electronics*, vol. 13, no. 3, p. 550, Jan. 2024, doi: [10.3390/electronics13030550](https://doi.org/10.3390/electronics13030550).
- [8] N. Salleh, A. Z. Zainudin, N. Md Yunus, and H. Mohd Ali, "Resident's Association Initiatives In Improving The Delivery Of Information To The Guarded Neighbourhood Community," *J. Inf. Syst. Technol. Manag.*, vol. 7, no. 25, pp. 120–126, Mar. 2022, doi: [10.35631/JISTM.725009](https://doi.org/10.35631/JISTM.725009).
- [9] B. Vo, E. Kongar, and M. F. Suarez-Barraza, "Root-Cause Problem Solving in an Industry 4.0 Context," *IEEE Eng. Manag. Rev.*, vol. 48, no. 1, pp. 48–56, Mar. 2020, doi: [10.1109/EMR.2020.2966980](https://doi.org/10.1109/EMR.2020.2966980).
- [10] A. Brusov, "Agile: Opportunities And Perspectives Of Application In Public Administration (Literature Review)," *Public Adm. Issues*, no. 2, pp. 134–158, 2022, doi: [10.17323/1999-5431-2022-0-2-134-158](https://doi.org/10.17323/1999-5431-2022-0-2-134-158).
- [11] Ayushi, D. Kaur, and S. S. Sehgal, "Agile Lifestyle of Development," in *2022 International Conference on Cyber Resilience (ICCR)*, Oct. 2022, pp. 1–5, doi: [10.1109/ICCR56254.2022.9996014](https://doi.org/10.1109/ICCR56254.2022.9996014).
- [12] P. Bellini, D. Bologna, M. Fanfani, L. A. Ipsaro Palesi, P. Nesi, and G. Pantaleo, "Rapid Prototyping & Development Life Cycle for Smart Applications of Internet of Entities," in *2023 27th International Conference on Engineering of Complex Computer Systems (ICECCS)*, Jun. 2023, pp. 142–151, doi: [10.1109/ICECCS59891.2023.00026](https://doi.org/10.1109/ICECCS59891.2023.00026).
- [13] M. Ylinen, "Incorporating agile practices in public sector IT management: A nudge toward adaptive governance," *Inf. Polity*, vol. 26, no. 3, pp. 251–271, Aug. 2021, doi: [10.3233/IP-200269](https://doi.org/10.3233/IP-200269).
- [14] S. Pratasik and I. Rianto, "E-DUK Application Development in HR Management Using Agile Development Method," *CogITO Smart J.*, vol. 6, no. 2, pp. 204–216, Dec. 2020, doi: [10.31154/cogito.v6i2.267.204-216](https://doi.org/10.31154/cogito.v6i2.267.204-216).
- [15] S. Mohseni, N. Zarei, and E. D. Ragan, "A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems," *ACM Trans. Interact. Intell. Syst.*, vol. 11, no. 3–4, pp. 1–45, Dec. 2021, doi: [10.1145/3387166](https://doi.org/10.1145/3387166).
- [16] S. B. AlKhuder and F. H. AlAli, "E-Learning Systems Requirements Elicitation," *Int. J. Virtual Pers. Learn. Environ.*, vol. 7, no. 1, pp. 44–55, Jan. 2017, doi: [10.4018/IJVPLE.2017010104](https://doi.org/10.4018/IJVPLE.2017010104).
- [17] H. Li, L. Gao, and X. Yang, "Research and Application of Application System Architecture Based on Big Data," *J. Phys. Conf. Ser.*, vol. 1852, no. 4, p. 042086, Apr. 2021, doi: [10.1088/1742-6596/1852/4/042086](https://doi.org/10.1088/1742-6596/1852/4/042086).
- [18] Z. Peng, X. Xu, and X. Wan, "Research on Data Storage Mode and Storage Access Interface Design of TSDA Server," in *2022 International Conference on Cloud Computing, Big Data Applications and Software Engineering (CBASE)*, Sep. 2022, pp. 23–26, doi: [10.1109/CBASE57816.2022.00012](https://doi.org/10.1109/CBASE57816.2022.00012).
- [19] B. Wang, R. Peng, Y. Li, H. Lai, and Z. Wang, "Requirements traceability technologies and technology transfer decision support: A systematic review," *J. Syst. Softw.*, vol. 146, pp. 59–79, Dec. 2018, doi: [10.1016/j.jss.2018.09.001](https://doi.org/10.1016/j.jss.2018.09.001).
-

- 
- [20] Liza Trisnawati, D. Setiawan, and S. Ryanto, "ANTIDERM Dermatitis Consultation Application User Interface Design Model Using Design Sprint Method," *J. Int. Multidiscip.*, vol. 1, no. 1, pp. 1–17, Jan. 2023, doi: [10.58794/jim.v1i1.193](https://doi.org/10.58794/jim.v1i1.193).
- [21] R. Punithavathi, A. D. C. Rani, K. Jamuna, V. Keerthi veni, K. P. Poornima, and K. S. P. Raajeshwari, "Preserving Data by Role based Access Control and Query Auditing System," in *2022 International Conference on Sustainable Computing and Data Communication Systems (ICSCDS)*, Apr. 2022, pp. 665–669, doi: [10.1109/ICSCDS53736.2022.9760836](https://doi.org/10.1109/ICSCDS53736.2022.9760836).
- [22] Y. Du, M. H. Arkesteijn, A. C. den Heijer, and K. Song, "Sustainable Assessment Tools for Higher Education Institutions: Guidelines for Developing a Tool for China," *Sustainability*, vol. 12, no. 16, p. 6501, Aug. 2020, doi: [10.3390/su12166501](https://doi.org/10.3390/su12166501).
- [23] E. Erlinawati and M. Muslimah, "Test Validity and Reliability in Learning Evaluation," *Bull. Community Engagem.*, vol. 1, no. 1, p. 26, Jan. 2021, doi: [10.51278/bce.v1i1.96](https://doi.org/10.51278/bce.v1i1.96).
- [24] Sudaryono, U. Rahardja, Q. Aini, Y. Isma Graha, and N. Lutfiani, "Validity of Test Instruments," *J. Phys. Conf. Ser.*, vol. 1364, no. 1, p. 012050, Dec. 2019, doi: [10.1088/1742-6596/1364/1/012050](https://doi.org/10.1088/1742-6596/1364/1/012050).
- [25] M. S. Malik and M. S. Priyadi, "Analysis Of Instrument Validity Assessment Items For Student Learning Achievement," *J. Ilm. Didakt. Media Ilm. Pendidik. dan Pengajaran*, vol. 21, no. 2, p. 127, Feb. 2021, doi: [10.22373/jid.v21i2.6702](https://doi.org/10.22373/jid.v21i2.6702).
- [26] S. Seno, M. Magito, and D. H. Perkasa, "The Influence of Career Development, Competence and Work Conflict on Job Satisfaction," *Talent J. Econ. Bus.*, vol. 1, no. 01, pp. 14–22, Dec. 2023, doi: [10.59422/jeb.v1i01.189](https://doi.org/10.59422/jeb.v1i01.189).
- [27] A. W. D. Putri and E. Setiyawati, "The Effect of Picture and Picture Interactive Model to Increase Student's Learning Motivation in Elementary Science Learning," *Edunesia J. Ilm. Pendidik.*, vol. 4, no. 3, pp. 1097–1108, Jun. 2023, doi: [10.51276/edu.v4i3.518](https://doi.org/10.51276/edu.v4i3.518).
- [28] Z. Dragičević and S. Bošnjak, "The Role Of Agile Software Architect In The Age Of Digital Disruption And Transformation," *Balk. J. Emerg. Trends Soc. Sci.*, vol. 3, no. 2, pp. 148–162, Dec. 2020, doi: [10.31410/Balkans.JETSS.2020.3.2.148-162](https://doi.org/10.31410/Balkans.JETSS.2020.3.2.148-162).
- [29] Kausar Parveen, "Rapid Digital Transformation Using Agile Methodologies for Software Development Projects," *Lahore Garrison Univ. Res. J. Comput. Sci. Inf. Technol.*, vol. 5, no. 3, pp. 54–64, Sep. 2021, doi: [10.54692/lgurjsit.2021.0503218](https://doi.org/10.54692/lgurjsit.2021.0503218).
- [30] A.-M. Gheorghe, I. D. Gheorghe, and I. L. IATAN, "Agile Software Development," *Inform. Econ.*, vol. 24, no. 2/2020, pp. 90–100, Jun. 2020, doi: [10.24818/issn14531305/24.2.2020.08](https://doi.org/10.24818/issn14531305/24.2.2020.08).
-