JTIF : Journal of Technology Information E-ISSN: 3063-0088
Volume 2 No. 1, March 2025 p 38 – 50

DEVELOPMENT OF A WASTE MANAGEMENT APPLICATION FOR RECYCLING IN BUMDESA JAMBUDIPA.

Asep Deden Rahmat[1], Widianjani[2], Syifa NAR[3]

[13]Informatics Engineering, Faculty of Engineering, Putra Indonesia University, Indonesia
[2]Informatics Management, Faculty of Engineering, Putra Indonesia University, Indonesia

Email: [1]asden.alc@gmail.com

Article Info

Article history:
Received Oct, 31, 2024
Revised
Accepted March, 30, 2025

Keywords:
Populations,
BUMDESA,
Management,
Programming.

ABSTRACT

Waste management in cities with large populations is a challenging issue, as residents produce waste daily, necessitating a specialized approach to address it, with recycling being one viable solution. In Jambudipa Village, the recycling process is managed by BUMDESA, where residents either bring their waste to sorting locations or have it collected by BUMDESA waste collectors. Waste transactions are manually recorded in a waste savings account. Given the high waste volume and large population, the recording of waste deposits needs to be more efficient and streamlined. This research aims to develop a waste management and savings application that not only tracks waste type and volume but also manages the proceeds from recycled waste, enabling residents to withdraw funds from BUMDESA's financial department at any time. The study employs the Extreme Programming system development method. The system is built using the CodeIgniter framework, with PHP as the programming language and MySQL for the database. The result of this research is a web-based application system that assists BUMDESA Jambudipa staff in computerizing waste exchange activities.

Corresponding Author:
Asep Deden Rahmat
Informatics Engineering Study Program, Faculty of Engineering
Universitas Putra Indonesia, Indonesia
Email: asden.alc@gmail.com

1. INTRODUCTION

In a community setting, waste-related issues have become increasingly difficult to solve due to the rising volume of waste, which grows in line with population increases in an area [1]. People frequently dispose of unused items or waste, and as the waste volume escalates, residents often discard it carelessly in their environment, creating additional problems [2]. Large amounts of waste thrown into ditches or drains can cause blockages and flooding, while excessive waste in temporary disposal sites leads to air pollution, foul odors, and the potential for disease outbreaks [3]. Consequently, waste management in urban areas, where residents lack land for waste disposal, is addressed by individuals, NGOs, and government agencies [4]. Local governments, however, are the most competent authorities for managing these waste challenges in larger cities [5].

The government has made various efforts to manage and address waste by providing several solutions, such as establishing temporary disposal sites and final disposal sites [6]. Additionally, according to the 2022 National Waste Management Information System data, national urban waste management in Indonesia recorded 20,500,305.03 tons per year, with a reduction rate of 17.26% per year, a handling rate of 48.72% per year, managed waste at 65.98% per year, and unmanaged waste at 34.02% per year [7]. Waste, therefore, is the byproduct of human activities, which holds no economic value if not effectively managed [8]. Proper waste management is essential to create economic value and benefit the surrounding community [9]. As stated in the Ministry of Environment Regulation No. 13 of 2012, a "waste bank" is a place for sorting and collecting waste that can be recycled or reused, thus holding economic value [10][11]. Typically, waste banks are managed by an organization within a community [12]. Waste banks operate by allowing customers to deposit waste and withdraw money earned from the waste collection [13][14]. Moreover, these waste banks maintain records of relevant waste data. From field observations, the author noted that in the Jambudipa village, Cianjur, specifically at the BUMDESA Jambudipa, there is currently no system in place to aid in waste management. Customer data recording, waste type categorization, transaction processes, and end-of-report generation are still

handled manually.

This approach is considered less accurate and can potentially disadvantage both the customers and BUMDES Jambudipa. Based on the explanation above, the author aims to develop a website focused on waste management and waste-for-groceries transactions for customers who participate in waste exchanges. The intended outcome of creating this website is to assist and streamline waste management processes, making them more efficient and user-friendly.

2. METHOD

2.1 Research Object

This research was conducted at the Village-Owned Enterprise (BUMDES) Jambudipa, located in Warungkondang District, Cianjur Regency, with the address Jl. Jambudipa RT. 007 RW. 002, Jambudipa Village, Warungkondang District, Cianjur Regency, West Java 4326. Cianjur Regency is bordered by West Bandung and Garut Regencies to the east, Bogor and Purwakarta Regencies to the north, Bandung Regency and the Indian Ocean to the south, and Sukabumi Regency to the west.

2.2 Research Method

In this research, a qualitative method with a descriptive approach is utilized.

2.3 Data Collection Method

The data collection methods for this study include: Observation, Interview, Documentation, and Literature Review.

2.4 System Design Methodology

In system design, the author employs the Extreme Programming (XP) method for system development. The following is an explanation of the Extreme Programming model used in this study [15]:

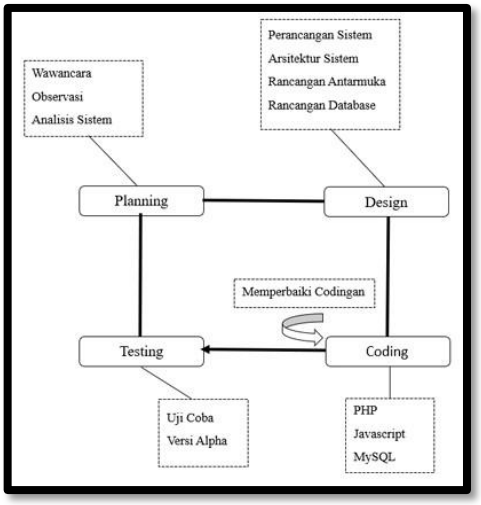

Figure 1. Extreme Programming Process Flow for the Application

The following outlines the process flow for the Recycling Waste Management Program application:

1. Planning: This stage begins with gathering requirements to help understand the context of the application. Here, the expected outputs, features, and functions of the application are defined. During the development of the Recycling Waste Management Program application, problems are identified, and user requirements for the system are analyzed.
2. Design: In this stage, system design is developed based on the analysis obtained. System architecture and database structures are built to illustrate the relationship between the database and the interface design, defining the layout of the developed system. The system design uses Unified Modeling Language (UML) and includes use case diagrams, activity diagrams, class diagrams, and sequence diagrams.
3. Coding: This is the implementation of the model into the user interface using programming languages. The programming languages used to build the Recycling Waste Management application include the CodeIgniter framework, which is a PHP framework, along with HTML, JavaScript, and CSS. MySQL is used for database management.
4. Testing: Only alpha testing is conducted by the researcher, designed after receiving interview results and using the black-box testing method. The goal of alpha testing is to review the design

2.5 System Requirements Specification

To realize the Recycling Waste Management application and ensure smooth development, the author categorizes the necessary system specifications into two main requirements:

a. Software Requirements: During the design and development of the Recycling Waste Management application, several software requirements must be provided, including: 1. XAMPP, 2. PHP CodeIgniter, 3. MySQL, 4. Sublime Text, 5. Google Chrome.
b. Hardware Requirements: To achieve the best results in the design process, the author uses a computer or laptop with specifications that meet the requirements of this application system, which include: 1. Processor: Core i3/i5, 2. RAM: 4 GB, 3. Laptop/PC.

3. RESULTS AND DISCUSSION

The results and discussion at this stage provide an explanation of the outcomes of the research implementation in accordance with the development methods followed in this study. This research focuses on the implementation of the research method or the planned application development.

3.1 System Analysis and Design

The development of the Waste Management Application aims to assist in managing waste data and storing data for customers and staff of the Village-Owned Enterprises (BUMDES), thereby preventing data discrepancies between customers and BUMDES staff. To design this application, a systematic approach is necessary, utilizing a system design method to analyze the system being developed.

In this study, the author employs the Extreme Programming (XP) method for the requirements of system application development. The design phase of the application requires a planning stage to analyze the system's needs and establish a timeline for completion. The subsequent stage involves designing the system workflow and the user interface. Following this, the coding phase entails writing the application code, while the final stage involves testing, where black-box testing is conducted to assess the application's distribution to users and gather feedback regarding the application created.

3.1.1 Planning

In the planning stage, this is the initial phase for analyzing the requirements needed for the system, establishing a timeline for system development, and estimating the project completion duration.

a. User Story

The user story phase involves analyzing requirements to understand the needs for designing the system. From the field research, the author collected the following data regarding the issues faced:

1. There is currently no application available to assist BUMDES staff in the waste data deposit program.
2. A reporting system for waste deposits for BUMDES staff is not yet available.
3. There is a risk of data discrepancies between customers and BUMDES staff because individual storage is not centralized, which can lead to losses for either the customers or the BUMDES staff.

Based on these issues, the researcher outlines an overview of the application being developed as follows:

1. The creation of a supportive system for the waste data deposit application.
2. The development of a supportive application system with features for waste data deposits to facilitate reporting needs.
3. The waste management application allows for the alignment of data between customers and BUMDES staff, preventing losses for either party.
4. The system acts as an intermediary for waste data deposits. The system is designed with two user roles to ensure it meets expectations. The two user roles are as follows:
   - Admin: The admin can access and manage all data within the system.
   - Customer: Customers can only access specific pages within the system.

b. Functional Requirements: The functional requirements needed for the system are as follows:

1. General Requirements: Both the admin and customers must log in using their email and password to access the main page.
2. Admin: a. Add, edit, delete, and update master data within the system. b. Add, edit, and delete customer accounts. c. Edit the admin profile. d. Conduct deposit and withdrawal transactions for customers. e. Manage waste data deposits.
3. Customer: a. Edit the customer profile. b. View deposit transaction history. c. View withdrawal transaction history.

c. Non-Functional Requirements: This analysis describes the hardware and software requirements necessary to operate the designed system. The specifications required for running the Waste Management Application are as follows:

1. Hardware: The hardware analysis includes non-functional requirements to support the system's efficient operation. The hardware used to run the Waste Management Application includes a laptop or PC with the following minimum specifications: Intel® Celeron® CPU N3350 @ 1.10 GHz and an internal storage RAM of 2 GB.
2. Software: This application requires supporting software to function effectively when implemented. The user specifications needed are as follows:
   - Users should understand how to operate a laptop or PC.
   - Users should be familiar with using websites to access the Waste Management Application.
3. User-Friendliness: Admins and customers will easily understand the available features, as the system is designed with a simple interface. The interface of this system is created using Bootstrap templates to provide a modern and responsive appearance.
4. Accessibility: The implementation of this system includes different user access rights for admins and customers. The programming languages used in this system are JavaScript and PHP, with a MySQL database to ensure data connectivity across different systems, utilizing the CodeIgniter web development framework.

3.2 Design

The results of the functional requirements analysis were then designed using UML. In addition to being designed with UML, the database design and interface design were also created. The following are the designs made in this research:

3.2.1 Use Case Diagram

The use case diagram illustrates the interactions that occur between actors and the system from the user's perspective. Below is the use case diagram for the Waste Management Application:

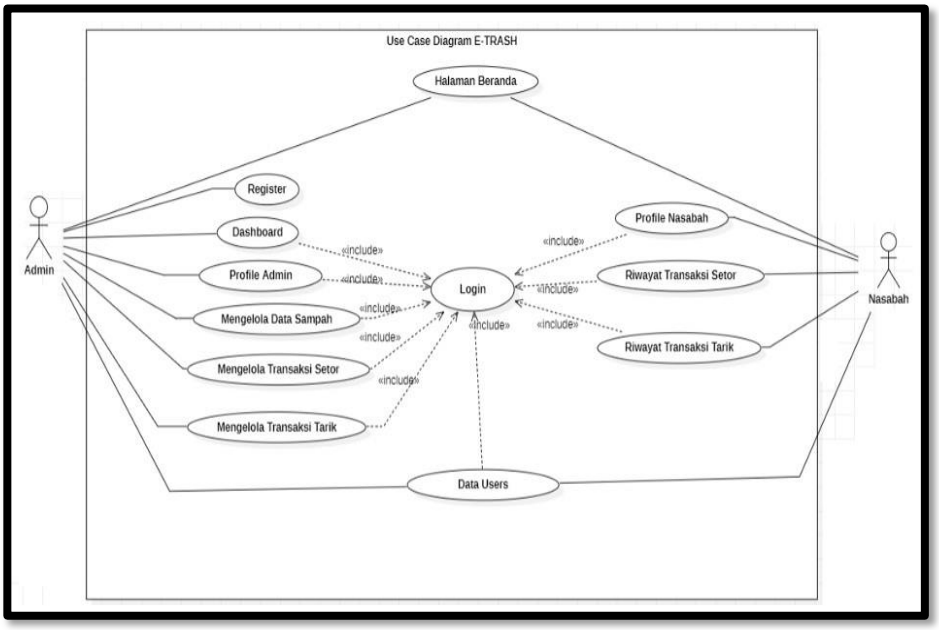

Figure 2. Use Case Diagram of the Application.

The use case diagram above illustrates that the admin is the staff of BUMDES Jambudipa responsible for managing the Recycling Waste Management Application, while the customers are the residents of Jambudipa. Users can utilize the application and enjoy its features after logging in. Before logging in, both users first access the homepage. To access the login, the admin must register or sign up first, while customers can register through the admin, who will then register them. After completing the registration process, users log into the application by entering the email and password they have registered as either admin or customer. Once logged in, the admin can access the dashboard feature, admin profile page, waste data page, deposit transaction page, withdrawal transaction page, and user data page. Meanwhile, customers can access their profile page, deposit transaction history page, and withdrawal transaction history page.

3.2.2 Activity Diagram

The activity diagram is utilized to model the processes occurring within the application, explaining the activities associated with each function. Below are the activity diagrams for the Recycling Waste Management Program:

1. Activity Diagram for Main Menu.
2. Activity Diagram for Admin (Activity Diagram for Registration, Activity Diagram for Admin Login, Activity Diagram for Editing Admin Profile, Activity Diagram for Adding Waste Data, Activity Diagram for Editing Waste Data, Activity Diagram for Deleting Waste Data, Activity Diagram for Adding Deposit Transaction, Activity Diagram for Deleting Deposit Transaction, Activity Diagram for Adding Withdrawal Transaction, Activity Diagram for Deleting Withdrawal Transaction, Activity Diagram for Adding User Data, Activity Diagram for Editing User Data, Activity Diagram for Deleting User Data).
3. Activity Diagram for Customers (Activity Diagram for Customer Login, Activity Diagram for Editing Customer Profile, Activity Diagram for Deposit Transaction History, Activity Diagram for Withdrawal Transaction History).

These activity diagrams provide a clear and structured representation of the workflows and interactions within the Recycling Waste Management Application, aiding in the understanding and development of its functionalities.

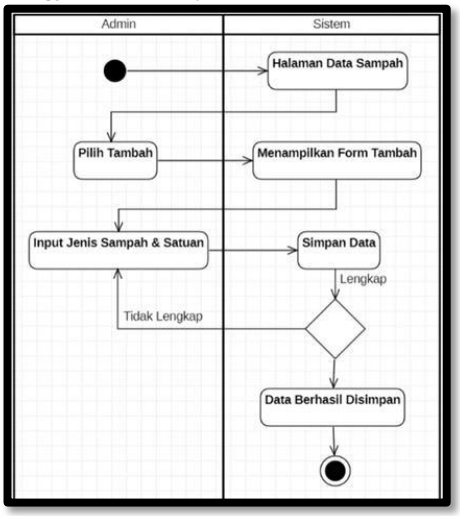

Figure 3. Activity Diagram for Adding Waste Data

In Figure 3, the admin manages waste data, allowing new waste data to be added when necessary.

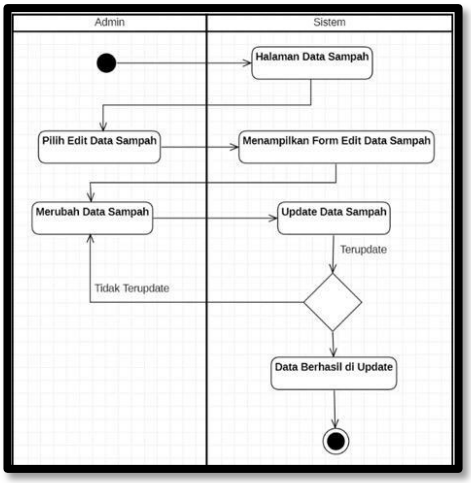

Figure 4. Activity Diagram for Editing Waste Data

In Figure 4, the admin manages waste data, allowing edits to be made if any waste data is inaccurate.

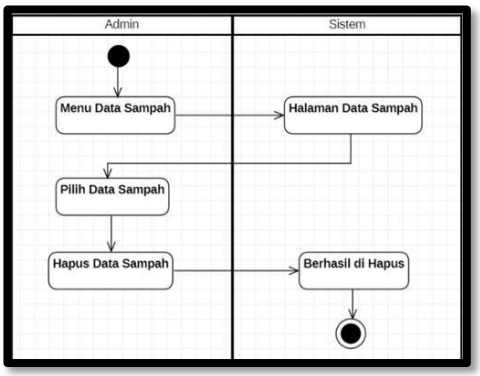

Figure 5. Activity Diagram for Deleting Waste Data

In Figure 5, the admin manages waste data, allowing deletion of any waste data that is inaccurate or unnecessary.

3.2.3 Class Diagram

A class diagram represents the structure of the system by defining the classes that are created to build the system. Each class in the system structure must be able to perform functions that meet the system's requirements. Below is the class diagram design for the Waste Management Application.

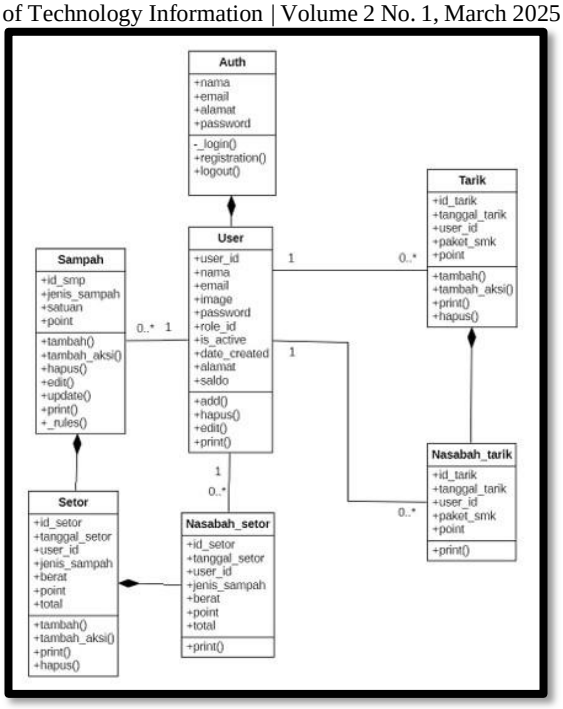

Figure 6. Class Diagram of the Application

In the class diagram above, there are 7 classes: *auth*, *user*, *sampah* (waste), *setor* (deposit), *tarik* (withdrawal), *nasabah_setor* (user deposit), and *nasabah_tarik* (user withdrawal). The following is an explanation of each class in the Waste Management Application class diagram:

1. The *auth* class is connected to the *user* class, as *user* requires attributes from *auth*. If the *user* class does not exist, then *auth* cannot function, as the attributes in *user* are linked to *auth*.
2. The *user* class, aside from being connected to *auth*, is also connected to *sampah*, *setor*, *tarik*, *nasabah_setor*, and *nasabah_tarik*. There are two types of users in the *user* class: *admin* and *nasabah* (customer). This class can be managed by *admin*, who can add, update, and delete data within this class.
3. The *sampah* class is connected to *user* and can be accessed by *admin*. *Admin* can manage the data in *sampah*, including adding, updating, and deleting information.
4. The *setor* class is linked to both the *user* and *sampah* classes and can be accessed by *admin*. *Admin* can manage data in *setor*, including adding and deleting records.
5. The *tarik* class is connected to the *user* class and can be accessed by *admin*, who can manage the data within *tarik*, including adding and deleting records.
6. The *nasabah_setor* class is connected to *user* and *setor*, and it is accessible by *user* (customer). The *user* (customer) can view detailed information within *nasabah_setor*, which is linked to *setor*.
7. The *nasabah_tarik* class is connected to *user* and *tarik* and can be accessed by *user* (customer). The *user* (customer) can view detailed information within *nasabah_tarik*, which is linked to *tarik*.

3.2.4 Sequence Diagram

A sequence diagram illustrates the dynamic collaboration among various objects. This diagram is useful for showing the sequence of messages exchanged between objects and the interactions among objects within the application. Below is the sequence diagram design for the Waste Management Application:

1. Sequence Diagram for the Homepage
2. Sequence Diagram for Admin: Sequence Diagram for Register, Sequence Diagram for Admin Login, Sequence Diagram for Edit Profile, Sequence Diagram for Add Waste Data, Sequence Diagram for Edit Waste Data, Sequence Diagram for Delete Waste Data, Sequence Diagram for Add Deposit Transaction, Sequence Diagram for Delete Deposit Transaction, Sequence Diagram for Add Withdrawal Transaction, Sequence Diagram for Delete Withdrawal Transaction, Sequence Diagram for Add User Data, Sequence Diagram for Edit User Data, Sequence Diagram for Delete User Data.
3. Sequence Diagram for Customer: Sequence Diagram for Login, Sequence Diagram for Edit Customer Profile, Sequence Diagram for Deposit Transaction History, and Sequence Diagram for Withdrawal Transaction History.

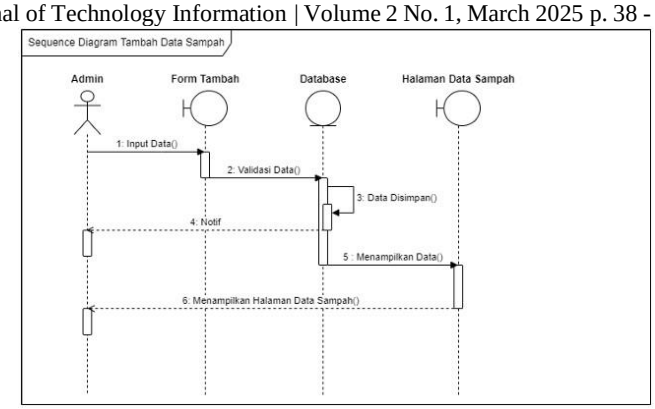

Figure 7. Sequence Diagram for Adding Waste Data

The sequence diagram above illustrates how the admin inputs data for adding waste information. The admin enters waste data, and the system performs validation. Once the data is saved to the database, the system sends a notification to the admin, then displays the newly entered data and updates the waste data page.

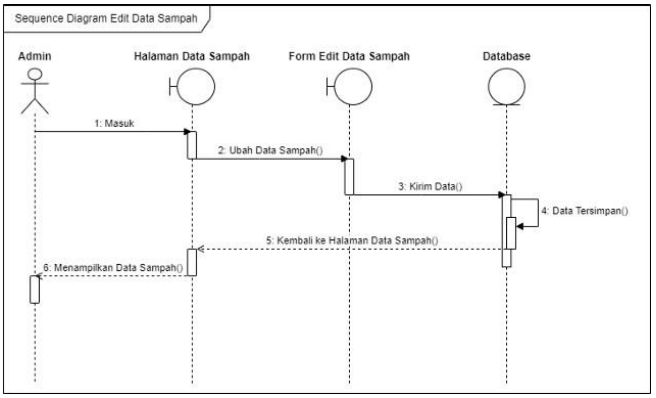

Figure 8. Sequence Diagram for Editing Waste Data

The sequence diagram above illustrates how the admin edits waste data. The admin accesses the waste data page, selects the waste data to be edited, and makes the necessary changes. After editing, the data is sent and saved in the database. Once saved, the system displays the updated waste data page.

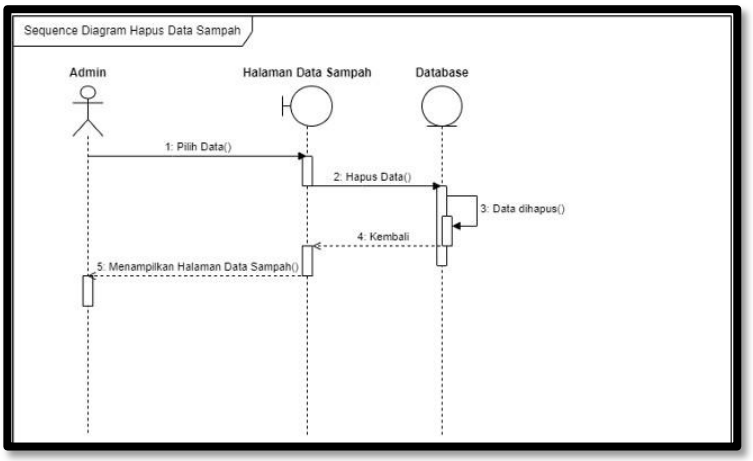

Figure 9. Sequence Diagram for Deleting Waste Data

The sequence diagram above illustrates how the admin deletes waste data. The admin accesses the waste data page, selects the data to be deleted, and confirms the deletion. After the data is deleted, the system refreshes and displays the updated waste data page.

3.2.5 Application System Architecture

The application system architecture outlines the overall design of the application system along with its supporting infrastructure. Below is a representation of the system architecture for the Waste Management Application.

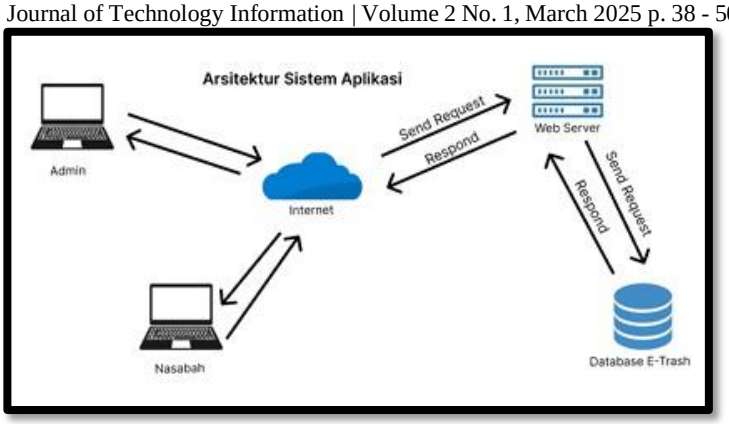

Figure 10. Application System Architecture

The application system architecture diagram above illustrates how the Waste Management Application operates. Both the admin and the users can access the application via the internet, which connects to the web server. The web server retrieves data from the database. The database responds by sending the data contained within it back to the web server, which then sends the response to both the admin and the users, enabling them to access the application.

3.2.6 Database Design

The database design provides an overview of the structure of the data storage medium. The database design for the Waste Management Application includes the following tables: User Table, Waste Table, Points Table, Deposit Table, Groceries Table, and Withdrawal Table.

Table 1. Waste Table

Field	Type Data	Length	Keterangan
Id_smp	Integer	11	Primary Key dan Auto Increment
Jenis_sampah	Varchar	255	Unik
Satuan	Varchar	255	
Point	Integer	11	

3.2.7 Interface Design

In system development, an interface design is essential to facilitate communication between users and the system. The interface provides the layout of the system's pages, which is used for inputting data and producing output that meets user requirements. Below are the interface designs for the Waste Management Application: Homepage, Admin Registration Page, User Login Page (Admin and Customers), Dashboard Page, Admin Profile Page, Edit Admin Profile Page, Waste Data Page, Add Waste Data Page, Edit Waste Data Page, Deposit Transaction Page, Add Deposit Transaction Page, Withdraw Transaction Page, Add Withdraw Transaction Page, User Data Page, Add User Data Page, Edit User Data Page, Customer Profile Page, Edit Customer Profile Page, Deposit Transaction History Page, Withdraw Transaction History Page.

Figure 11. Homepage

Figure 11 shows the initial page when the user opens the waste recycling application implemented by BUMDES Jambudipa.

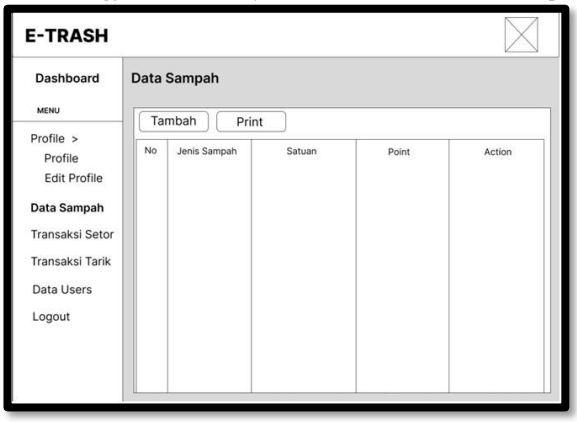

Figure 12. Waste Data Page

Figure 12 displays the waste data page that shows the waste that has been recycled at BUMDES Jambudipa.

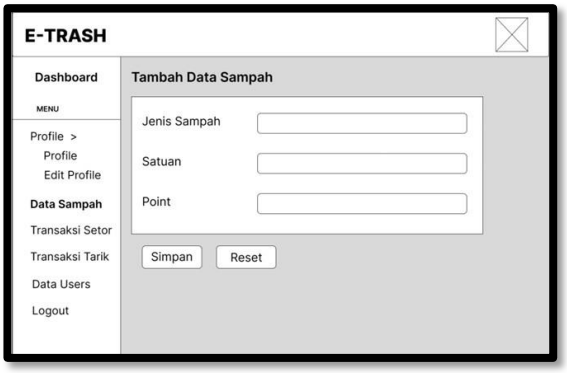

Figure 13. Add Waste Data Page

Figure 13 shows the add waste data page that is used by the operator at BUMDES Jambudipa.

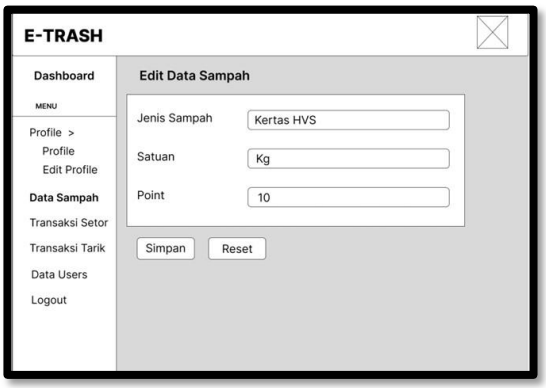

Figure 14. Edit Waste Data Page

Figure 14 illustrates the edit waste data page that is used by the recycling customers of BUMDES Jambudipa.

3.3 Coding

Coding is a stage of translating the design results of the Waste Management Application into sequences of program code based on the use case, activity diagram, and interface design. In this study, the Waste Management Application is implemented using PHP programming language to create the website and MySQL as the database or data storage medium. The researcher also utilizes several additional components, such as CSS for styling various elements to ensure the website is well-structured and uniform. Additionally, the author employs the Bootstrap framework to enhance the visual appeal of the website while utilizing MySQL as the database.

3.4 Testing

Testing is the final stage of this research, and the system testing is conducted in three phases: alpha version, beta version, and release version one. Below are the testing stages of the Waste Management Application. The alpha version testing is performed directly by the researcher, and the alpha testing itself is implemented using the black-box testing method, which involves evaluating the software from the perspective of functional specifications without considering the design and source code. Black-box testing is solely aimed at determining whether the input and output functions of the developed software meet the required specifications.

3.5 System Implementation

In this stage, the author attempts to implement the hardware and software necessary to run the designed application, including both supporting applications and specialized software. The steps for implementation are as follows:

3.5.1 Hardware Implementation

To run this application effectively, hardware is required to ensure that the designed system can be implemented. In this study, the hardware specifications used are as follows: Laptop Model: Asus VivoBook X415JA (Operating System: Windows 11 Home Single Language 64-bit, Processor: Intel(R) Core(TM) i3-1005G1, CPU Speed: @ 1.20GHz dan Memory: 12GB RAM).

3.5.2 Software Implementation

Software implementation is a crucial aspect of application development. The software tools used in this implementation are: Sublime Text, CodeIgniter 3, Bootstrap, XAMPP, dan MySQL.

3.5.3 User Interface Implementation

The implementation of the designed interface, in addition to software and hardware, includes the user interface itself. The implemented interface has been divided into several pages or forms. Below are the interface displays of the developed system: Homepage, Admin Registration Page, User Login Page (Admin and Customers), Dashboard Page, Admin Profile Page, Edit Admin Profile Page, Waste Data Page, Add Waste Data Page, Edit Waste Data Page, Deposit Transaction Page, Add Deposit Transaction Page, Withdrawal Transaction Page, Add Withdrawal Transaction Page, Users Page, Add User Data Page, Edit User Data Page, Customer Profile Page, Edit Customer Profile Page, Deposit Transaction History Page and Withdrawal Transaction History Page.

3.6 System Testing

Testing a system is a crucial part of the application development cycle. The aim is to ensure that the developed application maintains high quality and meets the specified requirements effectively.

3.6.1 Alpha Version Testing

Alpha version testing is conducted directly by the researcher using the black-box testing method. Below are the results from the alpha testing of the Recycling Waste Management Application:

Table 2. Admin Page Testing Results

Testing Items	Testing Scenario	Expected Results	Testing Results of the Application	Testing Status
Register Menu	If the name, email, address, and password fields are not filled in, click the "Register" button.	The system denies access and displays the message: "Name, email, address, and password fields are empty. Please fill them in".	The system denies access if the name, email, address, and password fields are not filled in and displays the message: "Name, email, address, and password fields are empty. Please fill them in".	Success
	Name is not filled in	The system denies access and displays 'Name is empty, please fill it in.	The system denies access if the name is not filled in and displays the message 'Name is empty, please fill it in.	Success
	Email not filled in	The system denies access and displays a message saying 'Email is empty, please fill it in'	The system denies access if the email is not filled in and displays a message saying 'Email is empty, please fill it in'	Success
	Address is not filled in	The system denies access and displays 'Address is empty, please fill it in	The system denies access if the address is not filled in and displays the message 'Address is empty, please fill it in	Success
	Password is not filled in	The system denies access and displays the message: 'Password is empty, please fill it in'	The system denies access if the password is not filled in and displays the message: 'Password is empty, please fill it in'	Success
	Entering name, email, address, and password has been completed	The account is already registered; the system redirects to the login menu & displays 'Account already registered'	The account has been registered and will redirect to the login menu, then the system displays the message 'Account already registered'	Success
Login Menu	Email and password are not filled in; click the login button	The system denies access and displays the message: 'Email and password are empty, please fill them in'	The system denies access if the email and password are not filled in and displays the message: 'Email and password are empty, please fill them in'	Success

	Email is not filled in	The system denies access and displays the message: 'Email is empty, please fill it in'	The system denies access if the email is not filled in and displays the message: 'Email is empty, please fill it in'	Success
	Password is not filled in	The system denies access and displays 'Password is empty, please fill it in'	The system denies access if the password is not filled in and displays the message 'Password is empty, please fill it in'	Success
	Entering an email that has not been registered	The system denies access and displays the message that the email is not registered	The system denies access and displays the message that the email is not registered	Success
	Entering the wrong password	The system denies access and displays a message indicating the password is incorrect	The system denies access and displays the message 'incorrect password'	Success
	Entering the correct email and password	The system redirects to the main page	The system navigates to the main page and displays the main page	Success
Admin Profile Menu	Edit Profile	The system edits the profile data and displays 'Profile successfully edited'	admin dapat mengedit data profile & sistem menampilkan pesan profile berhasil di edit.	Success
	Detail Profile	The system displays the profile details	The admin accesses the profile, and the system displays the profile details	Success
Menu Waste Data	Add Waste Data	The system adds the waste data, saves it, and then displays a message indicating that the data has been successfully added	The admin adds waste data, and the system saves the data, then displays the message 'Data successfully added'	Success
	Edit Waste Data	The system edits the waste data and displays 'Data successfully edited	The admin edits the waste data, and the system will display the message 'Data successfully edited'	Success
	Delete Waste Data	The system displays the waste data and will delete the waste data	The system displays the waste data page, and the admin deletes the waste data; the system removes the waste data	Success
	Print Waste Data	The system grants access to print the waste data	The admin receives access from the system to print the waste data	Success
Deposit Transaction Menu	Add Deposit Transaction	The system adds the deposit transaction and saves it, then displays 'Successfully completed the deposit transaction'	The admin adds a customer's deposit transaction, and the system saves the deposit transaction, then displays the message 'Successfully completed the deposit transaction'	Success
	Delete Deposit Transaction	The system will display the deposit transactions and allow the user to delete a deposit transaction	Sistem menampilkan transaksi setor dan sistem dapat menghapus transaksi setor.	Success
	Print Deposit Transaction	The system grants access to print the deposit transactions	The admin receives access from the system to print the deposit transactions	Success
Withdrawal Transaction Menu	Add Withdrawal Transaction	The system adds the withdrawal transaction and saves it, then displays 'Successfully completed the withdrawal transaction'	The admin adds the customer's withdrawal transaction, and the system saves the withdrawal transaction, then displays the message 'Successfully completed the withdrawal transaction'	Success
	Delete Withdrawal Transaction	The system displays the withdrawal transaction and will delete the withdrawal transaction.	The system displays the withdrawal transaction, and it can delete the withdrawal transaction	Success
	Print Withdrawal	The system grants access	The admin receives access from the	Success

	Transaction	to print the withdrawal transaction	system to print the withdrawal transaction	
	Add User Data	The system adds user data and saves the user data, then displays, 'Your account has been successfully registered'	The admin adds user data, and the system will save the user data, then the system displays the message, 'Your account has been successfully registered'	Success
User Data Menu	Edit User Data	The system will edit the user data and display 'Data successfully edited'	The admin edits user data, and the system will display the message 'Data successfully edited'	Success
	Delete User Data	The system displays the user data and will delete the user data	The system displays user data, and the system can delete user data	Success
	Print User Data	The system grants access to print user data	The admin receives access from the system to print user data	Success

Table 3. Test Results for the Customer Page

Testing Items	Testing Scenario	Expected Results	Testing Results of the Application	Testing Status
Login Menu	Email and password are not filled in; click the login button	The system will deny access and display the message: 'Email and password are empty, please fill them in	The system denies access if the email and password are not filled in, and displays the message, 'Email and password are empty, please fill them in'	Success
	Email is not filled in	The system will deny access and display the message: 'Email is empty, please fill it in'	The system denies access if the email is not filled in and displays the message, 'Email is empty, please fill it in'	Success
	Password is not filled in	The system will deny access and display the message: 'Password is empty, please fill it in'	The system denies access if the password is not filled in and displays the message, 'Password is empty, please fill it in'	Success
	Entering an email that has not been registered	The system will deny access and display the message: 'Email is not registered'	The system denies access and displays the message, 'Email is not registered'	Success
	Entering an incorrect password	The system will deny access and display the message: 'Incorrect password'	The system denies access and displays the message, 'Incorrect password'	Success
	Entering the correct email and password	The system will redirect to the main page	The system redirects to the homepage and displays the homepage	Success
Customer Profile Menu	Edit Profile	The system will edit the profile data and display 'Profile successfully edited'	The user can edit their profile data, and the system will display the message 'Profile successfully edited'	Success
	Detail Profile	The system will display the profile data details	The user can access their profile, and the system displays the profile details	Success
Deposit Transaction History Menu	Deposit Transaction History Details	The system will display the details of the deposit transaction history	The user accesses the deposit transaction history menu, and the system displays the deposit transaction history	Success
	Print Deposit Transaction History	The system grants access to print the deposit transaction history	The user receives access from the system to print the deposit transaction history	Success
Withdrawal Transaction History Menu	Withdrawal Transaction History Details	The system will display the details of the withdrawal transaction history	The user accesses the withdrawal transaction history menu, and the system displays the withdrawal transaction history	Success
	Print Withdrawal Transaction History	The system grants access to print the withdrawal transaction history data	The user receives access from the system to print the withdrawal transaction history	Success

4. DISCUSSION

To further enhance the quality of the research on the development of the recycling waste deposit website and to increase the benefits of this application for BUMDES Jambudipa, this research has not yet progressed to the final processing stage, which includes the savings from the waste deposits. Additionally, the financial management of the proceeds from the sales can be accessed by residents who deposit waste, or it can be used for other purposes, such as paying for PDAM, electricity, taxes, and so on.

5. CONCLUSION

After conducting research on the waste recycling management application program, based on the results of the analysis, design, and system implementation, the author draws the following conclusions: 1. The Waste Recycling Management Application Program can assist the staff of BUMDESA Jambudipa in recording waste exchange activities in a computerized manner. 2. The Waste Recycling Management Application Program can help store data related to waste deposits and withdrawals during waste exchange activities. 3. The Waste Recycling Management Application Program can synchronize the data from deposits and withdrawals between the admin and the customers, preventing data discrepancies.

REFERENCES

[1] Zhechen Zhang, et.al. (2024). *Municipal solid waste management challenges in developing regions: A comprehensive review and future perspectives for Asia and Africa*. Science of The Total Environment, Volume 930, 172794, ISSN 0048-9697, https://doi.org/10.1016/j.scitotenv.2024.172794.

[2] Abubakar IR. et.al (2022). *Environmental Sustainability Impacts of Solid Waste Management Practices in the Global South*. Int J Environ Res Public Health. 2022 Oct 5;19(19):12717. doi: 10.3390/ijerph191912717. PMID: 36232017; PMCID: PMC9566108.

[3] Jakhar R, Samek L, Styszko K. (2023). *A Comprehensive Study of the Impact of Waste Fires on the Environment and Health*. Sustainability. 15(19):14241. https://doi.org/10.3390/su151914241.

[4] Anh Thu Nguyen, Nhan Nguyen, Phuong Phung, Nguyễn Yến-Khanh. (2023). *Residents' waste management practices in a developing country: A social practice theory analysis*, Environmental Challenges,Volume 13, 100770, ISSN 2667-0100, https://doi.org/10.1016/j.envc.2023.100770.

[5] Ashish Soni, et.al. (2022). *Challenges and opportunities of utilizing municipal solid waste as alternative building materials for sustainable development goals: A review*, Sustainable Chemistry and Pharmacy, Volume 27, 2022, 100706, ISSN 2352-5541, https://doi.org/10.1016/j.scp.2022.100706.

[6] Pheakdey DV, Quan NV, Khanh TD, Xuan TD. (2022). *Challenges and Priorities of Municipal Solid Waste Management in Cambodia*. International Journal of Environmental Research and Public Health. 2022; 19(14):8458. https://doi.org/10.3390/ijerph19148458

[7] Naztia Haryanti (2023). *Indonesia is the largest contributor of food waste in ASEAN*. https://infid.org/en/indonesia-penyumbang-sampah-makanan-terbanyak-se-asean/

[8] Niyitanga Evode, et.al. (2021). *Plastic waste and its management strategies for environmental sustainability*, Case Studies in Chemical and Environmental Engineering, Volume 4, 2021, 100142, ISSN 2666-0164, https://doi.org/10.1016/j.cscee.2021.100142.

[9] Younis Ahmad Hajam. et.al (2023). *Environmental waste management strategies and vermi transformation for sustainable development*, Environmental Challenges, Volume 13, 2023, 100747, ISSN 2667-0100, https://doi.org/10.1016/j.envc.2023.100747.

[10] I Putu Dody Lesmana and Beni Widiawan. (2019). *Design Of A Waste Management Model Using Integrated Organic And Solid Waste Management: A Case Of Gambirono SubDistrict, Jember*. The Second International Conference on Food and Agriculture. pp 211-217. ISBN : 978-602-14917-9-9.

[11] Kappy, T, M., Maimina, S., Latue, P, C., Rakuasa, H. (2023). *The Role of Beta Bank Sampah Community in Plastic Waste Management in Hative Besar Village, Ambon City*. Sinergi International Journal of Economics, 1(1), 86-99.

[12] Budiyarto A, Clarke B, Ross K. *Overview of waste bank application in Indonesian regencies*. Waste Management & Research. 2024;0(0). doi:10.1177/0734242X241242697

[13] Ernoiz Antriyandarti et.al (2024). *The Development of Waste Bank Management to Improve Household Income in Surakarta City*. Jurnal Teknologi Lingkungan. Vol. 25 No.2 (Juli 2024) 231-237. e-ISSN 2548-6101 p-ISSN 1411-318X. Journal Homepage: ejournal.brin.go.id/JTL

[14] Lubis, Ratna. (2019). *Managing Ecopreneurship: The Waste Bank Way With Bank Sampah Bersinar (Bsb) In Bandung City*, INDONESIA.

[15] Sari, Purma. (2018). *Development of Information Systems Using Extreme Programming Method*. Jurnal INFORM. 3. 95. 10.25139/inform.v3i2.968.