

i-Care: A Web-Based Online Goods Donation Management System

Nur Izzah Batrisyia Mohd Azhar ¹, Mazidah Mat Rejab ^{1*}

¹Fakulti Sains Komputer dan Teknologi Maklumat,
Universiti Tun Hussein Onn Malaysia, Parit Raja, Batu Pahat, 86400, MALAYSIA

DOI: <https://doi.org/10.30880/aitcs.2023.04.02.091>

Received 23 June 2023; Accepted 29 October 2023; Available online 30 November 2023

Abstract: Emergencies are described as urgent situations that have imminently threatened lives or livelihoods. For example, flooding. Hence, many people donate their useful goods with the aid of NGOs to help victims. However, dumping goods into these devastated areas has stymied relief efforts. Supplies needed by victims are often insufficient. Additionally, NGOs manual data management process is time and energy consuming. Therefore, a system entitled "I-Care: A Web-Based Online Goods Donation Management System" is to be developed. The system offers a structured management platform with a focus on emergency relief assistance. It is designed using an object-oriented approach and a System Prototyping Model as its methodology. It is expected to allow donors to donate their goods according to the goods category set by the administrator. Meanwhile, administrators are allowed to update the goods specifications needed by victims. Furthermore, it simplifies the NGO's task of recording the whole donation activity and processes.

Keywords: Donation Management System, Goods Donation, Emergency Relief Effort

1. Introduction

Donation is something that a person gives with the intention of alleviating the burden faced by the needy. Donations may take various forms, including goods such as clothing and hygiene essentials. Goods donation from public are important specially to help those who are in an emergency. Emergencies are described as urgent situations in which there is clear evidence that the event causes human suffering. For example, flooding and land sliding. Due to these devastating situations, many people donate their useful goods with the aid of the responsible NGOs. However, dumping donated goods into these devastated areas has stymied relief efforts. Furthermore, supplies needed by victims are insufficient. It is because donors often provided the same type of aid. Additionally, the manual data management process used by NGOs is time and energy consuming.

Therefore, a system entitled "I-Care: A Web-Based Online Goods Donation Management System" is to be developed. The system is a website that offers a structured management platform with a focus on emergency relief assistance. The system is expected to allow donors to donate their goods according to the goods category set by the administrator. Moreover, the system is expected to allow administrator

*Corresponding author: mazidah@uthm.edu.my

to update the goods specifications needed by victims through the system. As a result, the system simplifies the NGO's task of recording the whole donation activity and processes. The objectives of the project are to study the existing online donation management system, develop the I-Care System using web-based technology and test the functions of the I-Care System using alpha-testing. Lastly, the project's scope is focused on the administrator and donor. There are a total of eight modules exist in the system: Register Module, Login Module, Manage Emergency Relief Program Module, Manage Donation Activity Module, Manage Donation Category Module, View Donation Report Module, Make Goods Donation Module and View Donation History Module.

The rest of the paper is organized as follows: Section 2 discusses the literature review of the related work and existing applications. Next, the methodology used to develop the system including the analysis and design is described in Section 3. Section 4 discusses the results and discussion of the system's design, implementation, and testing. Finally, the last section concludes the current work and highlights the future work to be performed to enhance the i-Care system.

2. Related Work

2.1 Goods Donation

The form of contribution known as "goods donation" includes practical goods. For instance, clothing, books, toys, and hygiene essentials [1]. Even if the goods are used, if it is in excellent condition and useful to the recipient, it can be donated [2]. Although some donors may want to give everything they own, there are a few restrictions on what charities allow them to donate. These restrictions usually are set based on the emergency, the target receiver, and the type of aid they need. For example, flood relief aid programs, NGO accept goods such as diapers, blankets, and hygiene kits. However, they rarely accept porcelain kitchenware for the program due to its fragility and difficulty to distribute.

2.2 Donation Management System

A donation management system assists NGOs in collecting, tracking, and distributing donations, as well as maintaining donor data and interacting with them during any charity program. By using the system, it helps administrators save time by automating donation management tasks, such as approving donors' registration and donation. They do not have to go through loads of paperwork to approve the registration one-by-one. Other than that, it has better data collecting and analysis than the manual system.

2.3 Comparison with the Existing Systems

Based on comparisons made between the existing systems, there are certain similarities and differences in the features of these systems. The study's finding is used as a guide for developing an online goods donation management system. The finding from the comparison of the system's features is shown in Table 1.

Table 1: System's Comparison

Features/System	Salvation Army Malaysia	Global Giving	Yayasan Ikram Malaysia	i-Care
Login Module	√	√	√	√
Register Module	√	X	√	√
Online Transaction	X	√	√	X
Donation Module Make Goods	X	X	X	√
Donation Module View Donation	√	√	X	√
History Module Manage Emergency	√	√	√	√
Relief Program Module				
Manage Donation Activity Module	√	√	√	√
Manage Donation Category Module	√	√	√	√
View Donation Report Module	√	√	√	√

Based on Table 1, all systems feature a Register Module that allows first-time users to create an account. Thus, permitting them to access the system with a valid log-in. Only the Global Giving and Yayasan Ikram Malaysia has an Online Transaction Donation Module. Meanwhile, Salvation Army Malaysia only provides bank account details in the project description. However, the domain of the i-Care system is limited to donating goods for emergency relief effort. Thus, the Online Transaction Donation Module is disabled. Furthermore, only the i-Care system offers a Make Goods Donation Module. The module enables users to key-in their donation details before delivering it to the designated drop-off location. Thus, the NGO could control the amount of donation needed while also keeping track of the donation status.

3. Methodology

A software process model is a method in software engineering that divides software development work into discrete phases to better design, product management, and project management. It is often referred to as the software development life cycle [3].

3.1 Prototyping Model

The I-Care system is developed using system prototyping methodology. Through this methodology, the analysis, design, and implementation phases are performed concurrently (see Figure 1).

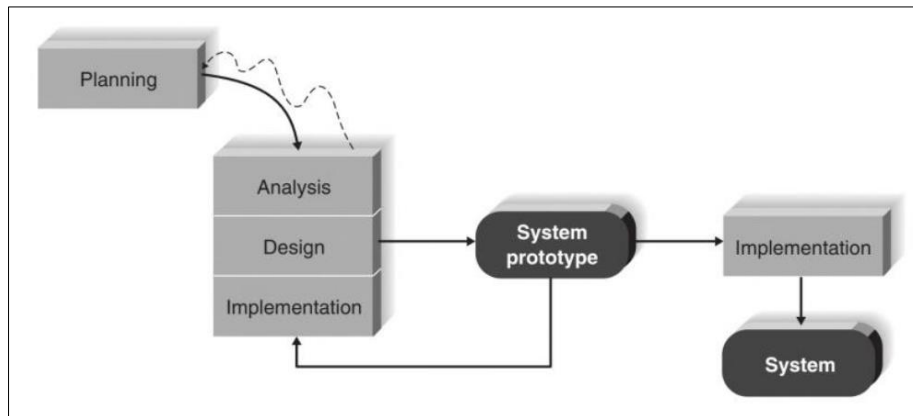


Figure 1: System Prototyping Model [4].

There are a total of five Software Development Life Cycle (SDLC) phases in the prototyping model. Each phase has a specific task that must be carried out throughout the entire project development, as indicated in Table 2.

Table 2: Tasks for each SDLC phases

Phases	Tasks
Planning	<ul style="list-style-type: none"> • Perform feasibility study • Prepare a proposal • Create Work Breakdown Structure. • Defense the proposal title
Analysis	<ul style="list-style-type: none"> • Study existing system and its problem. • Accumulate data for the project through questionnaire. • Specify functional and non-functional requirements. • Develop use case, sequence, activity, and class diagrams.
Design	<ul style="list-style-type: none"> • Design system’s interface and database
Implementation	<ul style="list-style-type: none"> • Program the system using PHP • Test the system’s functionality.

4. Results and Discussion

The outcomes of the study's analysis are shown utilizing several types of UML diagram.

4.1 Functional and Non-Functional Requirements

Functional requirements are the services and responses that users can anticipate from a system. Additionally, it outlines system and user goals [5]. The functional requirements for the developed system are shown in Table 3.

Table 3: Functional Requirements

No.	Module	Descriptions
1.	Register Module	<ul style="list-style-type: none"> • The system shall allow new user to enter their full name, email address , password, and phone number in the registration form. • The system shall warn new user for any invalid input. • The system shall verify the registration details submitted by the user.
2.	Login module	<ul style="list-style-type: none"> • The system shall allow user to log into the system using their valid email and password. • The system shall verify the email and password entered by the user in the email and password data fields. • The system shall warn the user for any invalid input. • The system shall redirect users to their respective homepage after successful login.
3.	Manage Emergency Relief Program Module	<ul style="list-style-type: none"> • The system shall allow user to add new emergency relief program • The system shall allow user to enter the emergency relief program details in the add new emergency relief program form. • The system shall upload the emergency relief program detail to donor's homepage. • The system shall allow user to edit emergency relief program detail. • The system shall allow user to delete emergency relief program detail. • The system shall allow user to cancel the deletion of selected emergency relief program.
4.	Manage Donation Activity Module	<ul style="list-style-type: none"> • The system shall allow user to view the submitted donation details. • The system shall label the donation status as "Pending" as soon as it is submitted to the system. • The system shall allow user to update the donation status as "Completed".
5.	Manage Donation Category Module	<ul style="list-style-type: none"> • The system shall allow user to add donation category. • The system shall allow user to edit donation category. • The system shall allow user to delete donation category. • The system shall allow user to cancel the deletion of selected donation category.
6.	View Donation Report Module	<ul style="list-style-type: none"> • The system shall allow users to select the emergency relief program from the dropdown list.

No.	Module	Descriptions
		<ul style="list-style-type: none"> • The system shall displays donation category totals and its summary based on the selected program. • The system shall allow users to print the emergency relief program report. • The system shall allow users to save the emergency relief program report to PDF format.
7.	Make Goods Donation Module	<ul style="list-style-type: none"> • The system shall allow user to enter their donated goods detail: name, category, image, quantity, condition and description, in the donate form. • The system shall allow user to select donation category of their choice.
8.	View Donation History Module	<ul style="list-style-type: none"> • The system shall allow user to view their donation history and its status.

Non-functional requirements outline the restrictions on the functions and services offered by the system [6]. Table 4 shows the non-functional requirements of the developed system.

Table 4: Non-Functional Requirements

No.	Requirements	Descriptions
1.	Security	<ul style="list-style-type: none"> • The system shall only be accessed by authenticated users.
2.	Learnability	<ul style="list-style-type: none"> • The system shall provide simple user interface and clear navigation that is easy to understand and learn.
3.	Reliability	<ul style="list-style-type: none"> • During a month, the system shall perform its functions flawlessly in 90 percent of use cases.
4.	Localization	<ul style="list-style-type: none"> • The system shall set the date format to be as follows: date.month.year.
5.	Usability	<ul style="list-style-type: none"> • The error rate of users submitting their donation details at the donate page must not exceed 10 percent.

4.2 Use Case Diagram

The system's functional requirements are often described in a use case diagram. Use case diagrams can help clients or users in comprehending how the system needs to function [7]. The use case diagram for the developed system is displayed in Figure 2.

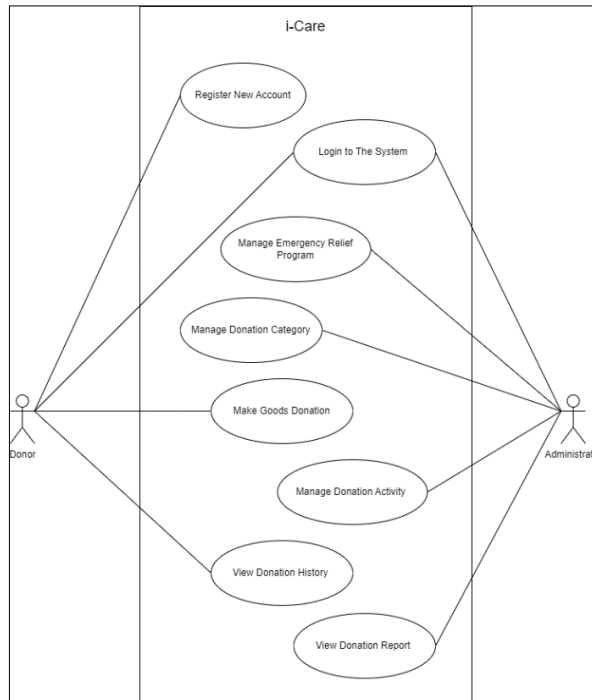


Figure 2: Use Case Diagram

4.3 Class Diagram

UML class diagrams are a crucial tool for developers to comprehend a system's basic structure [8]. The class diagram for the developed system is shown in Figure 3.

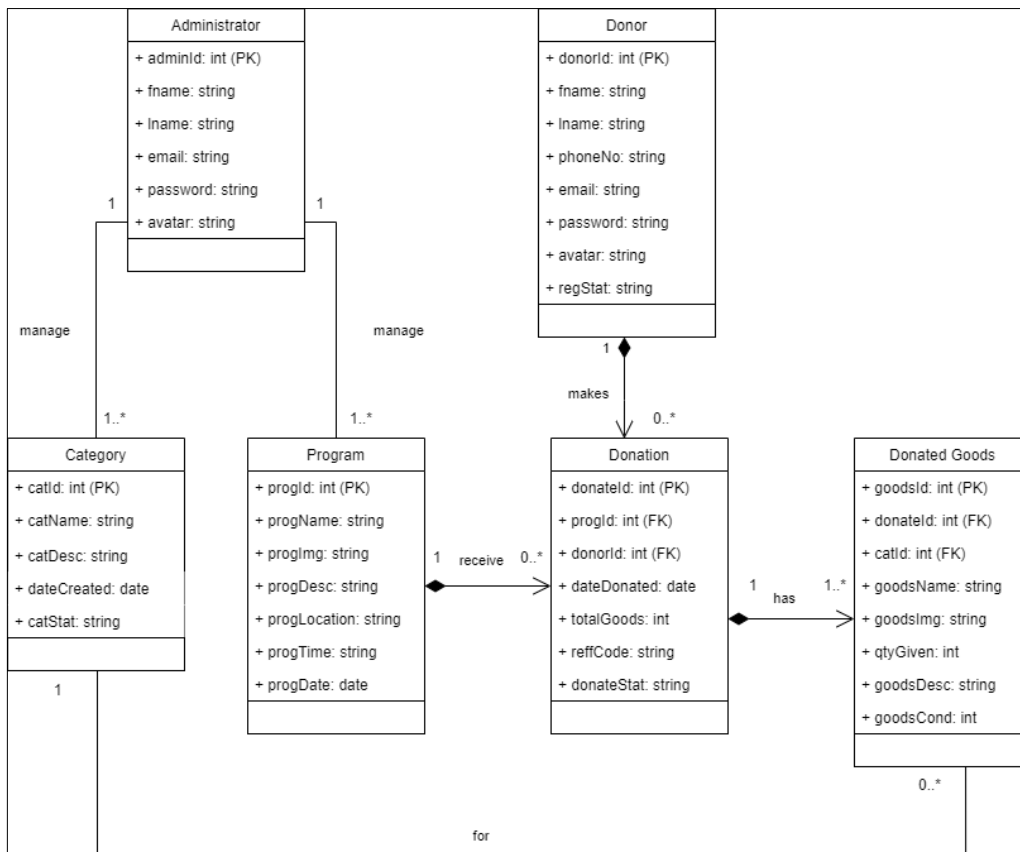


Figure 3: Class Diagram

4.4 System Flowchart

System flowchart defines the flow of procedures or functionality in the developed system[9]. Symbols such as rectangles and arrows indicate the flow of the process in the system flowchart. The system flowchart for the administrator paths is shown in Figure 4, whereas the system flowchart for the donor pathways is shown in Figure 5. Both figures are in Appendix A.

4.5 Implementation

The i-Care system is implemented using Visual Studio Code and XAMPP software. Visual Studio Code is used to program interface modules of the i-Care system. These modules are designed using HTML, CSS, JavaScript, and PHP. XAMPP software executes SQL statements and connects the user interface to the database for data storage.

4.5.1 Register New Account

Figure 6 shows the UI of the register new account module. This interface is exclusively for unregistered donors. Users need to provide their first name, last name, phone number, password, and profile picture. They must also confirm their password within the form and click the "Register" button to complete the registration.

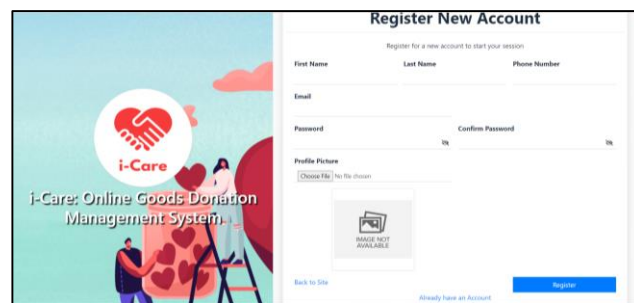


Figure 6: Register New Account Interface

4.5.2 Login to The System

Figure 7 depicts the UI of the administrator login module, and Figure 8 shows the donor login module interface. Users need to provide their registered email address and password in the login form to access the i-Care system based on their assigned role. They can then click the "Log In" button to proceed.

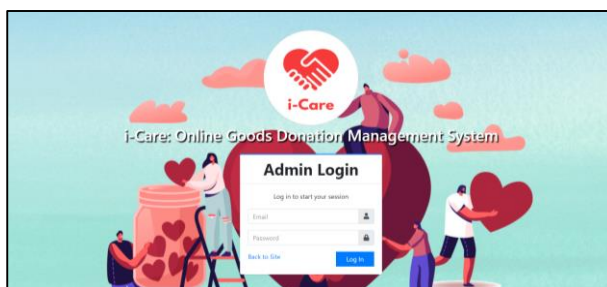


Figure 7: Admin Login Interface



Figure 8: Donor Login Interface

4.5.3 Manage Emergency Relief Program

Figure 9 presents the user interface of the module responsible for managing emergency relief programs. Within this interface, the administrator possesses the ability to add, view, edit, and delete these programs.

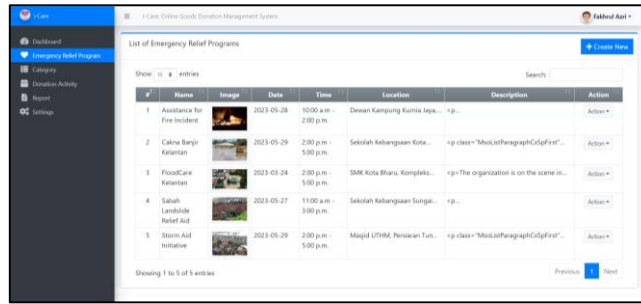


Figure 9: Manage Emergency Relief Program Interface

4.5.4 Manage Donation Category

Figure 10 presents the user interface of the module responsible for managing donation category. Within this interface, the administrator possesses the ability to add, view, edit, and delete these categories.

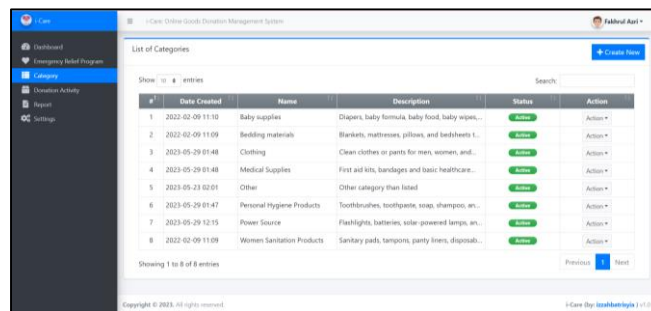


Figure 10: Manage Donation Category Interface

4.5.5 Make Goods Donation

Figure 11 showcases the make goods donation interface, which presents a donation form for the selected emergency relief program that donors wish to donate to. The form requires donors to provide specific details about their donated goods.

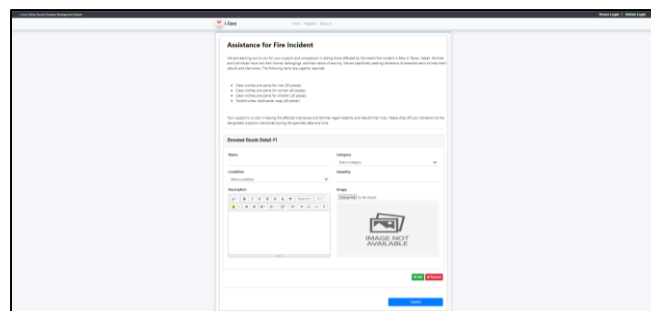


Figure 11: Make Goods Donation Interface

4.5.6 Manage Donation Activity

Figure 12 illustrates the user interface of the manage donation activity module. This interface allows the administrator to both view the details of a donation and update its status.

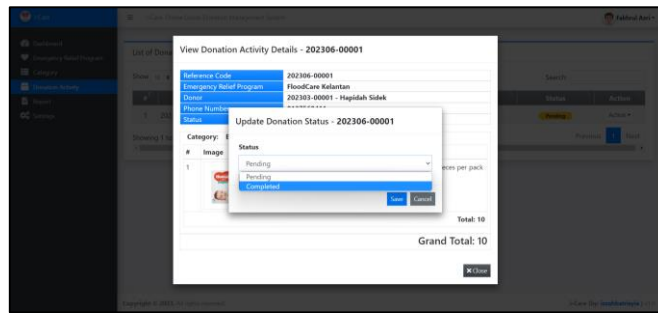


Figure 12: Manage Donation Activity Interface

4.5.7 View Donation History

Figure 13 presents the user interface of the view donation history module. These interfaces provide donors with the ability to access and view their donation history. Donors can view their goods details, reference code for the delivery process, and monitor the status of their donation.

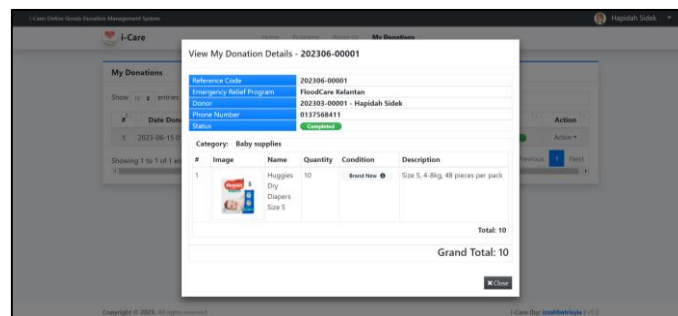


Figure 13: View Donation History Interface

4.5.8 View Donation Report

Figure 14 displays the interface for view the donation report module. Within this interface, the administrator has the capability to view, save to PDF, and print the donation report for a selected emergency relief program.

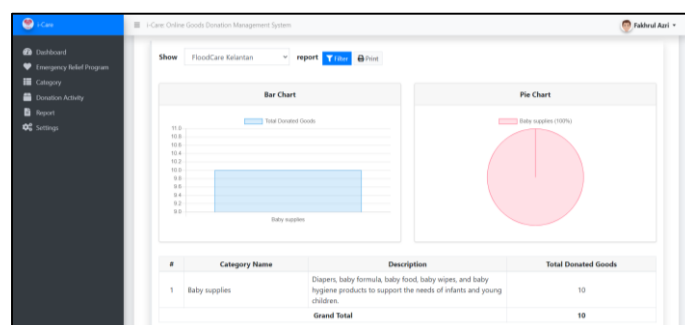


Figure 14: View Donation Report Interface

4.6 Testing

The testing phase is the crucial final step in the development of the i-Care system. Its primary purpose is to thoroughly test every function within the system to identify and rectify any existing deficiencies. This phase holds immense significance as it ensures that the developed system aligns with the project's objectives and effectively meets the requirements of its users.

4.6.1 Test Cases

A test case consists of an input and expected output, designed for execution by a tester with the purpose of uncovering any failures or defects in the software being tested [10]. The test cases for the i-Care system are derived from the functions identified in the test scenarios during its development. Table 5 presents a comprehensive list of test cases specifically tailored for the system.

Table 5: List of Test Cases

Functional Module	Test Case	Description	Expected Result	Test Result (Pass/Fail)
Register New Account	Enter valid registration details	Enter valid first name, last name, email address, password, phone number, and profile picture in the registration form.	Upon successful registration, the donor shall receive a confirmation message.	Pass
	Enter invalid registration details	Enter invalid first name, last name, email address, password, phone number, and profile picture in the registration form.	Upon unsuccessful registration, the registration shall not proceed until all invalid inputs are corrected.	Pass
Login to The System	Enter valid user credentials	Enter valid email and password in the login form.	Upon successful login, the system shall redirect the user to their respective homepage.	Pass
	Enter invalid user credentials	Enter invalid email and password in the login form.	Upon unsuccessful login, the system shall display a warning message indicating invalid input.	Pass
Manage Emergency Relief Program	View emergency relief program	View selected emergency relief program detail	The system shall retrieve and display the details of the selected emergency relief program accurately.	Pass
	Add emergency relief program	Enter program's image, name, date, time, drop-off location and description on the Add New Emergency Relief Program form.	The new emergency relief program shall be stored correctly in the system's database.	Pass

Functional Module	Test Case	Description	Expected Result	Test Result (Pass/Fail)
	Edit emergency relief program	Enter program's updated image, name, date, time, drop-off location and description on the Edit Emergency Relief Program form.	The modified details shall be correctly saved and updated in the system's database.	Pass
	Delete emergency relief program	Delete the selected emergency relief program detail.	Upon deletion confirmation, the program detail shall be permanently removed from the system's database.	Pass
Manage Donation Category	View donation category	View selected donation category detail	The system shall retrieve and display the details of the selected donation category accurately.	Pass
	Add donation category	Enter category's name, status, and description on the Add New Category form.	The new donation category should be stored correctly in the system's database.	Pass
	Edit donation category	Enter category's updated name, status, and description on the Edit Category form.	The modified details should be correctly saved and updated in the system's database.	Pass
	Delete donation category	Delete the selected donation category detail.	Upon deletion confirmation, the category detail should be permanently removed from the system's database.	Pass
Make Goods Donation	Fill in donation form for the selected program	Enter the details of donated goods such as name, category, image, quantity, condition, and description in the donation form.	The system should accept and store the entered information accurately.	Pass

Functional Module	Test Case	Description	Expected Result	Test Result (Pass/Fail)
Manage Donation Activity	View submitted donation details	View selected donation details that has been submitted.	The system should retrieve and display the details of the selected donation accurately.	Pass
	Update donation status	Select current donation status: "Pending" or "Completed"	The status change should be accurately processed and reflected in the donation details.	Pass
View Donation History	View donation history and status	View selected donation detail made in the past and present.	The system shall retrieve and display the donation detail for the donor accurately.	Pass
View Donation Report	Display donation category totals and summary	Select emergency relief program and filter the report detail	The system shall calculate and display the donation category totals, representing the aggregated donations for each category within the selected program in a table, pie chart, and bar chart manner.	Pass
	Save donation report to PDF	Save donation report to PDF on the device.	The system shall generate a PDF version of the donation report.	Pass
	Print donation report	Print the donation report on the selected printer.	The system shall generate a printable version of the donation report.	Pass

4.6.2 User Acceptance Testing

User acceptance testing is a critical phase where actual users evaluate the i-Care system's acceptance and usability based on defined requirements. An online survey using Google Forms was conducted, with 11 respondents participating. Notably, Mr. Kamsani bin Nasir, the CEO of Yayasan Ikram Malaysia, provided insights from an administrative perspective. The remaining 10 respondents were individuals who actively assist emergency victims, serving as donors within the i-Care system. Figure 15 showing the test results for interface-related questions in the first part of the survey.

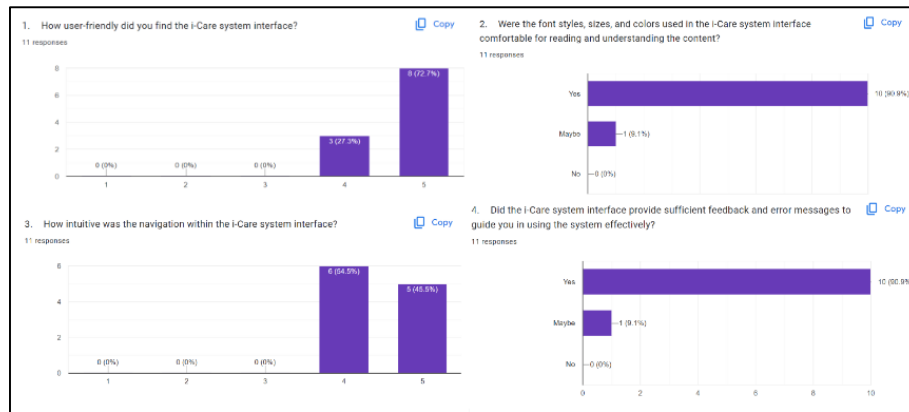


Figure 15: Graphs of Testing Results for System Interfaces

Regarding the system interface, the survey results indicate that a majority of respondents (72.7%) found it highly user-friendly, while 27.3% considered it mostly user-friendly. In terms of font styles, sizes, and colors, 90.9% of participants found them comfortable for reading and comprehending the content, though some had reservations. Regarding navigation, only 45.5% believed it was highly intuitive, while the rest considered it mostly intuitive. For feedback and error messages, the "Yes" option received the highest response rate, with 90.9% expressing satisfaction with the i-Care system interface's guidance. These findings provide valuable insights into user opinions and satisfaction levels regarding the interface. Figure 16 illustrates the test results for each question related to system's functionality. The second part of the survey consisted of four questions.

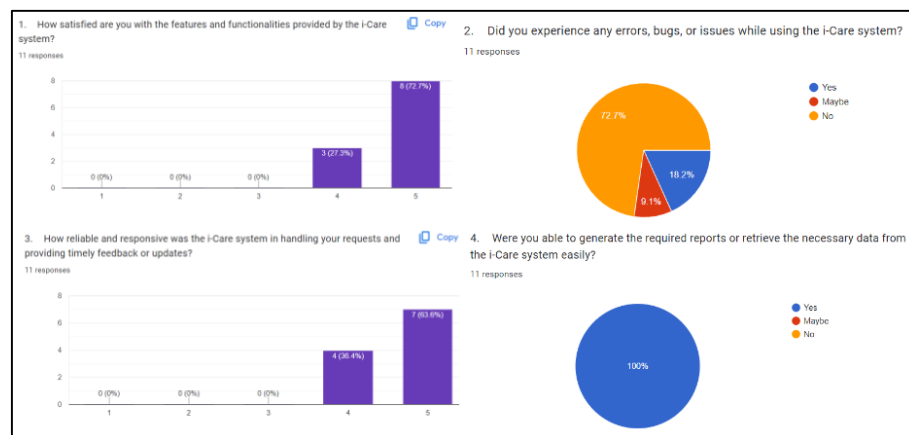


Figure 16: Graphs and Charts of Testing Results for System Functionality

Regarding system functionalities, the survey results show that a majority of respondents (72.7%) expressed high satisfaction, while 27.3% found it mostly satisfying. In terms of encountering errors, bugs, or issues, 72.7% reported such problems, while 9.1% may have experienced them and 18.2% confirmed encountering them. When it comes to reliability and responsiveness in handling user requests and providing timely feedback or updates, only 63.6% believed the i-Care system demonstrated high levels, while the rest considered it mostly reliable and responsive. For generating reports and retrieving data, 100% of participants agreed that the system performed these tasks with ease. These findings provide insights into user opinions and satisfaction levels regarding the system's functionalities.

5. Conclusion

In conclusion, the i-Care system has successfully addressed the limitations of indiscriminate donations, limited aid diversity, and manual data management. Through rigorous testing and observation, the system has demonstrated several advantages, such as customized aid for each

emergency relief program, efficient resource allocation, effective donation tracking, and a streamlined donation process. However, there are also identified shortcomings, including issues with displaying uploaded pictures, an unorganized mobile view of the program detail page, and the absence of a dropdown menu on the navigation. To improve the system further, recommendations have been proposed, including the implementation of a volunteer management module, a communication and notification system, and the addition of a dropdown menu. These enhancements will enhance the system's functionality and user experience. Overall, the i-Care system has been successfully developed, achieving its objectives, and providing a systematic approach to managing goods donation during emergency situations for NGOs.

Acknowledgment

The authors would like to thank the Faculty of Computer Science and Information Technology, Universiti Tun Hussein Onn Malaysia for its support.

Appendix A

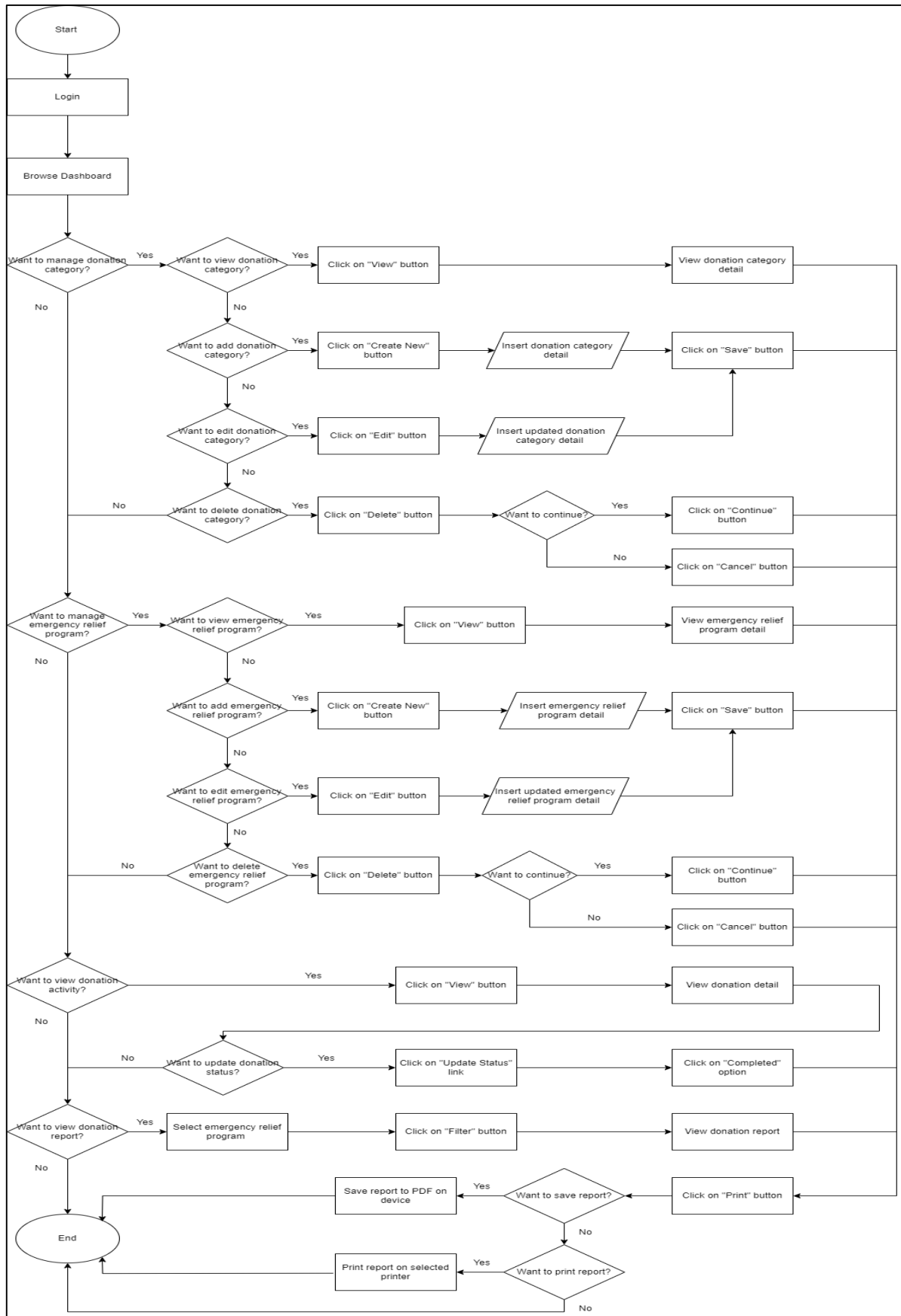


Figure 4: Administrator System Flowchart

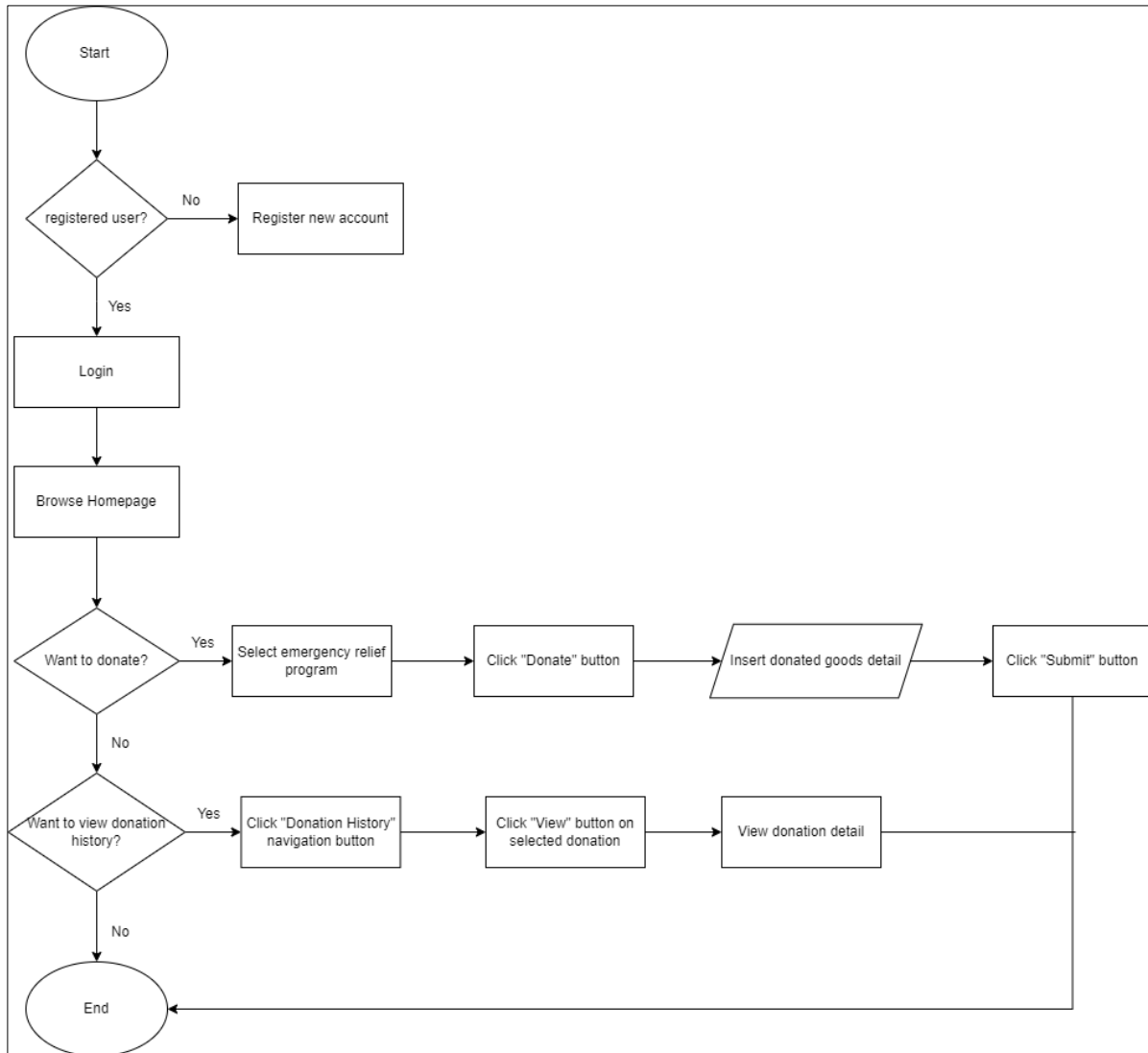


Figure 5: Donor System Flowchart

References

- [1] M. Siahaan and A. Bimantoro, "DESIGN AND DEVELOPMENT OF USEFUL ITEM DONATION APPLICATIONS WITH SCRUM METHOD," *CoMBInES - Conference on Management, Business, Innovation, Education and Social Sciences*, vol. 2, no. 1, Art. no. 1, Apr. 2022, Accessed: Dec. 17, 2022. [Online]. Available: <https://journal.uib.ac.id/index.php/combines/article/view/6656>
- [2] R. I. A. Pribadi, A. Pambudi, and Ardiansyah, "eDonation Android Application for Used Goods Donation using Location-based Service," *J. Phys.: Conf. Ser.*, vol. 1751, no. 1, p. 012037, Jan. 2021, doi: 10.1088/1742-6596/1751/1/012037.
- [3] K. Pal, "Markov Decision Theory-Based Crowdsourcing Software Process Model," 2020, pp. 1–22. doi: 10.4018/978-1-5225-9659-2.ch001.
- [4] A. Dennis, B. H. Wixom, and R. M. Roth, *Systems Analysis and Design*. John Wiley & Sons, 2019.
- [5] N. Rahimi, F. Eassa, and L. Elrefaei, "An Ensemble Machine Learning Technique for Functional Requirement Classification," *Symmetry*, vol. 12, no. 10, Art. no. 10, Oct. 2020, doi: 10.3390/sym12101601.
- [6] M. G. Lorgat and H. Paredes, *Handling Non-Functional Requirements in Software Architecture*. 2021.
- [7] R. Fauzan, D. Siahaan, S. Rochimah, and E. Triandini, "A Different Approach on Automated Use Case Diagram Semantic Assessment," *International Journal of Intelligent Engineering and Systems*, vol. 14, pp. 496–505, Feb. 2021, doi: 10.22266/ijies2021.0228.46.
- [8] B. Gosala, S. R. Chowdhuri, J. Singh, M. Gupta, and A. Mishra, "Automatic Classification of UML Class Diagrams Using Deep Learning Technique: Convolutional Neural Network," *Applied Sciences*, vol. 11, no. 9, Art. no. 9, Jan. 2021, doi: 10.3390/app11094267.
- [9] R. Setiawan and Z. Kisman, "Design Mobile Application for Health Consultation during Pandemic Covid 19," *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, pp. 4241–4244, Nov. 2020.
- [10] N. Setiani, R. Ferdiana, P. I. Santosa, and R. Hartanto, "Literature Review on Test Case Generation Approach," in *Proceedings of the 2nd International Conference on Software Engineering and Information Management*, in ICSIM 2019. New York, NY, USA: Association for Computing Machinery, Jan. 2019, pp. 91–95. doi: 10.1145/3305160.3305186.