Inform : Journal Scientific Field Technology Information and Communication
Vol.9 No.1 January 2024, P-ISSN : 2502-3470, E-ISSN : 2581-0367

# *ICONIX Process for Analysis and Design of Web-Based Savings and Loan Cooperative Applications*

Aisha Safa Asy'ari[1], Amanda Salsabilla Hidayah[2], Venira Citra Sabrina[3], Seftin Fitri Ana Wati[4*]

[1,2,3,4]*Information Systems Department, Universitas Pembangunan Nasional "Veteran" Jawa Timur, Indonesia*

[1]aishaasyari@gmail.com, [2]amandasalsabilla630@gmail.com, [3]veniracs7@gmail.com, [4]seftin.fitri.si@upnjatim.ac.id (*)

Received: 2022-06-30; Accepted: 2022-07-22; Published: 2023-09-21

***Abstract*—** The rapid advancement of information technology has led to the computerization of activities in virtually all domains. One notable example is the website, which is widely utilized by the public today to facilitate various business processes. Technology's proliferation has also significantly simplified and enhanced solutions to economic challenges, especially in cooperatives. Cooperatives play an instrumental role in fostering economic growth. While cooperative management utilizes applications, many administrative activities remain manual, lack real-time processing, and are only accessible to the cooperative's administrators. The process is challenging for cooperative members who wish to view transaction reports or conduct activities like installments, deposits, Surplus Dividends (SHU), or loans; they must physically visit the cooperative. This paper will delve into the steps in analyzing and designing a web-based cooperative application utilizing the Iconix process. The system analysis stages encompass data gathering via observation, interviews, and questionnaires. The system design stages cover the design of the user interface (GUI) graphics, use case diagrams, domain model diagrams, robustness diagrams, sequence diagrams, and class diagrams. This study's findings will benefit cooperative stakeholders looking to establish a web-based savings and loan system. This will provide them with a comprehensive overview of business processes through the analysis and design phases, all leveraging the Iconix process.

***Keywords*—** System Analysis; System Design; Cooperatives; Savings and Loans; Iconix Process.

## I. INTRODUCTION

The economy plays a pivotal role in determining the welfare of the people in Indonesia. This significance is reflected in the fifth precept, which emphasizes developing and enhancing the economy to ensure prosperity for all Indonesians. One effective approach to bolster the economy is promoting and engaging in cooperative activities within Indonesia [1]. The swift advancement of information system technology can be harnessed to address economic challenges. In today's digital era, web-based applications are indispensable for businesses. This is because information system technology can streamline and facilitate organizational or company processes, especially in activities related to cooperatives [2].

The case study discussed in this paper focuses on the STIE Perbanas Cooperative, a prominent cooperative situated at Hayam Wuruk Perbanas University in Surabaya. This cooperative operates in the savings and loan sector and undertakes many administrative activities. Integrating digital or computerized tools is crucial to ensure the efficient execution of all transactions. This enables the cooperative to offer swift, precise, and accurate services. By embracing digital technology, the cooperative can facilitate the tasks for its employees and substantially reduce risks associated with savings and loan operations.

The problem involved in this collaboration is that the STIE Perbanas cooperative savings and loan administration is still administered manually and is accessible only to admins, so cooperative members who want to know about the bank's savings and loan transactions report should come directly to the cooperative or manually. Administrative activities in cooperatives become inefficient and not optimal because they waste a lot of time and energy.

In a prior research literature review, the design of a website-based application was discussed, which aids in recording deposit, loan, and installment data. Previously, these tasks were handled conventionally and not stored in a database. This omission posed potential harm to both parties involved. The website-based application was created to address these issues to ensure easier access, improved efficiency, and minimized redundancy. This study resulted in the design of a cooperative website application with dual access for both admin and members [3]. Another literature study elaborated on the design of a web-based application intended to support data recording activities, report generation, and loan applications that were initially managed manually, necessitating members to visit the cooperative in person. This manual method often led to calculation errors and significantly slowed the process. Hence, a website-based application was introduced to streamline data acquisition, promoting efficiency and reducing errors. This research culminated in designing a website-based cooperative application granting access to admins and members [2]. Drawing from these literature reviews, this research's foundation has been established. It aims to assist in the administrative functions of savings and loan cooperatives, specifically in managing data related to savings, members, loans, and SHU and other data recording and payment processes by members. The expected output is a series of reports available on the website. The proposed analysis and design can effectively transition the savings and loan cooperative website to real-time operation. This ensures that data is consistently updated, presenting users with the most recent information. Moreover, the website, designed for admins and members, empowers members to easily access and review transaction reports.



DOI : https://doi.org/10.25139/inform.v9i1.4741

The Iconix process is a design method that comprises dynamic models that describe the system's behavior [4]. A notable advantage of the Iconix process is its utilization of UML, known for its simplicity and elegance. Another benefit of the Iconix process is its incorporation of robustness diagrams, which are instrumental in analyzing reliability. Given the abovementioned problems, this study aims to design a new proposed system: a website-based platform to aid cooperative members and employees in administrative tasks. This ensures that transaction reports, along with the residual business outcomes, are accessible in real time. The payment transactions can be executed online via the website. The anticipated outcome of this research is that the analysis and design of this application will streamline and enhance cooperative activities, making them more effective and efficient. Furthermore, this study can serve as a valuable reference for the Hayam Wuruk University Surabaya's savings and loans cooperative in managing their information system.

## II. RESEARCH METHODOLOGY

The Research Methodology used in "ICONIX Process for Analysis and Design of Web-Based Savings and Loan Cooperative Applications" is qualitative and Descriptive. Qualitative Research is Research that is directly related to human social events to understand problems that occur in social and human life and is usually directly related to the object of research. This research is different from quantitative research because it does not use statistics but through data collection and analysis [2]. Steps in research include:

### *A. Data Collection*

*1) Questionnaire:* A questionnaire is a data collection technique consisting of several written questions about the issue. It is presented to respondents to gather information based on their answers to the provided questions [5]. In this study, the questionnaire was distributed to 25 Hayam Wuruk University cooperative members.

*2) Observation:* Involves studying an event based on factual occurrences and reality [6]. In this study, direct observations were made at Hayam Wuruk University. Such observation aims to process and evaluate data, thereby aiding the research process.

*3) Interview Method*: An interview is a meeting of two or more people to exchange information or communicate ideas or a problem, employing a question and answer to explain the essence or idea of the problem into a new topic [6]. Interviews in this study were conducted with related parties, namely one of the cooperative members in charge of managing cooperative data and several cooperative members who became permanent employees, about how savings and loan cooperative activities worked and what obstacles were happening.

*4) Literature Study*: View or study and collect several references from journals that discuss the same problem or previous journals.

### *B. System Requirement Analysis*

In the Needs Analysis, the approach entails identifying the involved objects, their attributes, and the activities performed by these objects and delineating the relationships between them. The needs analysis assesses business processes and defines functional and non-functional requirements.

### *C. System Design*

This study was designed using a UML (Unified Modelling Language) diagram, which is a language that explains, visualizes, and builds based on graphs or images from a software development system. UML is also a writing standard that includes concepts from business processes and explanations of classes in program languages and databases. It describes some of the components needed so that the language in the program can be understood by humans and machines themselves [7]. As well as using the Iconix Process Method, which is a method that serves as a bridge between creating program code according to the needs of the system described in the use case created later

The system design in this study consists of GUI, domain model, use case diagrams, robustness diagrams, sequence diagrams, and diagram classes. This study used a literature review of the design of a website-based cooperative application.

### *A. Savings and Loans Cooperative*

Law Number 17 Article 1, paragraph 1 of 2012 states that a cooperative is a legal entity established by a group of people with the separation of the wealth of its members as capital to run its business that can meet common needs in the economic, social, and cultural fields. A savings and loan cooperative is a cooperative business activity that collects and distributes funds from and to its members, where the accumulated savings funds are used as working capital, which can also be distributed as loans to its members [8]

### *B. Iconix Process*

Iconix process is an approach among the Rational Unified Processes that aims to connect the creation of program code based on the use case designed for determining the model and behavior of the system [4]. In the use of this short Iconix process, it does not neglect analysis and design. Even using UML in the Iconix process will be more efficient because it focuses on each need [9]. UML is a multi-lingual and open-source project that can develop a unified modeling language platform in a practical, flexible, and complete manner at no cost to build software modeling. One of the important diagrams often used in illustrating the needs of a system is the use case diagram [10]. Figure 1 is an image of the Iconix cycle process used in this study. Figure 1 illustrates the Iconix process, which has several processes in its stages, including Requirement, Analysis / Preliminary Design, Detailed Design, and Implementation. In Iconix, the process above starts from the GUI (Graphical User Interface), an interactive visual



DOI : https://doi.org/10.25139/inform.v9i1.4741

component system that helps convey and present information to users.

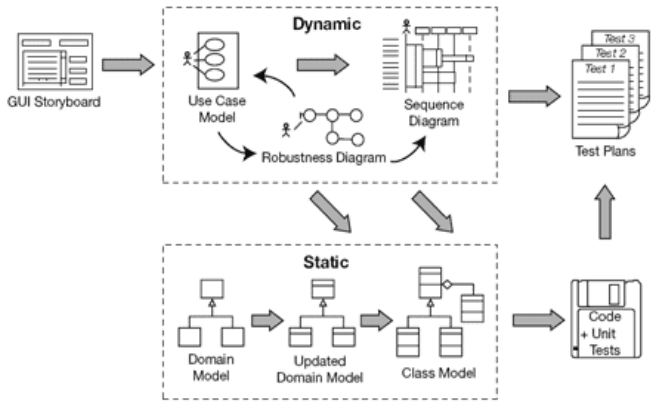


Figure 1. ICONIX Process

The GUI contains object components such as icons and buttons [11]. Next is the use case, which plays an important role in controlling the design of the Iconix process [10]. The use case diagram illustrates the functionality performed by the system and also states a visualization of the interaction between the actor and the system [12]. The subsequent step involves using a sequence diagram, which illustrates the interaction between objects in chronological order. Sequence diagrams depict the sequence of messages exchanged and interactions among objects that occur at specific moments during the system's execution [13]. The diagram consists of vertical dimensions (time) and horizontal dimensions (objects), and messages (messages) are depicted with a line between objects [10]. The next is the diagram class, which leads to the relationships that occur in the system [13].

### *C. Business Process*

A business process comprises a series of interconnected activities undertaken by an organization or a consortium of companies to accomplish a shared objective [9]. Business process modeling is executed to enhance existing business procedures [14].

## III. Result and Discussion

The research "ICONIX Process for Analysis and Design of Web-Based Savings and Loan Cooperative Applications" uses object-based design, namely UML and ICONIX processes.

### *A. Business Process analysis*

This study conducts business process analysis using BPMN (Business Process Modeling Notation). This modeling elucidates the current and proposed Cooperative Savings and Loan Application business processes. Below are the models of the current and suggested business processes for the cooperative's savings and loan activities that have been identified:

- The current cooperative transaction process
- The process of viewing current transaction reports
- The proposed cooperative transaction process
- Proposed view transaction reports

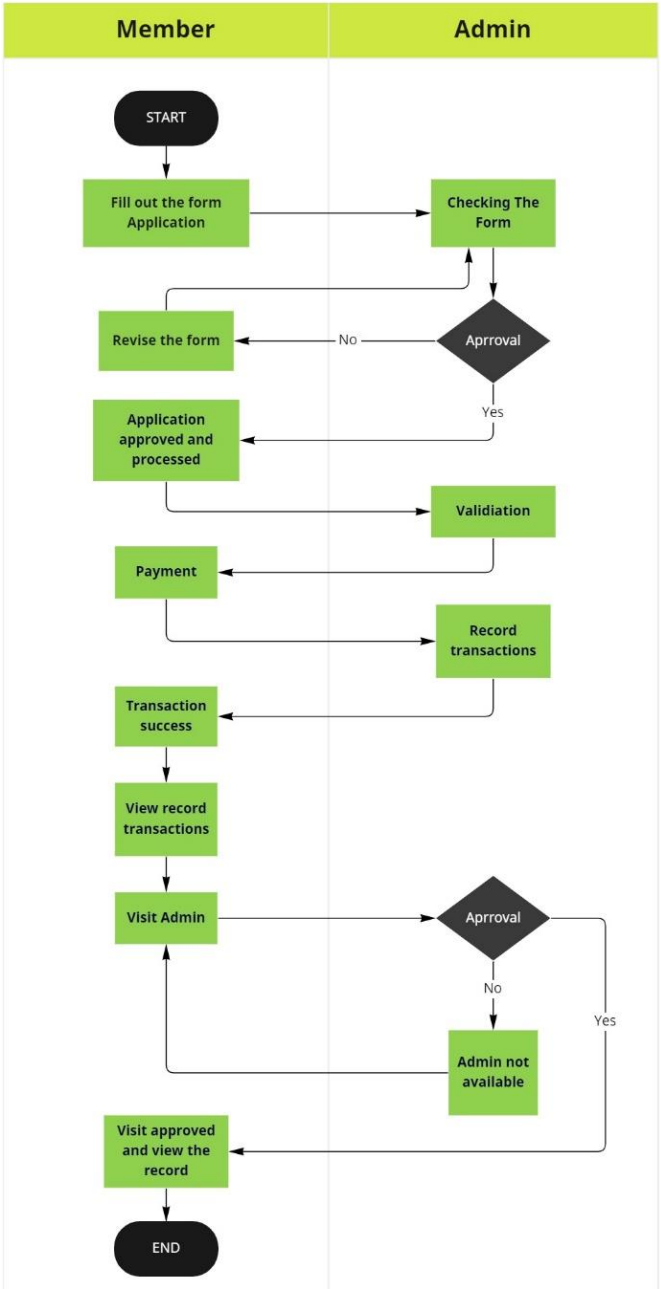


Figure 2. Modeling Current Business Processes

Figure 2 illustrates the business processes in current savings and loan cooperatives. To initiate a transaction, members begin by completing a form. Subsequently, the Admin reviews and, if deemed appropriate, approves the form. Upon approval, the transaction request undergoes further processing. If the admin disapproves the form, the member is tasked with making revisions. The member executes the payment transaction once the form is resubmitted and validated by the admin. After the payment is completed, the admin records the transaction, marking the successful completion of the process. To view a transaction report, members must request an in-person meeting



DOI : https://doi.org/10.25139/inform.v9i1.4741

with the admin, as only the admin has conventional access to the transaction reports. If the Admin consents, the member can then view the report. However, if the admin declines, often due to unavailability, members can make a subsequent request.

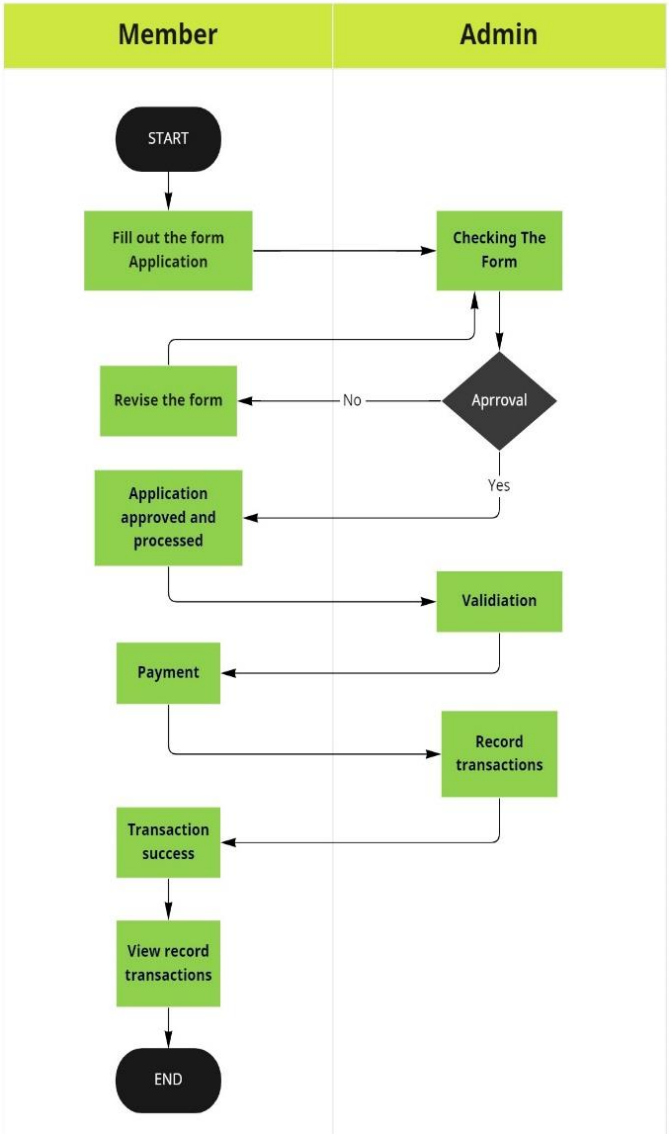


Figure 3. Business Processes modeling proposals

Figure 3 is the proposed modeling of business processes in savings and loan cooperatives. To make a transaction, members start by filling out the form, then the admin will check and approve the form, and then the transaction submission will be processed further. If the admin does not approve the submitted form, the member revises the submitted form. After that, the admin validates the data, and the member makes a payment transaction. After the member makes the payment, the admin will record the transaction and complete the process. Then, the member can view the transaction report. Members can view reports online without meeting the admin first as the business processes are currently implemented because the system is built in real-time and can be accessed anytime and anywhere on the *website* with the help of the *internet*.

*B. Requirement*

This study identified system needs according to the constraints and problems, and two actors were identified, including admins and members. The following are the specifications of functional needs, non-functional, Domain Model, Behavioral Requirement or GUI, and Use Case Diagram [15].

*1) Functional Requirement:* Functional requirements are specifications used in design that detail the processes essential to or utilized by the system. Table I explains the functional needs of 2 actors, admins, and members contained in the savings and loan cooperative system of Hayam Wuruk Perbanas University Surabaya.

TABEL I
FUNCTIONA NEEDS

| Actor | Necessity | Description |
|---|---|---|
| Admin | Dashboard | View the main page of a web-based application |
| | Login | Access to System Login |
| | Menu | Displays various menus of savings and loan transaction activities |
| | Member Data | display information and add or change all data of the member's savings and loan transaction activity |
| | SHU | View and add details of SHU calculations received by all members. |
| Member | Account Registration/Sign in | Cooperative member account registration access. |
| | Loan Application | It displays the total nominal value of the loan owned and the loan application form. |
| | Installment | Displays the remaining installment costs that need to be paid and the installment payment form. |
| | Deposits | Displays the total nominal of deposits owned and the form of adding deposits |
| | Savings and Loans Transaction Report | Display all data on savings and loan cooperative activities in real time. |
| | SHU | Displaying the details of the calculation of the received SHU |
| | Logout | Access to exit the system. |

*2) Non-Functional Requirement:* Non-functional requirements describe the behavior of a system. For this system, the non-functional requirements encompass security, ensuring limited user access since data security is predominantly accessible only by the admin. The system is user-friendly, allowing cooperative members to navigate with ease. Reliability is paramount, ensuring the system can deliver real-time information data and always remain available. Regarding operational needs, the hardware consists of laptops,



DOI : https://doi.org/10.25139/inform.v9i1.4741

computers, and smartphones, while the software includes Windows, websites, and StarUML.

*3) Behavioral Requirement:* A GUI display has been devised to illustrate the system's flow. A GUI, or Interface, represents the initial design or depiction of the system in development. In this study, the GUI is depicted in Figure 4, showcasing the login screen, and Figure 5, detailing the member's SHU data. Figure 4 is an interface image of the web page from the login menu for Members and Admins to enter the savings and loan cooperative application. Figure 5 is an interface image of the web page from the SHU Data menu for Members to see the total SHU nominal received by members on the savings and loan cooperative application.

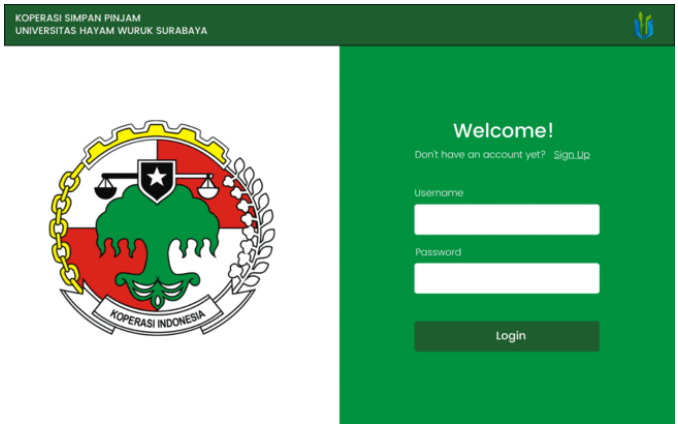


Figure 4. Interface Login

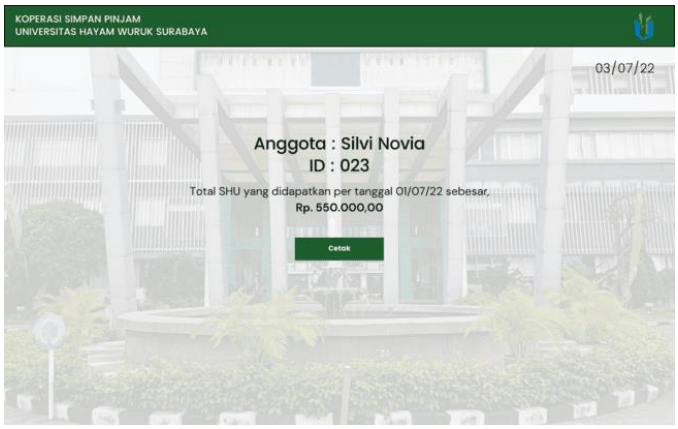


Figure 5. Member SHU Data Interface

4) *Domain Model:* The domain model explains the architectural structure of a static system, functioning to find out what objects are contained in the system that we design later. In this study, the Domain Model has Figure 6, namely the object structure of the savings and loan cooperative system. Figure 6 presents a Domain Model detailing the architectural structure of the savings and loan cooperative system, encompassing two actors: admins and members. Admin actors log in with their specific admin account. Given that this account forms an aggregation or is part of the admin entity, it is closely linked with member data and SHU. This implies that the admin interface will display menus related to member data and SHU. Within the member data, there's an association with transactions, signifying that these transactions, which span installments, deposits, and loan applications, are integral to the member data. The SHU menu is tied to the admin, highlighting that the admin account manages the SHU. On the other hand, the member actor is associated with account registration, implying that it's a crucial component of the member's identity. Without an account, a member cannot access the system. Following registration, the member's account links to reports and SHU, meaning that the system provides these members with access to transaction reports and SHU menus related to their account's transaction activities. In the system's functionality, members can make installments and deposits, which then undergo verification. Post-verification, members can proceed if they wish to take out a loan. Ultimately, members can view their transaction reports and SHU.

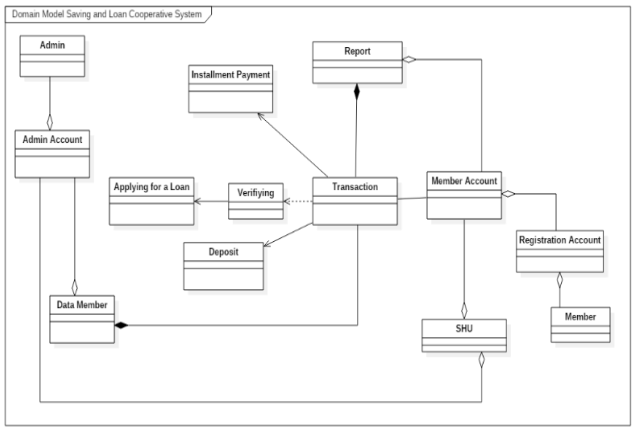


Figure 6. Domain Model

*5) Use Case Diagram:* Use case Diagrams describe the system from the user's point of view so that it focuses on the functionality of the system. Use Case Diagram Presents an interaction between the actor and the system. In this study, the Use Case is found in Figure 7. Figure 7 is a use case diagram that explains the savings and loan cooperative system consisting of 2 actors, namely admins and members. Each actor has a different task. Admins in the system, namely login, can manage member data containing installments, loan applications, Deposit Transactions, and Transaction Reports and can manage SHU. Members in the system can register, log in, view transaction reports, pay installments, apply for loans, deposit transactions, and view SHU. To include means that in the admin actor managing member data, there are installments, loans, and deposits. So, members must first make some of these transactions, and then the admin can manage the member's data. Meanwhile, the extension on viewing transaction reports is the output of deposit transactions and loan applications as well as installment payments, so when members want to see transaction reports, several transactions will appear that have been made.



DOI : https://doi.org/10.25139/inform.v9i1.4741

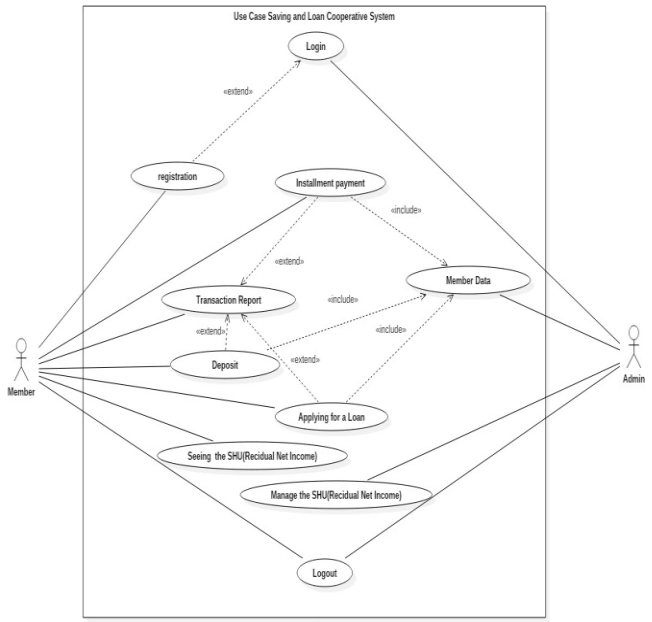


Figure 7. Use Case Diagram

### D. *Analysis of Preliminary Design*

The design process draws from previously established methods, connecting needs analysis to system design [15]. The primary goal of this modeling is to obtain a clear understanding of data flow, control mechanisms, functional processes, operational behavior, and embedded information [16]. Robustness Diagrams play a pivotal role in this phase. They clarify a use case, simplifying the flow for design purposes and enhancing both the use case text and the object model. The diagrams encompass several structures, including the Boundary Object, Entity Object, and Controller. Among the robustness diagrams, we have Figure 8, which illustrates account registration; Figure 9, which delineates login procedures; Figure 10, which details the admin's management of member data; Figure 11, which depicts savings and loan transaction reports; Figure 12, which outlines the addition of SHU data by admins; Figure 13, which presents members' viewing of SHU data; Figure 14, which describes the deposit application process; Figure 15, which shows installments; Figure 16, which provides insights into loan applications; and finally, Figure 17, which represents the logout process.

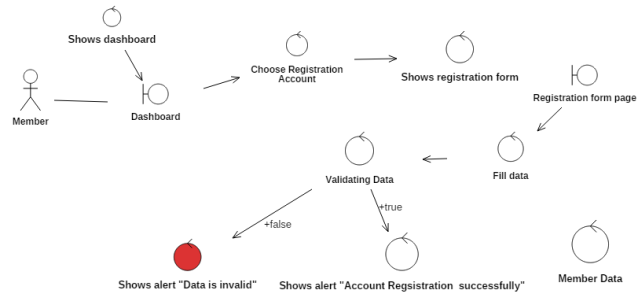


Figure 8. Robustness Of Account Registration Diagram

Figure 8 presents the Robustness Diagram for Account Registration undertaken by members. Members begin by logging in on the start page and then choosing the account registration menu, which includes the account registration form. Following this, members input their data into the form. The system checks the provided data for validity. If any discrepancies are detected, a notification stating "Data does not match" appears. However, if the data aligns with requirements, the system displays a message: "Account registration was successful", indicating that the member's account details have been successfully saved.

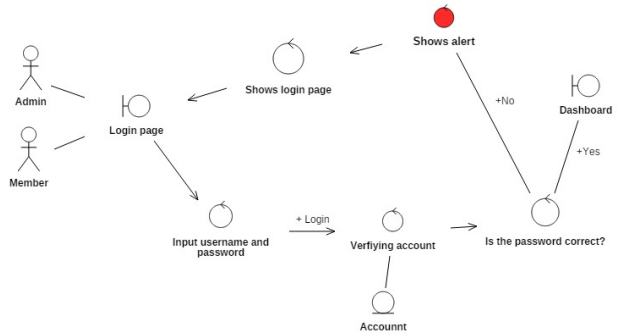


Figure 9. Robustness Diagram Login

Figure 9 depicts the Robustness Diagram for Login, which involves two actors: admins and members. The user (Admin or Member) accesses the savings and loan cooperative system website, which presents the login page. The user then enters their username and password and clicks the "Login" button. Subsequently, the system verifies the account credentials. If the entered username and password are correct, the system directs the user to the main page. However, suppose there is a discrepancy in the credentials. In that case, the system issues a notification stating, "The username and password you entered are not appropriate!" and the user is returned to the login page.

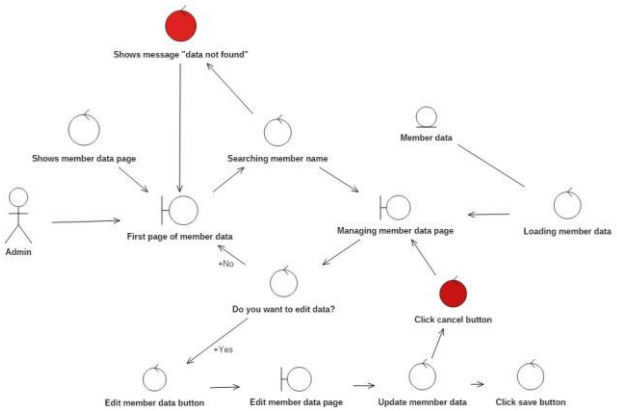


Figure 10. Robustness Diagrams of Managing Member Data

Figure 10 depicts a Robustness Diagram related to the admin's savings and loan cooperative system management. Upon entering the system's start page dedicated to member records, the Admin searches for a specific member's name. A "data not found" message is displayed if the name is not located.



DOI : https://doi.org/10.25139/inform.v9i1.4741

However, if the member's name is validated, the system redirects the admin to the "manage member data" page, where the pertinent member data is presented. Should the admin choose to modify any data, they click the "edit member data" button, make the necessary changes on the subsequent page, and then click "Save" to update the information. Opting not to save any changes can be done by clicking the "cancel" button. If the admin decides against editing the member's data, the system returns them to the initial member data menu.

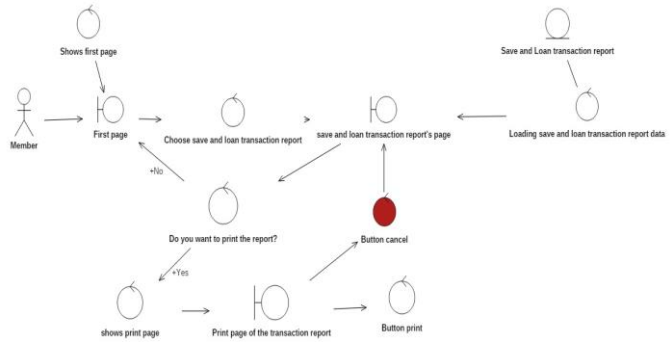


Figure 11. Robustness Diagram of Savings and Loans Transaction Report

Figure 11 is a Robustness Diagram of the Savings and Loan Transaction Report conducted by members. Members enter on the start page and select the savings and loan transaction report. Members enter on the deposit and loan transaction report page that has been loaded. If the member chooses to print the transaction report, enter the print report page, and there is a print and cancel button. If the member does not want to print the transaction report, the member will return to the start page.

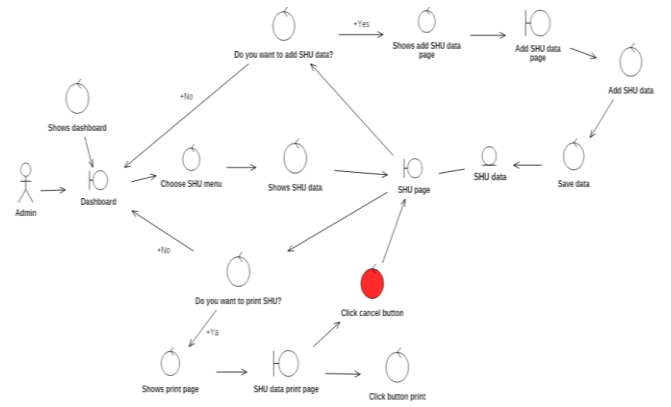


Figure 12. Robustness Diagram admin add SHU data

Figure 12 is a Robustness Diagram to add SHU data performed by an admin. The admin logs in on the main page and selects the SHU menu. The system will display SHU and admin login data on the SHU page. If the admin chooses to print SHU data, enter on the print page, and there are print and cancel buttons. If the admin does not want to print SHU data, the admin will be back on the main page. If the admin chooses to add SHU data, then enter on the add SHU data page, then the admin can fill in additional SHU data after which the data is saved. If the admin does not want to add SHU data, the admin will return to the main page.

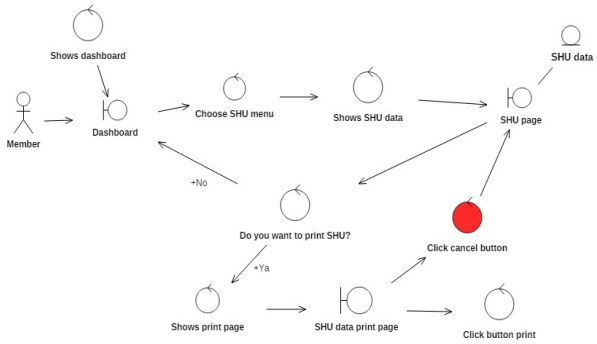


Figure 13. Robustness Of Member Diagrams Viewing SHU Data

Figure 13 is a Robustness Diagram looking at SHU data performed by members. Members log in on the main page and then select the SHU menu. The system will display SHU and member login data on the SHU page. If the member chooses to print SHU data, enter on the print page, and there are print and cancel buttons. If the member does not want to print the SHU data, then the member will return to the main page.

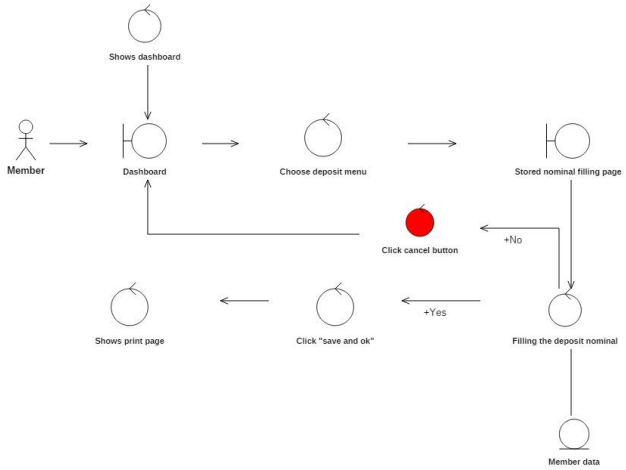


Figure 14. Robustness of Deposit Filing Diagram

Figure 14 presents a Robustness Diagram illustrating Deposit Filings by members. Members initiate from the Start Page, choose the Deposit Transaction menu, and the system then displays a page for entering the deposit amount. Members input the desired deposit amount. If they decide not to proceed with the transaction or wish to change the amount, they can click the "cancel" button, which redirects them to the start page. Members click "ok" to save if the deposit amount is finalized. Subsequently, the system generates a receipt, and the entered data is stored in the member's profile. Figure 15 showcases a Robustness Diagram related to Installment Payments by members. Members select the Installment menu from the Start Page, prompting the system to present the Installment Payment page. Here, members can finalize their installment payment by clicking "pay". If they opt against proceeding, a notification



DOI : https://doi.org/10.25139/inform.v9i1.4741

appears: "Installment Payment Failed to Be Processed", redirecting them to the main page. If the payment is pursued, the system creates a payment confirmation page, where members are prompted to enter their password and click "ok". Incorrect password entries trigger a notification: "The Password You Entered is Wrong", asking members to re-enter the password. The system provides a receipt upon successful password verification, and the payment data is saved in the member's profile.

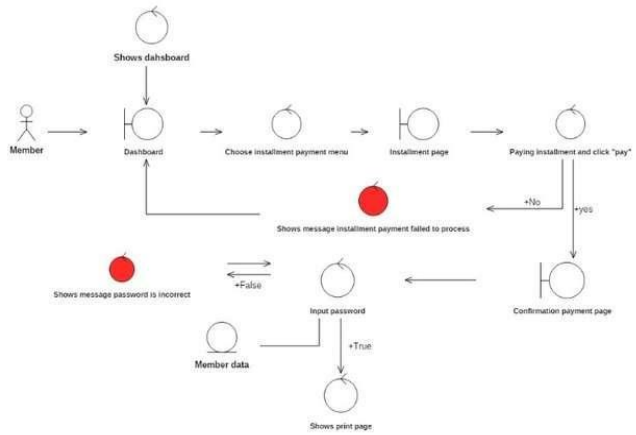


Figure 15. Robustness Diagram installment payment

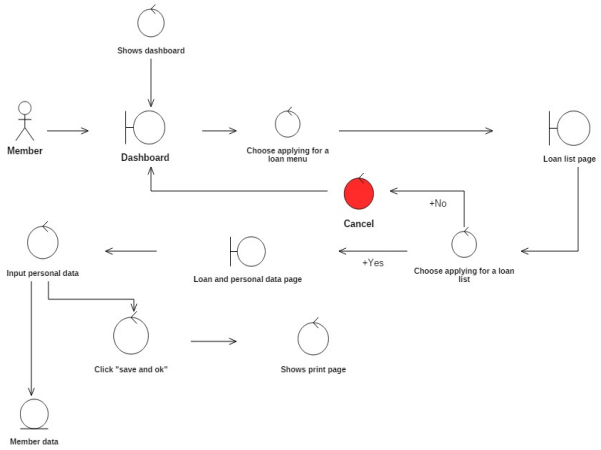


Figure 16. Robustness of Loan Application Diagram

Figure 16 depicts a Robustness Diagram related to Loan Applications initiated by members. Starting from the Initial page, members choose the loan application menu. The system then presents a page listing various loan applications. Members can select their desired loan option. If they abandon the transaction, they can click "cancel". However, if they wish to continue, the system displays a page requesting personal data for the loan application. Once members input their personal information and confirm their choices by clicking "ok", the system saves the data. It generates a printed record, which is subsequently added to the member's profile. Figure 17 is a Robustness Logout Diagram performed by two actors, namely admins and members. The User (Admins and Members) enters the Main Page, then the user presses the Logout button found on the Main Page. After the user presses the Logout button, the system will display the Login page again.

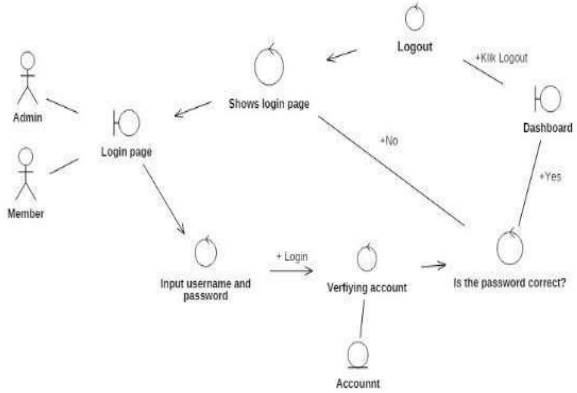


Figure 17. Robustness Diagram Logout

## E. *Detailed Design*

This stage describes the process in detail of the previously created object [17]. This process consists of a sequence diagram and a Class Diagram [18].

*1) Sequence Diagram:* A sequence diagram is a diagram that describes an interaction between objects in the system [4]. The following are sequence diagrams of the analysis and design of a website-based savings and loan cooperative application.

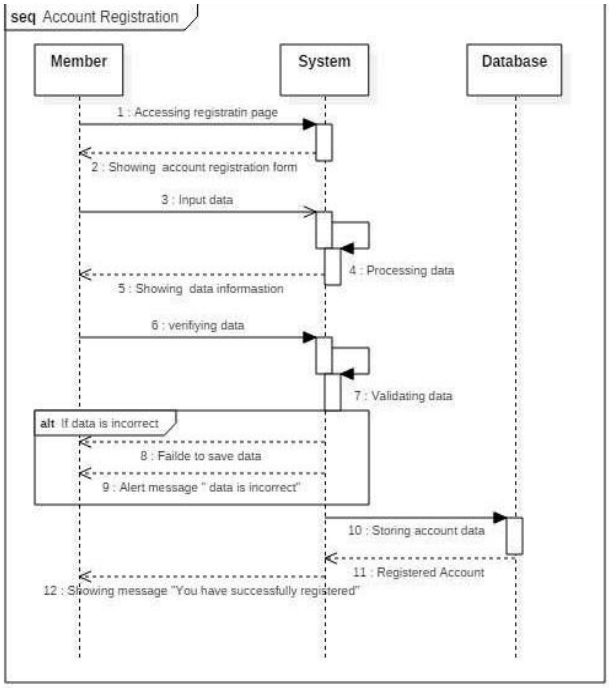


Figure 18. Sequence Diagram of Account Registration

Figure 18 illustrates a Sequence Diagram for Account Registration initiated by members. The member first logs in and accesses the Account Registration Page. The system then displays the registration form, prompting the member to provide the necessary details. Once filled, the system processes



DOI : https://doi.org/10.25139/inform.v9i1.4741

and validates this data. The system will not save the data if the entered information does not adhere to the expected format. Instead, it will display an error message: "The data you entered is inappropriate." However, if the provided data meets the system's criteria, it will be stored in the database, and the account will be registered successfully. The system will then notify the member with an "Account Registration Successful" message.

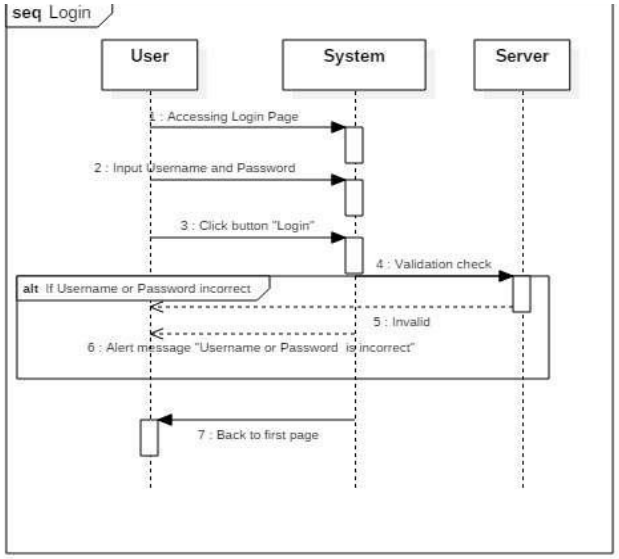


Figure 19. Sequence Diagram Login

Figure 19 showcases a Login Sequence Diagram involving two key actors: the admin and the members. A user, either an admin or a member, begins by accessing the savings and loan cooperative system's web interface, which presents the login page. They input their username and password before clicking the "Login" button. The system then cross-references these credentials with the server. The user is directed to the main page if the provided details are correct. Conversely, an alert notification is displayed if there's a mismatch, stating, "The Username and Password you entered is Wrong!"

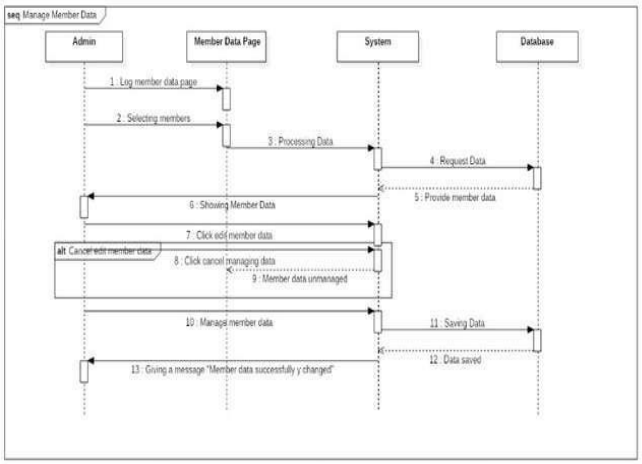


Figure 20. Sequence Diagrams Managing Member Data

Figure 20 illustrates a Sequence Diagram for managing member data executed by the admin. Initially, the admin navigates to the member data page and chooses the member data they wish to modify or manage. Following this, the system processes the request and fetches the relevant data from the database. The system then presents this member data for review. If the admin opts to edit, they can make the necessary changes. However, the member data remains unchanged if the "cancel edit" option is selected. Should the admin finalize their changes, the modified data is stored in the database. Once the update is successful, the admin receives a notification stating that the member data has been modified.

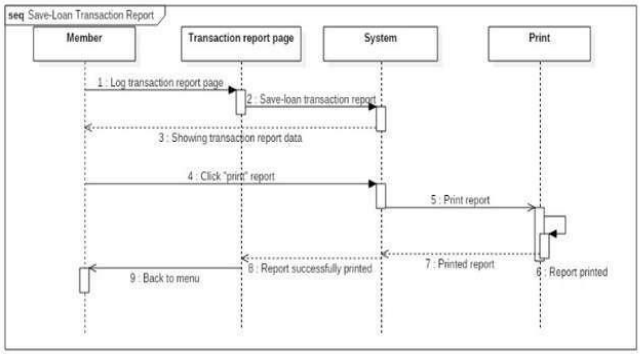


Figure 21. Sequence Diagram of Savings and Loans Transaction Report

Figure 21 illustrates a Sequence Diagram for the Savings and Loan Transaction Report initiated by members. Members begin by accessing the transaction report menu. The system processes the request and presents the savings and loan transaction report data upon selection. Following this, members can select the "print report" button. Once chosen, the system processes the printing task until the report is successfully printed. After the printing process concludes, the user is redirected back to the main menu.

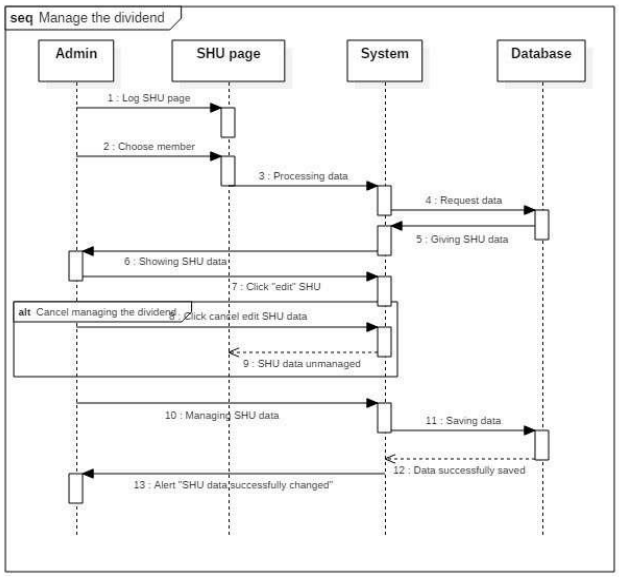


Figure 22. Sequence Diagram Managing SHU Data



DOI : https://doi.org/10.25139/inform.v9i1.4741

Figure 22 depicts a Sequence Diagram detailing admins' management of SHU data. Initially, the admin accesses the SHU page. From there, they can choose specific members whose SHU data needs modification or management. Once selected, the system processes this data and communicates with the database. The database then retrieves the saved SHU data for the selected members, which the system displays. At this stage, the admin can choose to edit the SHU data. If the admin decides to cancel the editing process, the SHU record remains unchanged. However, the updated data gets stored in the database if the admin proceeds with the modifications. Once saved, the admin receives a confirmation message indicating that "the SHU data was successfully changed."

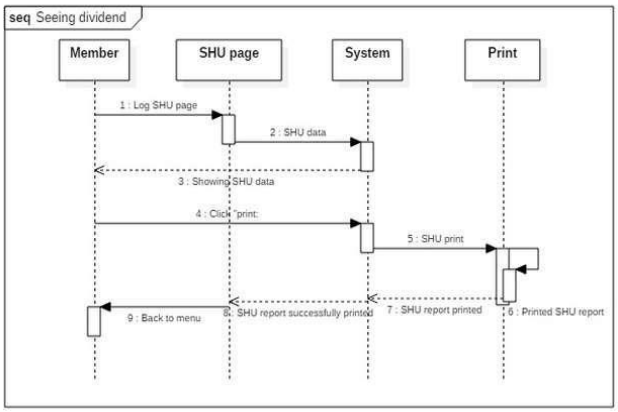


Figure 23. Sequence Diagram Viewing SHU Data

Figure 23 illustrates a Sequence Diagram focusing on the process of members viewing SHU data. Members start by accessing the SHU menu. This action prompts the system to retrieve and display the SHU data. After viewing, members can print the SHU data by selecting the "print SHU" button. The system then processes this request, resulting in the successful printing of the SHU data. Subsequently, members are redirected back to the main menu.

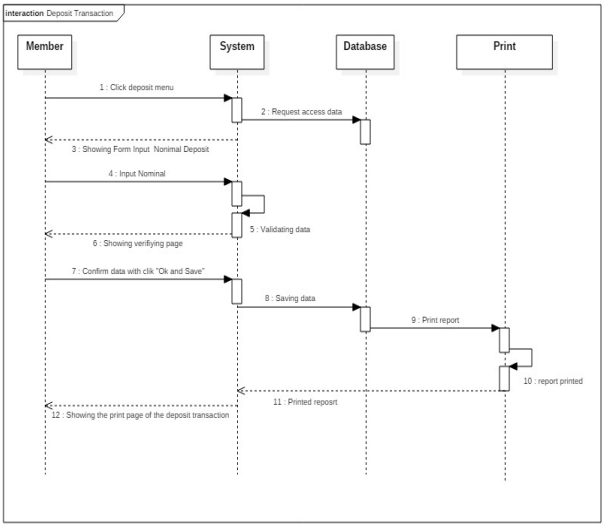


Figure 24. Order Deposit Submission Diagram

Figure 24 illustrates a Sequence Diagram detailing Deposit Submissions executed by members. First, the member selects the Deposit Transaction menu. This prompts the system to access the database, presenting the member with a form to input the deposit amount. Once entered, the system validates this data, subsequently displaying a confirmation page. After reviewing, the member affirms the details by clicking "OK" and "Save." The validated data is then stored in the database, and a print page for the deposit transaction is displayed for the member.

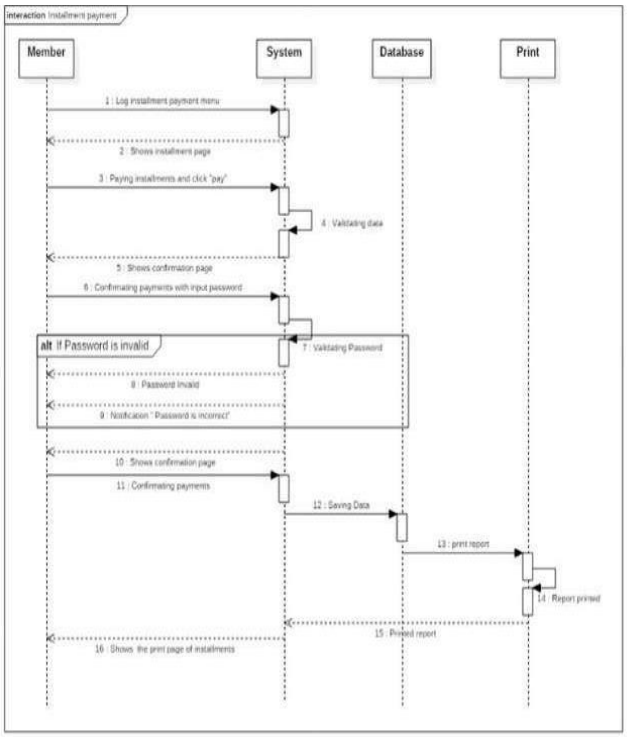


Figure 25. Sequence Diagram of Installment Payment

Figure 25 showcases a Sequence Diagram of Installment Payments undertaken by members. Initiating the process, the member opts for the Installment menu, which leads the system to reveal the Installment Transaction Payment page. The member then enters the installment amount and selects "Pay." The system validates the provided data, presenting a confirmation page. The member must enter a password to approve the payment, which the system verifies. If entered incorrectly, an alert stating "The Password You Entered Is Wrong" is displayed, prompting the member to re-enter the password. Once the correct password is entered, the system showcases another confirmation page. The member finalizes the payment by clicking "OK." This confirmed data is stored in the database, with a corresponding print page for the installment displayed to the member.

Figure 26 illustrates a Sequence Diagram detailing Loan Applications initiated by members. Members begin by selecting the deposit application menu. In response, the system presents a page listing loan application. Members are redirected to a



DOI : https://doi.org/10.25139/inform.v9i1.4741

form requesting personal data after choosing a desired loan application from the list. Upon completion, the system validates this information and showcases a confirmation page. Members can then review and confirm the provided data by selecting "OK" and "Save." Once confirmed, the information is saved in the database. The system then generates a printout of the loan transaction, displaying it for the member's reference.

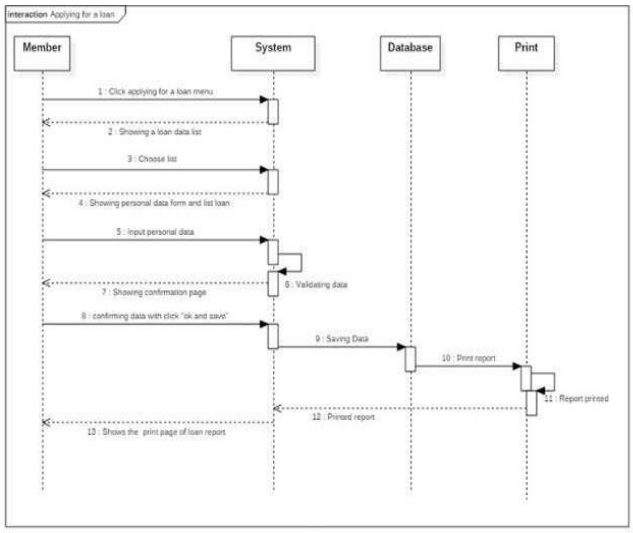


Figure 26. Sequence Diagram of Loan Application

Figure 27 depicts a Sequence Diagram for the logout process, executed by two distinct actors: the Admin and the Member. After logging in and accessing the main page, users can initiate the logout sequence by pressing the "Logout" button. Upon doing so, the system will navigate back to the login page.

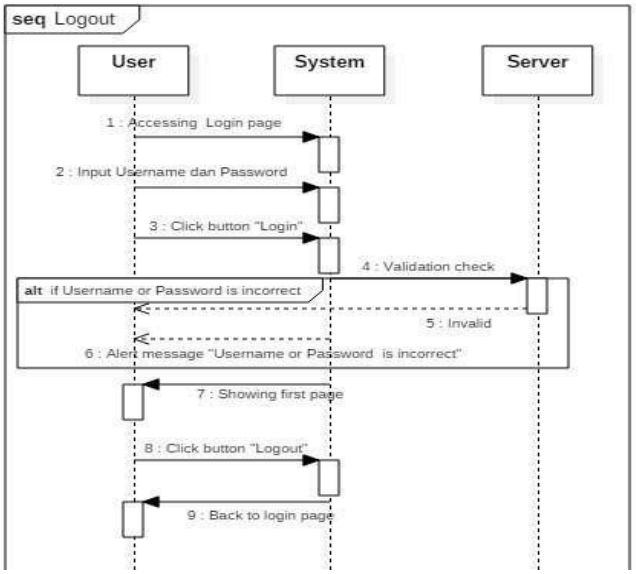


Figure 27. Sequence Diagram Logout

2) *Class Diagram*: UML features a static diagram focusing on system relationships. This class diagram provides a comprehensive application overview, detailing intricate processes in line with the code that must be programmed [18]. One such diagram is illustrated in Figure 28. Figure 28 portrays a Class Diagram with two primary actors: admins and members. There's an "admin account" class for the admin actor, which encompasses the "member data" class. This latter class has operations for saving and changing data. This composition signifies that the "member data" class is integral to the "admin account" class, allowing the admin account to either store or modify member data. Conversely, member data can only be adjusted by the admin. Furthermore, the "admin account" class also encompasses the "SHU" class, which has saving, editing, and printing operations. This implies that the "SHU" class is a part of the "admin account" class, thereby enabling the admin account to save, edit, and print SHU data. In return, the admin can only edit and print SHU data. For the member actors, the "member account" class is connected to the "login" class. This denotes that members must first log in to access the system, and to log in, one must have a member account. The "account registration" class is associated with the "login" class, meaning that one must undergo account registration first to log in. Once logged in, members can't register a new account since the account is already documented. Additionally, the "member's account" class links to the "transaction" class, signifying that the member's account can process transactions, and these transactions log the activities performed by the member. Finally, the "member account" class comprises the "SHU" class, indicating that the SHU class is a component of the member's account. This ensures that the member's account receives an SHU report, which the member can then access.

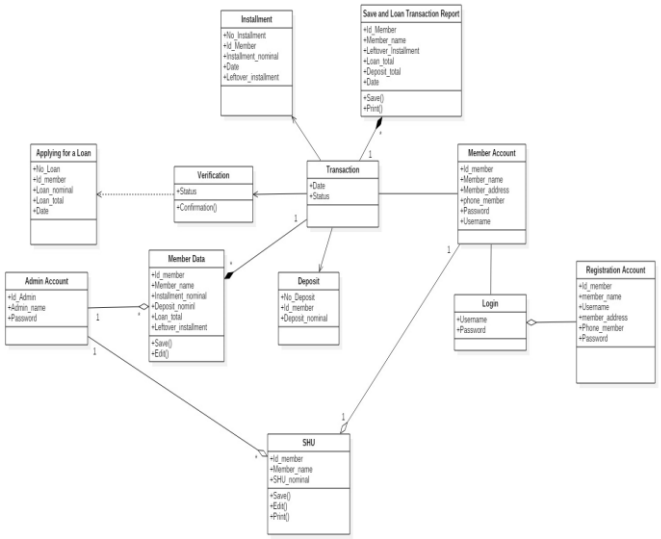


Figure 28. Class Diagram

The "transaction" class encompasses the "savings and loans transaction report" class, which includes "savings" and "print" operations. This indicates that the "savings and loans transaction report" is an integral component of the "transaction" class. Every transaction is recorded in the "savings and loan transaction report." Conversely, a "savings and loan transaction report" only exists when a transaction has been made. The "transaction" class is also linked with the "installment,"



DOI : https://doi.org/10.25139/inform.v9i1.4741

"verification," and "deposit" classes. This signifies that when transactions, such as installments, verifications, and deposits, are made, they are processed and recorded directly. These processes are unidirectional and passive, meaning they do not influence or act upon the "transaction" class in return.
Furthermore, the "verification" class has operations for "confirmation" and is dependent on the "Loan Application" class. This dependency suggests that to apply for a loan, one must first undergo and pass the verification process. Lastly, the "transaction" class comprises the "member data" class. This means that all transaction data is input and logged into the "member data" class. Reciprocally, the "member data" class houses a transaction data collection.

### *F. Implementation*

It is a stage for creating program code from developing a system that has been created and carried out. There is also a testing stage for making the program code. However, it has not been carried out at this stage because the research focuses on analyzing and designing applications only.

## IV. CONCLUSION

Several conclusions can be drawn based on the analysis and design conducted using the Iconix process for the cooperative savings and loan application at Hayam Wuruk University Surabaya. During the needs analysis stage, the system identified functional requirements that involve two primary actors: admins and members. Additionally, non-functional requirements were determined, encompassing areas like security, usability, reliability, operational capabilities, and behavioral demands. This led to the design of the application's GUI, a domain model, and a use case diagram. The preliminary design stage bridged the gap between the needs analysis and system design, yielding ten robustness diagrams. These diagrams represented processes like account registration, login, member data management by admins, savings, and loan transaction reports, SHU data management by admins, SHU data viewing by members, deposit filings, installment payments, loan applications, and logout. The detailed design stage achieved a more intricate understanding of the created objects. This stage produced ten sequence diagrams and a class diagram, paralleling the processes outlined in the robustness diagrams. As a recommendation stemming from this research, the Savings and Loans Cooperative of Hayam Wuruk University Surabaya must consider this analysis and design. The hope is that the design, especially in the SHU section, can be further developed into a more comprehensive program. Additionally, it would be beneficial to provide training to admins to ensure they can adeptly navigate and operate the system, preventing potential operational issues.



## REFERENCES

[1] F. Marwiyati, "Peran Koperasi Simpan Pinjam dalam Upaya Pengembangan Usaha Mikro Kecil dan Menengah (UMKM) di Kabupaten Aceh Barat ( Studi Kasus pada Koperasi Simpan Pinjam ( KSP ) Mandiri Jaya Meulaboh )," *J. Ilm. Basis*, pp. 72–90, 2020.
[2] A. Nugroho, R. Rachmatullah, and H. Prabandara, "Koperasi Simpan Pinjam Berbasis Web Pada Koperasi Subur Surakarta," *Go Infotech J. Ilm. STMIK AUB*, vol. 24, no. 2, p. 74, Dec. 2018.
[3] M. R. Vicky and L. Septiana, "PERANCANGAN SISTEM INFORMASI KOPERASI SIMPAN PINJAM BERBASIS WEB PADA KOPERASI HUTAN LESTARI JAKARTA," *J. Manaj. Inform. Jayakarta*, vol. 1, no. 3, pp. 208–219, 2021.
[4] A. Pratama, S. Mukaromah, S. A. Ithriah, and E. M. Safitri, "Entrepreneurship Information System Design with ICONIX Process for a Student Business Unit Marketplace," *J. Phys. Conf. Ser.*, vol. 1569, no. 2, 2020.
[5] A. Mubarak, "Rancang Bangun Aplikasi Web Sekolah Menggunakan Uml (Unified Modeling Language) Dan Bahasa Pemrograman Php (Php Hypertext Preprocessor) Berorientasi Objek," *JIKO (Jurnal Inform. dan Komputer)*, vol. 2, no. 1, pp. 19–25, 2019.
[6] A. Ristyawan *et al.*, "Proses Iconix Dalam Analisa Rancangan Aplikasi Informasi," *J. SIMETRIS*, vol. 10, no. 1, pp. 1–14, 2019.
[7] D. A. Budi, "Perancangan Sistem Login pada Aplikasi Berbasis GUI Menggunakan Qtdesigner Python," *J. SIMADA (Sistem Inf. dan Manaj. Basis Data)*, vol. 4, no. 2, pp. 92–100, 2021.
[8] A. Ramadani, "Program Studi Informatika," *Publikasi*, 2020.
[9] T. A. Kurniawan, "Pemodelan Use Case (UML): Evaluasi Terhadap beberapa Kesalahan dalam Praktik," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 1, p. 77, 2018.
[10] S. Kurniawan, T. Bayu, "Perancangan Sistem Aplikasi Pemesanan Makanan dan Minuman Pada Cafetaria NO Caffe di TAnjung Balai Karimun Menggunakan Bahasa Pemrograman PHP dan My.SQL," *J. Chem. Inf. Model.*, vol. 53, no. 9, pp. 1689–1699, 2020.
[11] A. A. Wicaksono and D. A. P. Putri, "SISTEM PENGELOLAAN KEUANGAN KEGIATAN PKK DI DESA PENGKOL," *J. Ilm. "Technologia,"* vol. 12, no. 1, p. 59, 2021.
[12] N. N. K. Sari and Widiatry, "Rancang Bangun Website untuk Memeriksa Plagiat E-Journal Fakultas Kedokteran Universitas Palangka Raya Cite this paper Related papers," *J. CoreIT*, vol. 5, no. 2, pp. 36–43, 2019.
[13] S. E. Zaluchu, "Strategi Penelitian Kualitatif dan Kuantitatif Di Dalam Penelitian Agama," *Evang. J. Teol. Injili dan Pembin. Warga Jemaat*, vol. 4, no. 1, p. 28, 2020.
[14] A. Nugroho, "Perancangan Aplikasi Pemesanan Makanan dan Minuman Berbasis ERP (Open ERP) di Cafe Cozy," *J. Ris. dan Apl. Mhs. Inform.*, vol. 2, no. 02, pp. 356–363, 2021.
[15] L. N. Safitri, S. A. Wicaksono, and M. C. Saputra, "Analisis dan Perancangan Sistem Informasi Manajemen Pusat Laktasi : Lactashare," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 6, pp. 2286–2294, 2018.
[16] S. Suci Romadhon and Desmulyanti, "Perancangan Website Sistem Informassi Simpan Pinjam Menggunakan Framework Codeiginter pada Koperasi Bumi Sejahtera Jakarta," *J. Inf. Syst. Informatics Comput.*, vol. 3, pp. 21–27, 2019.
[17] Ismanto, F. Hidayah, and Kristinanti, "Pemodelan Proses Bisnis Menggunakan Business Process Modelling Notation (BPMN) (Studi Kasus Unit Penelitian dan Pengabdian Kepada Masyarakat (P2KM) Akademi Komunitas Negeri Putra Sang Fajar Blitar)," *BRILIANT J. Ris. dan Konseptual*, vol. 5, no. 1, 2020.
[18] A. V. Vitianingsih, D. Cahyono, and A. Choiron, "Analysis and design of web-geographic information system for tropical diseases-prone areas: A case study of East Java Province, Indonesia," in *2017 4th International Conference on Information Technology, Computer, and Electrical Engineering (ICITACEE)*, 2017, pp. 255–260.



DOI : https://doi.org/10.25139/inform.v9i1.4741