



**COMMUNITY FUND ADVISOR (CFA)**

By  
GROUP BSE 22-02  
WEB CLOUDBASED SYSTEM  
DEPARTMENT OF NETWORKS  
SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY



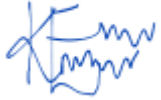
A Project Report Submitted to the School of Computing and Informatics Technology  
for the Study Leading to a Project in Partial Fulfillment of the  
Requirements for the Award of the Degree of Bachelor of  
Science in Software Engineering of Makerere University.

Supervisor  
DR Alex Mwotil  
Department of Networks  
School of Computing and Informatics Technology, Makerere University  
mwotila@gmail.com, +256-772-084725

September, 2022.

## Declaration

We, group BSE 22-2, hereby declare that the work presented is original and has never been submitted for an award to any university or institution of higher learning.

#	NAME	STUDENT NO.	REGISTRATION NO.	SIGNATURE
1	NAJJUMA DIANA	1800723408	18/U/23408/EVE	
2	KIKONYOGO KENNETH TREY	1800742094	18/U/42094/PS	
3	KALUNGI EDWARD	1800723419	18/U/23419/EVE	

# Approval

This project report titled “Community Fund Advisory” has been submitted for examination with my approval as the supervisor of group BSE 22-06.


DR Alex Mwotil

Department of Networks

School of Computing and Informatics Technology;

College of Computing and Information Sciences,

Makerere University

Signature: ...  .....

Date: .....25/09/22.....

Supervisor

# Dedication

We dedicate this work to our parents, friends, and our supervisor, for the great support they exhibited to us throughout the stages of the project, it wouldn't have been possible without them.

# Acknowledgements

First and foremost, we would like to thank God who has granted us the opportunity to complete this project with success and for blessing us with life, good health and the continuous financial provision throughout the four years of this course till this stage.

We would like to thank **Dr. Alex Mwotil**, our supervisor, whose valuable guidance has been the one that helped us patch this project and make it full proof success. Her suggestions and instructions served as the major contributors towards the completion of the project.

Our thanks and appreciation also go to our parents, guardians and friends for providing us with financial and moral support throughout the course of this program and project.

Finally, we thank our fellow students who, in conjunction, contributed in developing the project, they willingly helped us with their abilities right from requirements gathering as well as updating us about the relevant class information.

# Abstract

From 1986 upto today when Uganda as a country has had a stable political and economic environment, sustainable economic growth is still a challenge for the government. Several initiatives such as “Mbona Mbagagawale”, ”Emyooga” have been introduced to boost incomes to the lowest administrative units. Results achieved are not satisfying enough despite the huge funds utilized.

We believe that economic growth and development at the lowest administrative units requires much more than just provision of funds. It requires tracking, monitoring and analysis of all financial groups to guide the financial decisions.

## Table of Contents

<b><i>Declaration</i></b> .....	<b><i>ii</i></b>
<b><i>Approval</i></b> .....	<b><i>iii</i></b>
<b><i>Dedication</i></b> .....	<b><i>iv</i></b>
<b><i>Acknowledgements</i></b> .....	<b><i>v</i></b>
<b><i>Abstract</i></b> .....	<b><i>vi</i></b>
<b><i>SOFTWARE DESIGN DOCUMENT</i></b> .....	<b><i>9</i></b>
<b><i>SYSTEM IMPLEMENTATION, TESTING AND VALIDATION REPORT</i></b> .....	<b><i>36</i></b>
<b><i>USER MANUAL FOR COMMUNITY FUND ADVISOR SYSTEM</i></b> .....	<b><i>68</i></b>





# **SOFTWARE DESIGN DOCUMENT**

## Table of Contents

<b><i>SOFTWARE DESIGN DOCUMENT</i></b> .....	<i>Error! Bookmark not defined.</i>
<b><i>Table of content</i></b> .....	<i>Error! Bookmark not defined.</i>
<b><i>List of Figures</i></b> .....	<b><i>xii</i></b>
<b><i>List of Tables</i></b> .....	<b><i>xiii</i></b>
<b>1. INTRODUCTION</b> .....	<b>14</b>
<b>1.1 Purpose</b> .....	<b>14</b>
<b>1.2 Scope</b> .....	<b>14</b>
<b>1.3 Overview</b> .....	<b>15</b>
<b>1.4 Reference Material</b> .....	<b>16</b>
<b>1.5 Definitions and Acronyms</b> .....	<b>16</b>
<b>2. SYSTEM OVERVIEW</b> .....	<b>17</b>
<b>3. SYSTEM ARCHITECTURE</b> .....	<b>18</b>
<b>3.1 Architectural Design</b> .....	<b>18</b>
<b>3.2 Decomposition Description</b> .....	<b>19</b>
<b>3.3 Design Rationale</b> .....	<b>20</b>
<b>4. DATA DESIGN</b> .....	<b>21</b>
<b>4.1 Data Description</b> .....	<b>21</b>
<b>4.2 Data Dictionary</b> .....	<b>23</b>
<b>5. COMPONENT DESIGN</b> .....	<b>27</b>
<b>5.1. Automatic Fund Allocator component</b> .....	<b>28</b>
<b>5.2. Savings Module component</b> .....	<b>28</b>
<b>5.3. Loan Management component</b> .....	<b>29</b>
<b>5.4. VSLA management module</b> .....	<b>29</b>
<b>5.5. User level data access component</b> .....	<b>30</b>
<b>5.6. Tracking the typing patterns of the user component</b> .....	<b>30</b>
<b>6. HUMAN INTERFACE DESIGN</b> .....	<b>31</b>
<b>6.1 Overview of User Interface</b> .....	<b>31</b>
<b>6.2 Screen Images</b> .....	<b>31</b>
<b>6.3 Screen Objects and Actions</b> .....	<b>32</b>

<b>7. REQUIREMENTS MATRIX .....</b>	<b>33</b>
<b>8. APPENDICES.....</b>	<b>35</b>

## List of Figures

Figure 3. 1: Client-Server Architecture of CFA .....	18
Figure 3. 2: Sequence diagram for the loan module .....	20
Figure 4. 1: Entity Relation Diagram for the CFA system database .....	22
Figure 5. 1: CFA component diagram .....	27
Figure 6. 1: Log into the system .....	31
Figure 6. 2: Create or add new user with role in the system.....	32
Figure 6. 3: TRO's dashboard for managing the various VSLAs.....	32

## List of Tables

Table 4. 1: List of entities and attributes.....	21
Table 4. 2: User entity for inserting new user.....	23
Table 4. 3: Role entity for the system roles .....	23
Table 4. 4: VSLA entity for inserting new VSLA .....	23
Table 4. 5: Loan entity .....	24
Table 4. 6: Creditor entity .....	24
Table 4. 7: Budget entity.....	25
Table 4. 8: VSLA entity.....	25
Table 4. 9: Share out entity .....	26
Table 6. 1 .....	32
Table 6. 2 .....	33
Table 7. 1: A requirement Traceability Matrix .....	33

# 1. INTRODUCTION

## 1.1 Purpose

The purpose of this software design document is to show how the CFA software system will be structured to satisfy the requirements. The document specifies the architecture and system design of CFA. The CFA system will be used for automatic fund allocation, portfolio rebalancing, loan optimization, and information management of Village Savings and Loan Associations (VSLAs). The system shall be cloud-based, and shall be used by the Trade Representation Officer (TRO) and the VSLAs. The intended audience for the document includes; system developers, system testers, project managers, system maintainers, and Kampala Capital City Authority (“KCCA”) staff.

## 1.2 Scope

The system shall provide a secure and responsive online version of the CFA. The project will cover features including Robo-advisory, intelligent portfolios, loan optimization and access control through user roles.

The objectives of the project are as follows:

1. To design a fund allocation algorithm that uses respective user data to make automatic decisions.
2. To design an algorithm that builds a portfolio for each of the members based on their provided personal information, balances, savings and credits.
3. To implement a loan module that stores information about loans and issues financial cards to creditors.
4. To implement user role-based access to the system data in order to enforce the principle of least privilege.
5. To design a two-factor authentication mechanism to make it harder for identity theft.

The system shall offer the following benefits to the TRO:

1. TRO will use the system to track transactions performed on their data to establish the source of errors as a result of data input and modification.
2. The data shall be protected from malicious people since it will be encrypted and any authorized person who gains access to it will not be able to make sense of it.
3. The system will allow only authorized access to data and so this will protect data from being accessed by untrusted users.
4. The system shall allow only authorized users to access data, which will protect data against any untrusted modifications.
5. The system will log and keep track of user activities to ensure data integrity. In case of any changes to the data, the log will keep the user, tasks done by that user, time of execution, and deliverable of that action.

## **1.3 Overview**

The document has six sections:

The first section is an introduction, which contains the purpose of the document, scope, reference material, definitions, and acronyms.

The second section gives the system overview, which describes the system being proposed.

Section three is the system architecture which has the architectural design and decomposition description of the system.

Section four is the data design, which has a data description, dictionary and the component design of the proposed system.

Section five gives the human interface design which contains an overview of the human interfaces, screen images, screen objects and actions of the system intended to be developed.

Section six is the requirements matrix which is used to trace system requirements.

## **1.4 Reference Material**

Forbes “Best Robo-advisors Of June 2022,” 2022 [Online] Available: <https://www.forbes.com/advisor/investing/best-robot-advisors/#:~:text=Most%20robo%2Dadvisors%20have%20low,help%20clients%20with%20investment%20planning> [Accessed: 19-June-2022].

## **1.5 Definitions and Acronyms**

SDD: Software Design Document

KCCA: Kampala Capital City Authority

FinTech: Financial Technology

CFA: Community Fund Advisor

VSLA: Village Savings and Loan Association



## **2. SYSTEM OVERVIEW**

CFA is a Uganda-based FinTech that will allow community members to save, and access credit in their respective VSLAs, and also allow tracking, assessment, and monitoring from KCCA division authorities. CFA will have four primary users namely TRO, Chairman, Ordinary member, and administrator with distinguished roles.

The CFA will be designed to replace the existing manual system(s) that do book-keeping and storage of hard copy files at insecure locations and moved whenever required. The TROs have no clear information to base on when recommending a group for the fund. With all these needs, the main purpose is to improve the efficiency of the existing manual processes.

The proposed system will comprise two main modules namely the VSLA management information system and the TRO's Administrative system. The purpose of the VSLA Management Information system is to store member portfolios, group reports, group activity details, and transactions (Loans, savings withdraws). On the other hand, the purpose of the TRO's Administrative system is to carry out reporting and monitoring to provide valuable and timely decision support to the TRO for effective actions.

# 3. SYSTEM ARCHITECTURE

## 3.1 Architectural Design

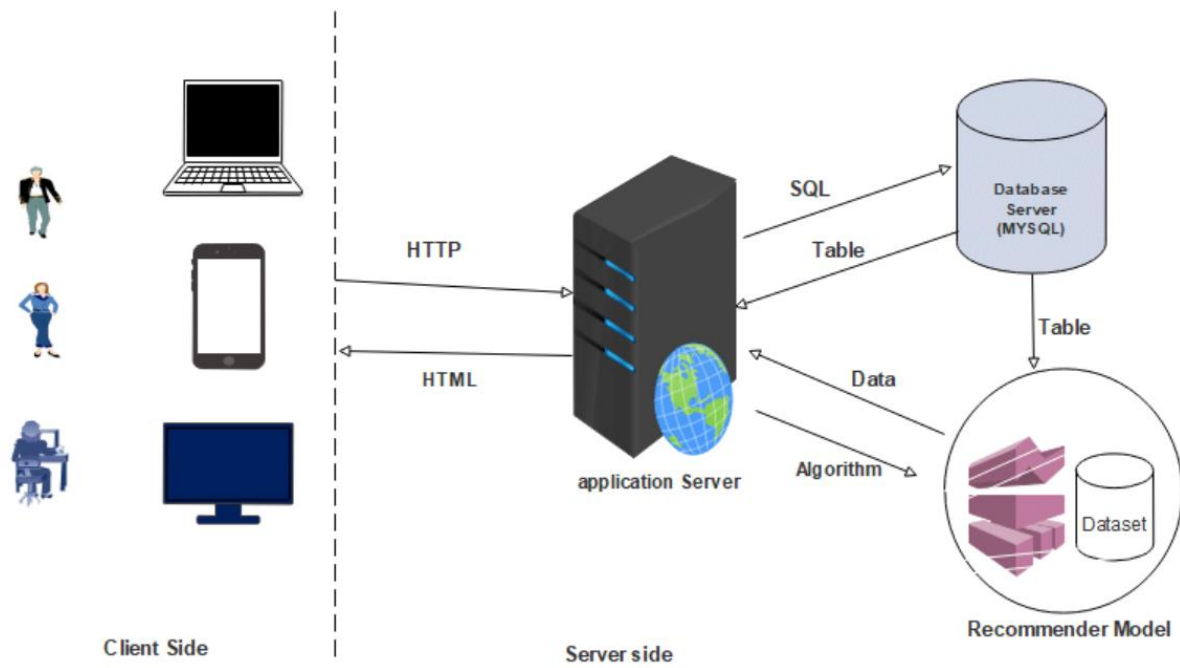


Figure 3. 1: *Client-Server Architecture of CFA*

**1) Client Side:** The CFA system involves participants that need to access and retrieve various data from various locations and devices. The advantage of web applications can be used to make the availability of the services throughout the country. Besides the scalability of the web applications on the front end, a number of users can be served at the same time from the same or different locations. This issue depends on the capacity of the server, speed of the internet and processing speed of the computer.

**2) Application Server:** Web Services expose a service interface to which all inbound messages are sent. The definition of the set of messages that must be exchanged with a service, in order for the service to perform a specific business task, constitutes a contract. The very important role of this tier is to accept the request from the client in terms of HTTP, process and pass the SQL query to the database server. Then by accepting a table in return, process and pass HTML form of information for the client. Also it can work being an intermediate of database server and recommender model (See the above figure - Architecture)

**3) Recommender Model:** In the middle tier there is a machine learning model placed called a recommender model. The recommender model accepts the algorithm passed from the application layer (web server) and processes accordingly. To work on the algorithm, it passes some data to the application layer in return so that the application layer sends a SQL to the database server, such that the database server passes a table to the recommender model. Using the received table, it passes a processed data to the application layer, which is a recommendation of a hotel.

**4) Database server:** The back-end tier is the database server where the VSLA information, member information and other system users' information will be stored. In this layer the different data that are collected via registration, request referral data, feedback data and different data are stored and used for retrieval.

## **3.2 Decomposition Description**

The system shall be composed of the following modules.

### **TRO**

Automatic Fund Allocator: -

The system handles budget allocation to the different VSLAs depending on the appropriated financial budget of the Division.

The system provides VSLA financial profiles without any operational hassles and paperwork from the TRO's end.

VSLAs Management System: -

The system will enable collection and storage of VSLA information from the time of registration to editing of that information.

Reporting module: - The system will be configured to have all the quantitative and qualitative Reports that can be accessed anytime from the system.

### **VSLAs**

Intelligent Portfolio: -

Monitors VSLA portfolio from time to time automatically updates it as needed. It does this based on the personal information provided by the member.

Loans Module: -

The system will support loan processing to a particular member belonging to a certain VSLA and provide a repayment plan.

Savings module: The systems will enable the members to keep track of their savings status.

User access control module: The system will authenticate users to ensure that only authorized users access the system.

Data protection module: The system will validate, encrypt and decrypt all the data.

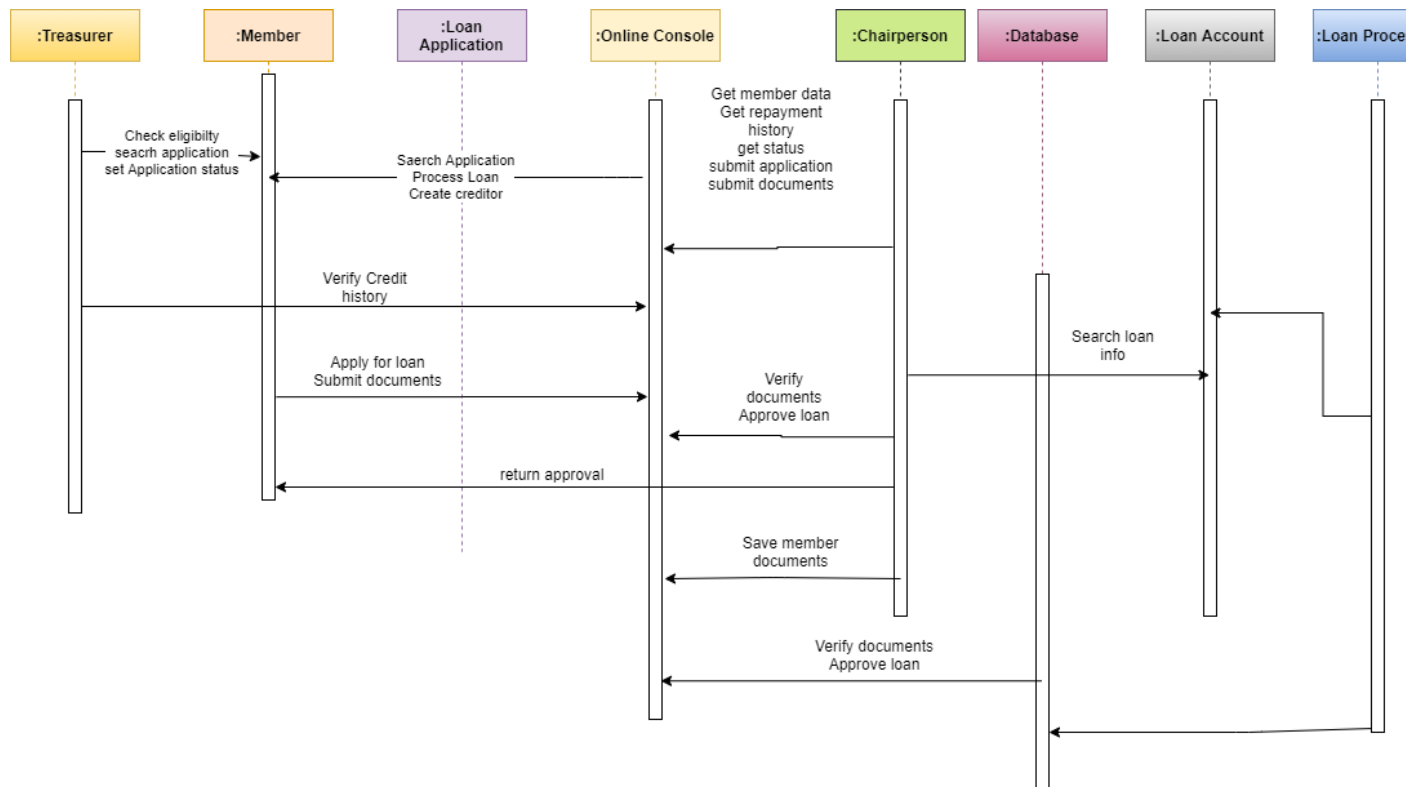


Figure 3. 2: Sequence diagram for the loan module

### 3.3 Design Rationale

Our design is solely based on the client-server architecture because its centralized architecture helps make it easier to protect data with access controls that are enforced by security policies.

The different users (chairperson, administrator, TRO) (remote processors) request and receive service from a centralized server (host computer) and the user's interface then displays the results the server returns.

## 4. DATA DESIGN

### 4.1 Data Description

Since the CFA is an information-based system, the users of the system are expected to create, manipulate, update and delete data. This therefore gives a description of the data structures of the system.

Table 4. 1: List of entities and attributes

Entity	Description	Attributes(s)
User	A user is any person involved in any activity related to the CFA.	<ul style="list-style-type: none"> <li>● id (primary key)</li> <li>● firstName</li> <li>● lastName</li> <li>● email</li> <li>● phone_number</li> <li>● pin</li> <li>● role_name (foreign key)</li> </ul>
Role	Users on the system will have different roles for example TRO, Chairman, member, admin and super-admin.	<ul style="list-style-type: none"> <li>● id (primary key)</li> <li>● role_name</li> <li>● permissions</li> </ul>
VSLA	A savings and loan group or association to which members belong. It is led by a chairperson.	<ul style="list-style-type: none"> <li>● id (primary key)</li> <li>● user_id</li> <li>● vsla_name</li> <li>● capacity</li> <li>● Location</li> <li>● status</li> </ul>
Loan	Amount of money borrowed by a member from the VSLA, its purpose and time references.	<ul style="list-style-type: none"> <li>● id (primary key)</li> <li>● user_id</li> <li>● VSLA_id (foreign key)</li> <li>● loan_purpose</li> <li>● Amount taken</li> </ul>

		<ul style="list-style-type: none"> <li>● Interest</li> <li>● Total amount payable</li> <li>● Date Taken</li> </ul>
Savings	Amount accumulated by a particular VSLA member over time	<ul style="list-style-type: none"> <li>● id(primary key)</li> <li>● VSLA_id (foreign key)</li> <li>● Amount</li> <li>● Date</li> </ul>
Receipt	A financial document showing a member's balances on their account.	<ul style="list-style-type: none"> <li>● id (primary key)</li> <li>● receipt</li> <li>● date</li> </ul>
ShareOut	Amount of money each member receives at the end of the year when finally withdrawing their savings.	<ul style="list-style-type: none"> <li>● id (primary key)</li> <li>● amount</li> <li>● rate</li> <li>● user_id</li> </ul>

Described here is the database structure composed of the user, role, VSLA loan, creditor, card, receipt and share\_out tables. Fields indicated PK are primary keys for each table and fields marked FK are foreign keys.

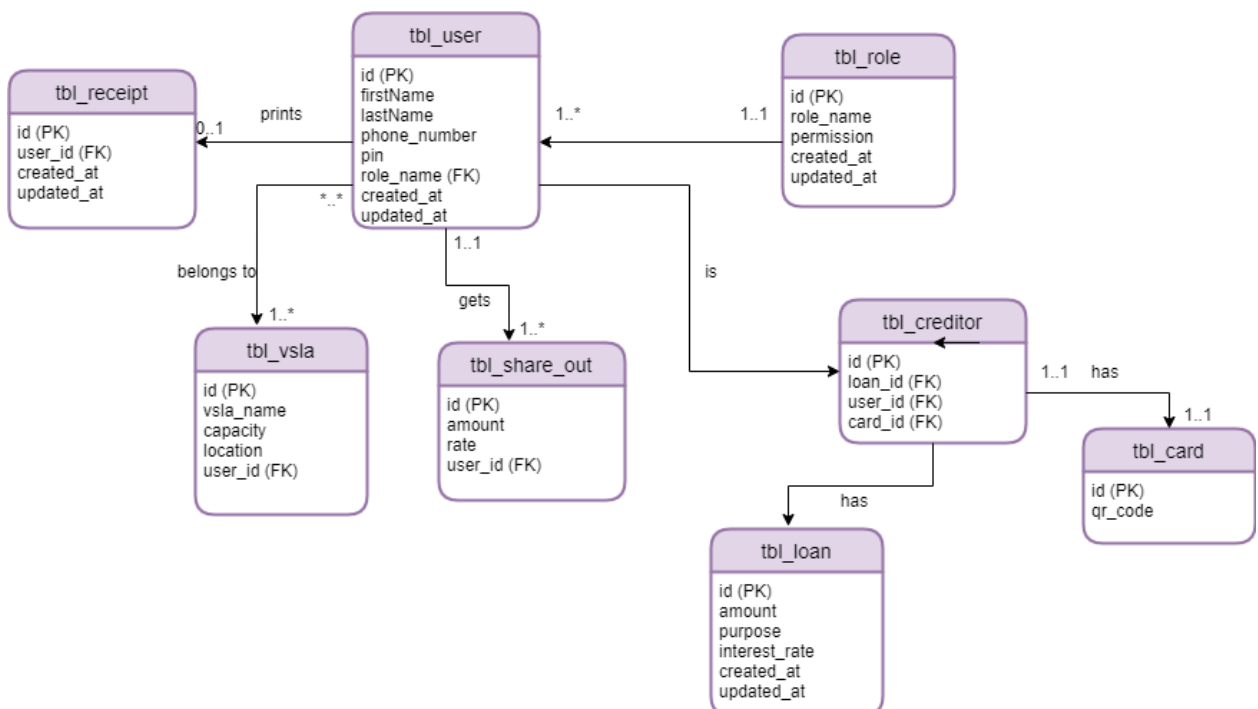


Figure 4. 1: Entity Relation Diagram for the CFA system database

## 4.2 Data Dictionary

Table 4. 2: User entity for inserting new user

Field Name	Data type	Field size for display	Description
id	integer	11	Unique Identifier for each user
firstName	varchar	25	User's first name
lastName	varchar	25	User's last name
phone_number	integer	15	User's phone number / contact
pin	integer	4	Pin for the user
role_name	varchar	15	Unique identifier for each user role
created_at	datetime		Date the user is created
updated_at	datetime		Date the user is updated

Table 4. 3: Role entity for the system roles

Field Name	Data type	Field size for display	Description
id	integer	11	Unique identifier for role
role_name	varchar	25	User's role
created_at	datetime		Date the role is created
updated_at	datetime		Date the role is created

Table 4. 4: VSLA entity for inserting new VSLA

Field Name	Data type	Field size for display	Description
id	integer	11	Unique identifier for each VSLA

user_id	integer	11	Unique identifier for each user
vsla_name	varchar	30	Vsla name
location	varchar	20	Where vsla members live and meetings happen.
status	varchar	10	Status of the VSLA(Active or Inactive)
created_at	datetime		Date vsla is created
updated_at	datetime		Date vsla is updated

Table 4. 5: Loan entity

Field Name	Data type	Field size for display	Description
id	integer	11	Unique identifier for each loan
amount	integer	11	Amount of money borrowed
purpose	varchar	255	Use of the loan
creditor_id	integer	11	Unique identifier for creditor
status	Enum ('00','11','10','01')		Status of loan
created_at	datetime		Date loan is created
updated_at	datetime		Date loan is updated

Table 4. 6: Creditor entity

Field Name	Data type	Field size for display	Description
id	integer	11	Unique identifier for the creditor
user_id	integer	11	Unique identifier for user
loan_id	integer	11	Unique identifier for loan



Table 4. 7: Budget entity

Field Name	Data type	Field size for display	Description
id	integer	11	Unique identifier for each card
budget	BLOB	16M	Identifier of the submitted KCCA budget

Table 4. 8: VSLA entity

Field Name	Data type	Field size for display	Description
id	integer	11	Unique identifier for each receipt
vslaName	varchar	25	Name of a particular VSLA
capacity	integer	11	Number of members in a VSLA
location	varchar	25	Description of city location of a VSLA
chairperson	varchar	25	Individual responsible for effective leadership and running of VSLA
status	varchar	11	Current operation status of a VSLA
meeting	varchar	10	Group meeting rate per time
males	integer	25	Number of male individuals in VSLA
females	integer	25	Number of females in the VSLA
Savings	integer	25	Savings accumulated in the previous year
Averageage	integer	25	Average collective age index of a group
Shareouts	integer	25	Amount shared by the entire group inclusive of returned loans

Loans Taken	integer	11	Amount taken by the entire VSLA in a particular year
Loans Returned	integer	11	Amount returned from loans in a particular period
Activity	varchar	25	Economic activity that joins all VSLA members
Division	varchar	255	City division where VSLA belongs

Table 4. 9: Share out entity

Field Name	Data type	Field size for display	Description
id	integer	11	Unique identifier for each share out
amount	integer	11	Amount of share out
rate	integer	11	Interest rate of share out
user_id	integer	11	Unique Identifier for user

## 5. COMPONENT DESIGN

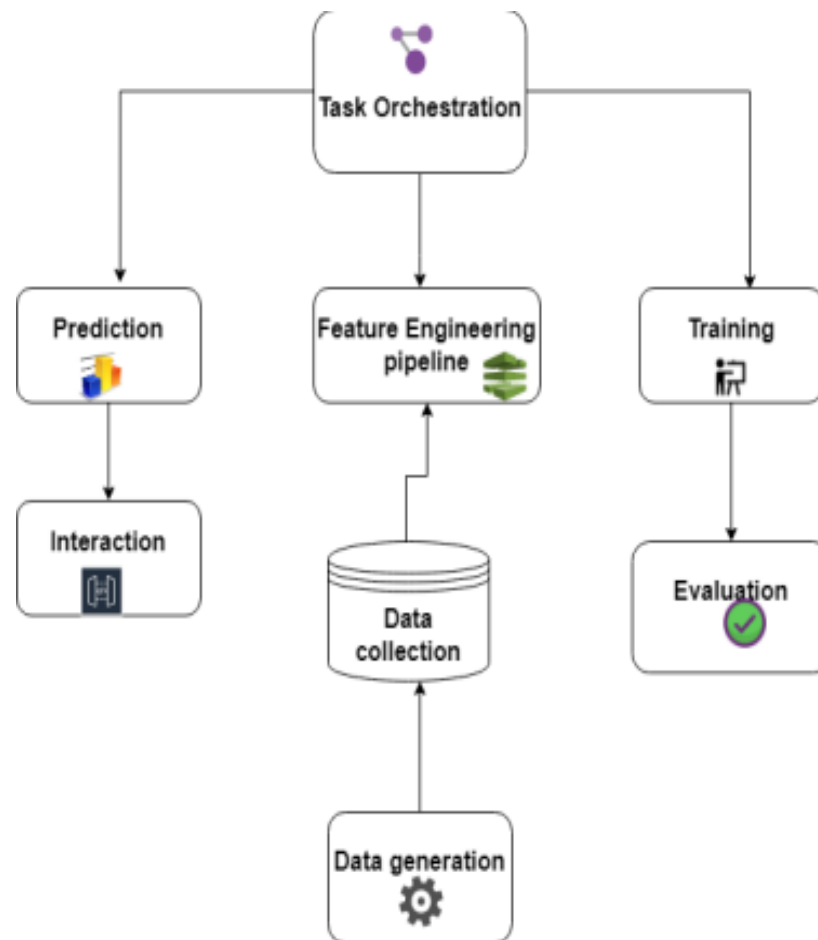


Figure 5. 1: CFA component diagram

1. **Data Generation:** Our Community Fund Advisor lives off data. That data comes from the Trade Representation Officer's module of the system.

2. **Feature Engineering Pipeline:** CFA's algorithm can't make sense of raw data. So, as a team, we wrote a script that selects, transforms/cleans, combines, and otherwise prepares the data so the algorithm can find useful patterns.

3. **Training:** This is where the magic happens. We apply the algorithm, and it learns patterns from the data. Then it uses these patterns to perform the intended task; advising on which Village Savings and Loan Association should receive the fund in a given financial year.

4. **Evaluation:** Here, we need to carefully test how well our algorithm

performs on data it hasn't seen before (during training). This ensures that the prediction model doesn't work well only on "seen" data, but not in real-world settings.

5. **Task Orchestration:** Feature engineering, training, and prediction all need to be scheduled on our computer infrastructure (Google Cloud Platform) – usually with non-trivial interdependence. So we need to reliably orchestrate our tasks.

6. **Prediction:** This is our moneymaker. We use the model we've trained to perform new tasks and solve the TRO's main problem – and this means making a prediction.

7. **Infrastructure:** Even in the age of the cloud, CFA has to live and be served somewhere. We are using GCP for setup and maintenance.

8. **Interaction:** We need some way to interact with our model and give it problems to solve. This is in the form of an API, and a user interface.

## 5.1. Automatic Fund Allocator component.

### Pseudo code;

Check how many VSLAs are registered in the system

Check the savings potential in monetary terms on each VSLA's account

Check the period spent for each VSLA in service //active or deactive

Compare current statistical data about each VSLA

Assigns funds to a VSLA with a leading saving and active potential.

## 5.2. Savings Module component

### Pseudo code;

Check VSLA under which a member is registered.

Check for member account

Input member ID

Retrieve Member saving details

Display details

### **5.3. Loan Management component**

#### **Pseudo code;**

Check if a member is registered under a certain VSLA

If a member is registered

    Check balance on their VSLA account.

    Check if the balance on their account meets the minimum amount on which a loan can be disbursed

        Disburse loan to a member

    Save all information in the database

ELSE

    Members should be advised to make some savings to qualify for a loan.

ELSE

    Redirect member to register under a VSLA

### **5.4. VSLA management module.**

#### **Pseudo code;**

Check all registered VSLAs in the system.

If VSLA is not registered in the system

Register VSLA

Save the details to a database

ELSE

Redirect User to register VSLA

IF VSLA is registered in the system

Edit information about VSLA

Save information in the database

## **5.5. User level data access component.**

### **Pseudo code;**

If a user id logged in

If user role is x

Display specific dashboard(sub-menu) for user role x

ELSE

Redirect the user to the login interface.

## **5.6. Tracking the typing patterns of the user component.**

### **Pseudo code;**

If user is logged in and performs an action on that data

Types in the text field

System monitors the typing speed of the user

The system retrieves the previous typing patterns

The system compares the current and retrieved typing speed patterns

ELSE

System raises a warning to the manager data and marks the user data as suspected.

ELSE

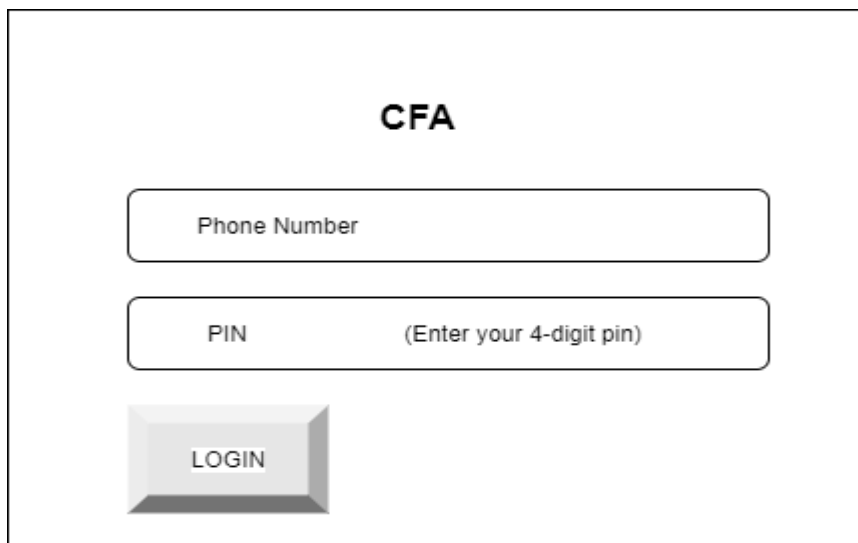
Redirect the user to the login interface.

## 6. HUMAN INTERFACE DESIGN

### 6.1 Overview of User Interface

The system will be a role-based system where the overall administrator is the Data manager. Admin will be responsible for adding new system users and their roles as shown in figure.

### 6.2 Screen Images



The image shows a login screen for a system labeled 'CFA'. At the top center, the text 'CFA' is displayed. Below it, there are two input fields. The first field is labeled 'Phone Number'. The second field is labeled 'PIN' and includes the instruction '(Enter your 4-digit pin)'. Below these fields is a button labeled 'LOGIN'.

Figure 6. 1: Log into the system

**ADD NEW USER**

**User Role** ▾
 

TRO  
 Chairperson  
 Admin

Figure 6. 2: Create or add new user with role in the system.

**TRO**
 Logout

VSLA	CHAIRPERSON	MEETING	CAPACITY	STATUS	LOCATION
Tuli bumu	Enoch Kasajja	Sunday	12	Active	Kamwokya
Kyosiga	Miriam Naluso	Saturday	8	Inactive	Kamwokya
Nussu	Mzee Jeremiah	Sunday	10	Active	Kiira
Cash Cash	Syliva Ahendo	Sunday	10	Unknown	Kamwokya

Figure 6. 3: TRO's dashboard for managing the various VSLAs.

### 6.3 Screen Objects and Actions

Table 6. 1

Screen Object	Action	What happens on performing the action
Phone number	Enter the phone number for the user being created	User phone number stored
User pin	Enter and create the pin the user will use to access the account	The user pin stored



Table 6. 2

User role	The role played by the user in the system. For example, Admin	Display the user role that performed the action
Pin	Enter the pin to access user account	If the pin matches the stored - pin of the users account in the database , access is granted and vice versa.
First name	User name of the person performing the action	Display user name of the person who performed the action
Last name	User name of the person performing the action	Display user name of the person who performed the action

## 7. REQUIREMENTS MATRIX

Table 7. 1: A requirement Traceability Matrix

Business Use Case	Functional Requirements		Priority	Test Case	
	Requirement ID	Description		ID	Description
Capture user details	REQ-1	Authenticate all users before they access the system	High	TC01	Verify if user is authenticated as they access the system
	REQ-2	Logging of all user activities	High	TC02	Verify if all user activities are logged
	REQ-3	Authenticate all admins and super admins with one-time pin code	High	TC03	Verify if the admin and super admin provide correct codes
Capture savings details	REQ-1	Capture member`s savings through USSD code	High	TC04	Verify that the system is able to capture savings to the database through the USSD code
	REQ-2	Capture member`s savings through the mobile app	High	TC05	Verify that the system is able to capture member savings using the mobile cloud based app

	REQ-3	Capture member`s savings through the web app	High	TC06	Verify that the system retrieves and captures savings on the web to and fro the database
Loan Advisory	REQ-1	Assess member`s loan eligibility	Medium	TC07	Verify that the system can determine which member is eligible
	REQ-2	Grant member specific loan amount	High	TC08	Verify that the system provides the given range or specific amount a member deserves
	REQ-3	Track instalment payback of a given member loan	High	TC09	Verify that the system updates member credit history after a given installment is paid
TRO assessment of VSLA	REQ-1	Assess VSLA savings and loan performance	High	TC10	Verify that the system is able to track VSLA savings and loan performance in a given month, quarter, or year
	REQ-2	Rate VSLA performance in the specific division	Medium	TC11	Verify that the system provides in order for the VSLAs performance in the entire division
	REQ-3	Evaluate financial support or funds need	High	TC12	Verify that the system provides poor performing VSLAs in the division
Encryption of user data	REQ-1	Allow user enter raw data	Medium	TC13	Verify if a user provides correct input data
	REQ-2	Encrypt data entered	High	TC14	Verify that the system is able to encrypt data provided by the user
	REQ-3	Save encrypted data to the data	High	TC15	Verify that the encrypted data is saved appropriately

## 8. APPENDICES

### BLOGS

Kenneth Trey Kikonyogo

<https://trey.hashnode.dev/community-fund-advisor-cfa>

Diana Najjuma

<https://communityfundadvisor.hashnode.dev/diananajjuma>

Kalungi Edward

<https://kalungi.hashnode.dev/>

# **SYSTEM IMPLEMENTATION, TESTING AND VALIDATION REPORT**

## Table of Contents

<i>SYSTEM IMPLEMENTATION, TESTING AND VALIDATION REPORT</i> .....	36
<i>Table of content report</i> .....	<b>Error! Bookmark not defined.</b>
<i>List of Tables</i> .....	xxxix
<i>Abbreviations/Acronyms</i> .....	xl
<i>Chapter 1: Introduction</i> .....	41
1.1 Background and scope of the project.....	41
1.2 Overview of the document.....	42
<i>Chapter 2. System Specifications.</i> .....	43
2.1 Version of requirements and Version Control.....	43
2.2 Input.....	43
2.3 Output.....	43
2.4 Functionality .....	44
2.5 Limitations and safety .....	46
2.6 Default settings.....	47
2.7 Special requirements .....	47
2.8 Errors and alarms.....	48
<i>Chapter 3: Design output.</i> .....	52
3.1 Implementation (coding and compilation).....	52
<i>Chapter 4: Inspection and testing</i> .....	55
4.1 Introduction .....	55
4.2 Test plan and performance .....	56
4.2.1 Test objectives .....	56
4.2.2 Scope and Relevance of tests.....	57
4.2.3 Levels of tests.....	58
4.2.4 Types of tests .....	58
4.2.5 Sequence of tests .....	59
4.2.6 Configuration and calculation tests .....	60
<b>4.3 Precautions</b> .....	<b>60</b>
<b>4.3.1 Anomalous conditions</b> .....	<b>60</b>
4.3.2 Precautionary steps taken.....	60
<i>Chapter 5: Installation and system acceptance test.</i> .....	<b>61</b>
<b>5.1 Input files</b> .....	<b>61</b>

5.2 Supplementary files.....	61
<b>5.3 Installation qualification .....</b>	<b>61</b>
<b>Chapter 6: Performance, servicing, maintenance, and phase out. ....</b>	<b>64</b>
6.1 Service and maintenance.....	64
6.2 Performance and Maintenance.....	64
<b>Chapter 7: Conclusion and Recommendations.....</b>	<b>67</b>

## List of Tables

Table 1. 1: Design Details .	54
Table 1. 2: Inspection plan and performance	55
Table 1. 3: Test Table	59
Table 1. 4: Checklist of the Installation and system acceptance test	61
Table 1. 5: Performance and maintenance details	66

## List of Figures

Figure 1. 1: No internet connection	46
Figure 1.2: Invalid login credentials interface	48
Figure 1.3: Invalid email interface	49
Figure 1. 4: Reset password interface	50
Figure 1. 5: User created successfully	50
Figure 1. 6: No record found	51

## **Abbreviations/Acronyms**

VSLAs - Village Savings and Loans Association

CFA - Community Fund Advisor

MFIs - Monetary Financial institutions

SMEs - Small Medium Enterprises

KCCA - Kampala Capital City Authority

MSC - Microfinance Support Center

CDOs - Community Development Office

TRO - Trade Representative Officer

NIN - National Identification Number

NIRA - National Identification and Registration Authority



# Chapter 1: Introduction

## 1.1 Background and scope of the project

The Community Fund Advisor (CFA) system is a software project comprised of two main modules namely the VSLA management information system and the TRO's Administrative system. The purpose of the VSLA Management Information system is to store member portfolios, group reports, group activity details, and finances (Loans, savings withdraws). On the other hand, the purpose of the TRO's Administrative system is to carry out reporting and monitoring in order to provide valuable and timely decision support to the TRO for effective fund allocation.

The Community Fund Advisor (CFA) system is designed to replace the existing manual system that basically does book-keeping and storage of hard copy files that are kept at the group's chairman's home in a metallic suitcase and carried to the venue of the meeting whenever they are needed. Initially, the TRO had no clear information to base on when recommending a group for the fund. With all these needs, the CFA comes in handy to tackle both management and decision-making challenges.

The machine learning component of the CFA system has the ability to automatically improve experience and replicate the cognitive decision-making functions of the TRO through computational statistics. It is based upon an algorithm that selects training data to build a predictive futuristic model.

## 1.2 Overview of the document

This document describes the implementation, testing and validation findings for the CFA system. It is divided into the following section:

Section 1: This section gives an overview of the document

Section 2: This section illuminates the specifications of the CFA system

Section 3: This section particularizes the system design, i.e. coding tools and documentation.

Section 4: This section expounds the inspection and testing of the system.

Section 5: This section describes the installation and acceptance testing.

Section 6: This section elucidates the performance, servicing, maintenance and phase Out.

## **Chapter 2. System Specifications.**

The section describes and specifies the system completely and is the basis for the validation process.

### **2.1 Version of requirements and Version Control**

The requirements specified in the SRS were sufficient and approved. These were followed to design and implement the system. For version control of the code, git has been used and commits are available on group repositories.

### **2.2 Input**

#### **At TRO Level**

VSLA data: This is the major input to the system. It contains raw data entered at KCCA level by the TRO. This data is later used to generate reports, monitor VSLA performance and make cognitive decisions. It includes VSLA address/location, leadership, activities and financial records over time.

#### **At Chairperson Level**

Member data: This data contains individual member details entered at the village level by the chairpersons. It is later used to generate reports.

### **2.3 Output**

Text: This is the data displayed after entry. This data is derived from the database

Cipher text: This the data after encryption

Graphs: This is the data displayed after analysis.

## 2.4 Functionality

### Loan Management

- Loan origination: The borrower applies for a loan and the chairperson processes it after evaluating risks. It provides an individual's credit history to suggest one's credibility for the loan.
- Loan servicing: This feature helps in managing loans since every loan is different; they have different interest rates, plans, collectors and more.
- Debt collection: The system notifies the chairperson when member accounts become delinquent. He also gets notified when the borrower pays back or when a repayment is due. The collection system even calculates late fees for him.
- Reporting and analytics: The system provides comprehensive reports on the cash flow. He can create reports based on interactions with a single member or entire VSLA. Reporting allows him to visualize and understand in which direction his VSLA is moving.

### Member management

- Online registration — users can create accounts for themselves and those below them in hierarchical leadership.

Member profiles — chairperson's account details are held in a centralized database that's updated automatically so chairpersons can easily search for information and contact members if needed

- Mobile optimization — The CFA application is mobile responsive, and therefore allows users to manage activities on the go.

- Messaging — The CFA system allows the TRO and chairpersons to send out general communicational messages or announcements to members in an instant.

### **Decision making**

- This feature of the CFA system analyzes and synthesizes vast amounts of data to assist in decision-making. With this information, it produces reports that may project savings, share outs, or manage VSLAs. Through the integration of multiple variables, the CFA produces a number of different outcomes based on the VSLA's previous data and current inputs.

### **Performance Requirements**

- The system shall support concurrent work of at least 2000 users that are logged on the system.

#### Response time

- The maximum response time for the submission and retrieving of data will be 1 minute.
- The maximum response time for login and logout will be 30 seconds.

#### Scalability

- The system shall be able to accommodate more than 10 users and can be expanded for businesses that have branches (different stores where this system is used) accommodating more than 50 users on a bigger scale depending on the number of branches.

#### Workload

- The system shall be able to accommodate up to 300 transactions a day depending on the speed of the operator.
- The system shall allow more than one user to update the system database at a time.

#### Platform

- The system shall operate on the web platform through the browser for all user devices such as desktop, mobile phone, laptop and all operating systems that allow browser installation..
- The response time shall not exceed 30s if the system is accessed from a local area network and 60s if accessed from a WAN.
- The web server / web portal shall consume not more than 3 Gigabytes of RAM at any point of time. The average memory consumption must not be higher than 1 Gigabytes.
- The client part of the system shall consume not more than 500 Megabytes of RAM at any point of time. The average memory consumption must not be higher than 200 Megabytes.
- The client part of the system shall consume not more than 1 GB of disk space.

## 2.5 Limitations and safety

- ❖ The system requires an active internet connection for all its transactions and in case there is none, a user will be prompted as illustrated below.

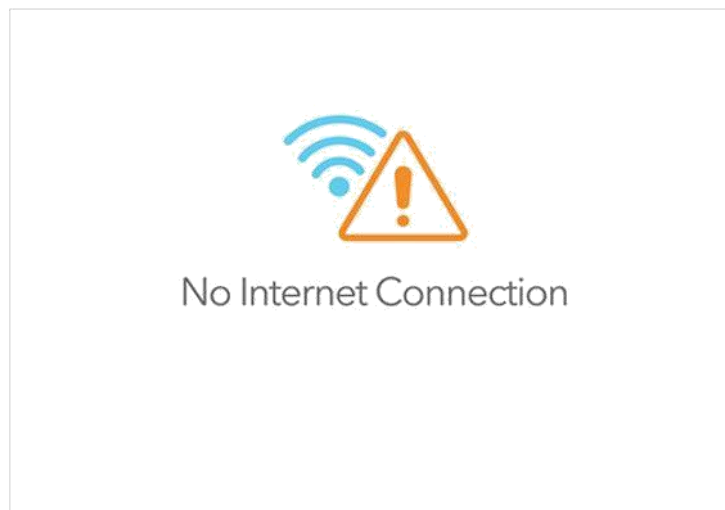


Figure 1. 1: No internet connection

- ❖ The system terminates all inactive user sessions for security purposes after a duration of 5 minutes (300 seconds equivalent).
- ❖ The TRO shall only have rights to add the VSLA locations stating the correct city division, county, sub county and village in order to avoid adding incorrect addresses.
- ❖ The system shall only permit Ugandan citizens to be registered by providing their NIN numbers and a background verification check from NIRA.

## **2.6 Default settings**

By default, when the system is first installed, it contains only the TRO as the primary user for security reasons. The TRO adds VSLA and its respective chairman who later adds the village members.

TRO, chairman and members are provided with default usernames and passwords which **MUST** be altered after first sign-in using the two-factor authentication model for phone number and email.

By default, when the system is first installed, it contains only one user (admin with password admin 123). For security reasons, the password can be changed via the admin dashboard.

## **2.7 Special requirements**

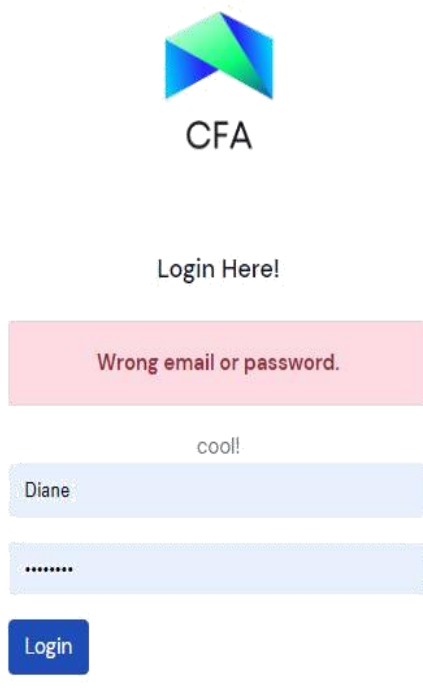
First of all, the CFA developers' team in collaboration with the management of Kampala Capital City Authority (KCCA) financial department is expected to provide the following facilities to ensure the security and confidentiality of the application related equipment.

- Controlling access to the data repository to only the authorized users
- Sharing the application code via version control tools such as GitHub to receive contributions from the other users of the application.

**Data Security:** Since VSLAs utilize their members’ personal data, it is imperative that they provide impeccable data protection. The CFA system provides advanced security features, including authentication, encryption, security protocols and safe cloud storage, so the integrity of critical data is guaranteed.

## 2.8 Errors and alarms

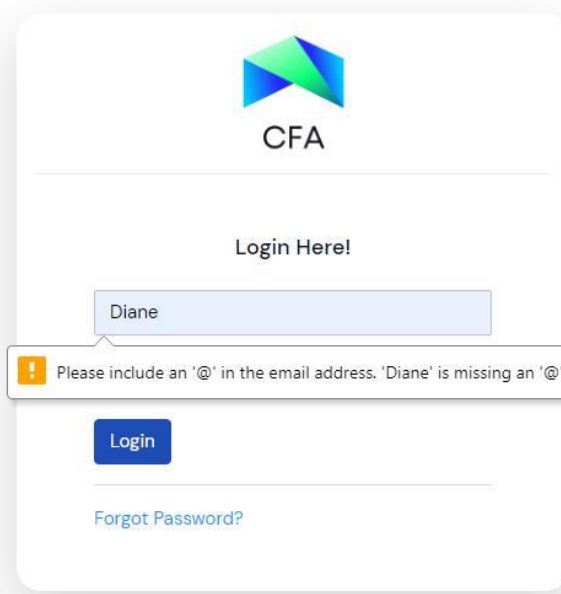
In the process of logging in, if the user logs in with wrong credentials, an error message will be displayed on top of the login form “invalid email or password, try again.”



*Figure 1. 2: Invalid login credentials interface*

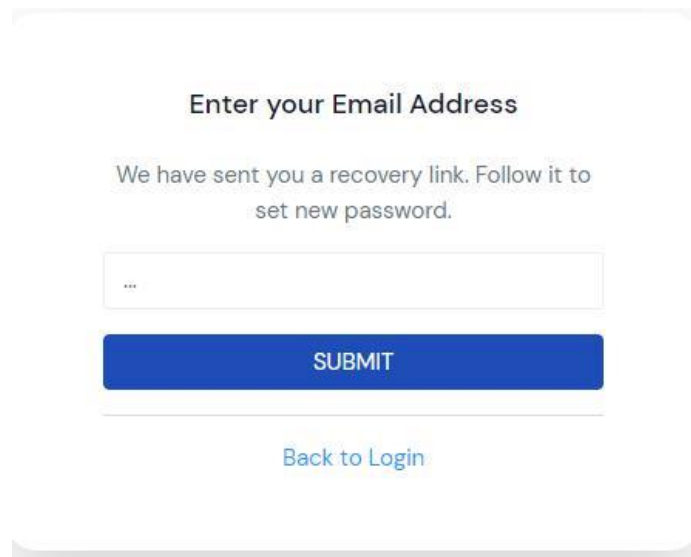


The user will be prompted to submit the correct email in the event that it doesn't follow the email syntax or it's invalid.



*Figure 1. 3: Invalid email interface*

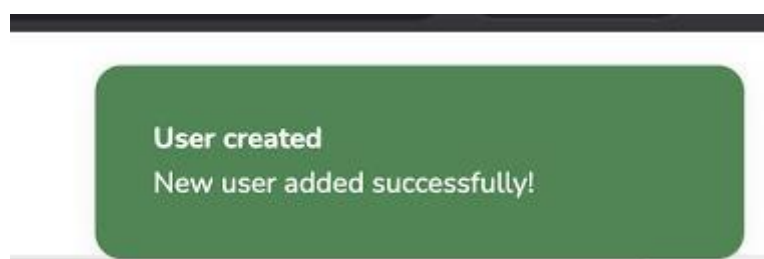
The user will be prompted to reset password through the recovery email sent from the system.

The image shows a web interface for resetting a password. At the top, it says "Enter your Email Address". Below that, a message reads: "We have sent you a recovery link. Follow it to set new password." There is a text input field with three asterisks inside, indicating a password field. Below the input field is a blue button labeled "SUBMIT". At the bottom of the form, there is a blue link that says "Back to Login".

*Figure 1. 4: Reset password interface*

On successful login, the user is directed to their respective dashboard namely TRO, Chairman, Ordinary Member.

On successful creation of a user, admin is notified through a toast message that the user has been created and thus added to the database.



*Figure 1. 5: User created successfully*

Initially before a member receives any loan from the VSLA, no loan record is found in the database therefore the member is made aware.

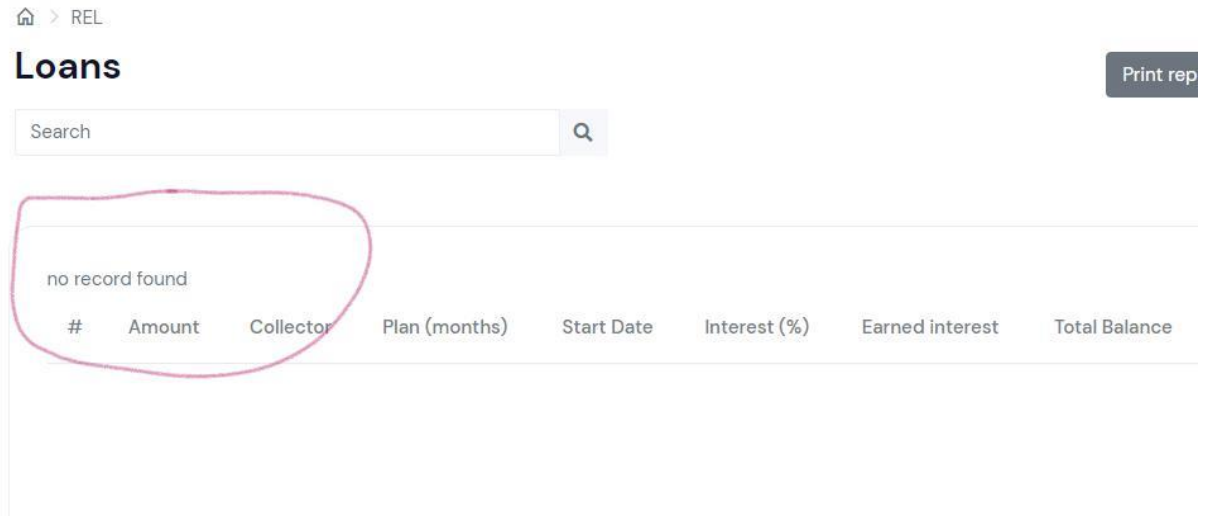


Figure 1. 6: No record found

# Chapter 3: Design output.

## 3.1 Implementation (coding and compilation)

### Development tools

#### ★ Visual studio code

We used Visual studio code, a streamlined source code editor with support for software development operations like debugging, task running and version control.

#### ★ JupyterLab

We used jupyterlab, an anaconda package that is also a web-based interactive development environment for notebooks, code and data. Its flexibility allowed us to configure workflows in data science and machine learning.

### Server

Xampp: Is a completely free, easy to install Apache distribution containing MariaDB, PHP for configuring code and databases.

### Anomalies

The GFA application uses node packages, which might break by application versions are

#### ★ Composer

Composer is a tool for dependency management in PHP. It allowed us to declare the libraries that our project depended on and managed (install/update) them for us.

#### ★ Conda

Conda is an open source package management system and environment management system that we used to quickly install, run and update packages and their dependencies.

#### Integration details

#### ★ SMS API

We integrated a Twilio SMS API that provides us with an interface for sending short message service (SMS) with a one-time password (OTP) that is sent to the system user's cell phone during two factor authentications on the system logon process. We also used the API for sending out announcements in the form of messages.

### 3.4 Documentation

Table 1. 1: Design Details.

Topics	Design output	
<p>Good programming practice</p> <p><i>Efforts made to meet the recommendations for good programming practice...</i></p>	<p>Source Code is...</p> <ul style="list-style-type: none"> <li>● Modularized</li> <li>● Encapsulated</li> <li>● Functionally divided</li> <li>● Strictly compiled</li> <li>● Fail-safe (handling errors)</li> </ul>	<p>Source code contains...</p> <ul style="list-style-type: none"> <li>● Revision notes</li> <li>● Comments</li> <li>● Meaningful names</li> <li>● Readable source code</li> <li>● Printable source code</li> </ul>
<p>Dynamic testing</p> <p><i>Step-by-step testing made dynamically during the implementation...</i></p>	<ul style="list-style-type: none"> <li>● All statements have been executed at least once</li> <li>● All functions have been executed at least once</li> <li>● All case segments have been executed at least once</li> <li>● All loops have been executed to their boundaries</li> <li>● Some parts were not subject to dynamic test</li> </ul> <p>Comments:</p>	

# Chapter 4: Inspection and testing

## 4.1 Introduction

Table 1. 2: Inspection plan and performance

Topics	3.3.1 Inspection plan and performance	Date / Initials
Design output Results from the Design Output section inspected...	<ul style="list-style-type: none"><li>• Program coding structure and source code</li><li>• Evidence of good programming practice</li><li>• Design verification and documented reviews</li><li>• Change-control reviews and reports</li></ul> Comments:	25/09
Documentation Documentation inspected...	<ul style="list-style-type: none"><li>• System documentation, flow charts, etc</li><li>• Test results</li><li>• User manuals, On-line help, Notes, etc.</li><li>• Contents of user manuals approved</li></ul> Comments:	25/09
Software development environment Environment elements inspected...	<ul style="list-style-type: none"><li>• Data integrity</li><li>• File storage</li><li>• Access rights</li><li>• Code protection</li><li>• Installation kit, replication and distribution</li></ul> Comments:	25/09

Result of inspection Approval of inspection.	<ul style="list-style-type: none"> <li>● Inspection approved</li> </ul> Comments:	
---	---	--

## 4.2 Test plan and performance

Based on the user requirements it is important to test the software and check if the different modules that have been implemented meet the user requirements. This helps to assure the quality of the software before it is released. The testing process requires substantial resources in software development. Thorough testing is necessary to be confident that the system works as it was intended to in its intended environment.

### 4.2.1 Test objectives

The main objective is to verify that the functionality of the CFA system version 1.0 works according to the specifications provided in its Requirements Specification Document. The tests were executed to verify that the test cases identify, fix and retest the defects in the severity classes. Other objectives include;

- To examine the different modules that have been implemented have the correct functionality based on the requirements.
- To improve the software's quality.
- To identify and locate the errors in the software.
- To define the tools to be used throughout the testing process



- To communicate to the responsible parties, the items to be tested, the schedule, the test budget and defined environment needs for testing.

#### **4.2.2 Scope and Relevance of tests**

The scope of the tests that were conducted is based on code coverage and module-based coverage.

Code coverage is based on code, for example php code and module-based coverage is based on the modules that are the Loan management module, member management module, decision making module, and finally the typing pattern module. In some cases, it may be impossible to cover certain parts of the system. For example, a program may contain code to detect and handle error conditions that are very hard to simulate in a test.

The tests were designed to cater for verification and validation testing techniques. Verification testing covered the systems requirements during the requirements phase while validation testing was employed during the implementation phase.

Testing is also essential to ensure the following

- It helps ensure that the satisfaction of the end product is maintained
- It helps to ensure the effective performance of the system
- It helps to ensure the high quality of the application

### **4.2.3 Levels of tests**

Module tests: These tests were largely white-box oriented. Each module was tested independently (Loan management module, member management module, decision making module)

Integration tests: After the successful testing of the individual modules, we combined the modules and tested them as a group. This was intended to expose any defects that may arise in the interactions between the integrated modules in the system.

System Acceptance test: This was performed to ensure that the application complies with the system requirement specifications

### **4.2.4 Types of tests**

Input tests: These were intended to test for the issues that may arise as a result of the user input.

Functionality tests: These were performed to ensure that the application proves positive against the functional specifications. The functions were tested by feeding them with appropriate input and examining the output.

Usability Tests: These were carried to ensure that the system is easy to use and the end user will be able to interact with the system easily.

## 4.2.5 Sequence of tests

Table 1. 3: Test Table

Test ID	Test case description	Test procedures	Expected results	Passed
01	Checking if user receives the SMS with the OTP code and email containing the OTP	Input correct username and password	The user receives the SMS	Yes
02	Checking that the data is synced to main database online	Check main database to see that data from the dashboard is there	The data exists in the main Database	Yes
03	Checking if the user roles are created successfully	Login as a Admin and create a user	The user is inserted in the database	Yes
04	Checking if data being sent to the database is encrypted using the Hashing mechanism	Login to phpmyadmin panel and enter a new observation slip	The data in the database is encrypted	Yes

## **4.2.6 Configuration and calculation tests**

While configuring the system, the user does not need to update the node modules because this will cause the system to misbehave.

## **4.3 Precautions**

### **4.3.1 Anomalous conditions**

Anomalies are events that differ from the standard events defined in the application.

Therefore, anomalous conditions in our application basically refer to the deviations of the system from the user expected behavior. From this application, the following are the possible anomalous conditions that may occur during operation of the application.

The user might expect an immediate sms or email after login, but there might be a delay due to the network. This means the user must have a good internet connection to receive the email

### **4.3.2 Precautionary steps taken**

The user is requested to check their internet cable, modem or router. They can also try reconnecting to WIFI.

# Chapter 5: Installation and system acceptance test.

This section ensures that all system elements are properly installed in the host system and that the user obtains a safe and complete installation, especially when installing software products

## 5.1 Input files

For the data analytics component of the system, the installation will only require an installer executable program. The rest of the features will be accessed by a link on the web.

## 5.2 Supplementary files

A user manual will be provided to guide the users on how to use the application.

## 5.3 Installation qualification

Table 1. 4: Checklist of the Installation and system acceptance test

<i>Topics</i>	Installation summary
Installation method  <i>Automatic or manual installation...</i>	<ul style="list-style-type: none"><li data-bbox="656 1465 1312 1570"><input checked="" type="checkbox"/> Automatic - installation kit located on the installation media</li><li data-bbox="656 1619 1312 1696"><input type="checkbox"/> Manual - Copy &amp; Paste from the installation media</li></ul>

	Comments:
Installation media <i>Media containing the installation files...</i>	<input type="checkbox"/> Diskette(s) <input type="checkbox"/> CD-ROM <input checked="" type="checkbox"/> Source disk folder (PC or network) <input type="checkbox"/> Download from the Internet  Comments:
Installed files	<ul style="list-style-type: none"><li>• JS files</li><li>• WDR desktop app installer</li></ul>

<i>Topics</i>	Installation procedure	<i>Date / Initials</i>
<p>Authorization</p> <p><i>Approval of installation in the actual environment.</i></p>	<p><b>The person responsible:</b> Development team leader</p>	<p>16th/08/2022</p>
<p>Installation test</p> <p><i>The following installations have been performed and approved...</i></p>	<ul style="list-style-type: none"> <li>● Tested and approved in a test environment</li> <li>● Completely tested according to test plan</li>   <li>● Partly tested (known extent of update)</li> <li>● Tested and approved in the actual environment</li> </ul>	<p>16th/08/2022</p>

# **Chapter 6: Performance, servicing, maintenance, and phase out.**

## **6.1 Service and maintenance**

As all software systems evolve there are a number of issues that should be taken care of for the software to remain robust and reliable. These needs arise because of a number of factors including the change in the operating environment of the system, upgrades and updates in the technology and programming languages used among other factors. Because of these factors, systems require constant servicing and maintenance.

The team responsible for the development of this particular system is always available to work with the stakeholders on any anomalies that may arise while using the system. For this reason, the current system is the first of the many versions that are yet to come. This will be influenced by the different concerns from the users.

## **6.2 Performance and Maintenance**

Many factors will inform the need for maintenance, these include too much delay in the synchronization of data to the online database for example beyond one minute. The other factor is where software component upgrades are required for example migrating to newer versions of the programming language or the need to adapt a new technology on the market.

The team will be able to provide the following support where need be.

Training of users on how to use the system. There will be training sessions on the site of the users to enable users learn how to use the new system.



Installation, integration and configuration of the system. The time will do the installation and any necessary integrations.

Data migration from the old system. There will be a phased cleaning and migration of the data in the old system to the new system

Software upgrades. Software upgrades will be informed by new user requirements and bugs discovered. The software development lifecycle will be followed from start to end and all the necessary artifacts will be produced during upgrades

Debugging in case of any bugs discovered. Bugs that require immediate attention and rectification will be documented post implementation.

Table 1. 5: Performance and maintenance details

Topics	Performance and maintenance	Date / Initials
Problem / solution	When the number of records that are locally stored and not yet synched too the online database increases, there is a risk of losing that data. Its therefore	16th/08/2022
	encouraged to frequently connect to the internet to ensure that the records are synched to the online database.	
Functional maintenance	The system uses hashing to encrypt and decrypt data. If any issue is discovered, there should be a change to better algorithm of encryption	21st/08/2022
Functional expansion and performance improvement	<ul style="list-style-type: none"> <li>● Use of wireless technology instead of wired connections by the TRO.</li> <li>● Inclusion of an iterator to keep track of changes in the data over time.</li> <li>● Ensure that not too much data is left not synched to the online database</li> </ul>	21st/08/2022

## **Chapter 7: Conclusion and Recommendations**

Ultimately, in the CFA system, the result of all diligence is the ability to manage loans seamlessly, monitor VSLAs and make cognitive decisions effortlessly. This software reduces the amount of manual data import and provides more performance. It is very friendly and can easily be used by anyone. It also reduces the amount of time it takes to document member and VSLA information manually. The CFA system will help increase consistency, improve efficiency and reduce costs.

In order to achieve this, we carried out system analysis and design, implementation, testing and validation which are all the steps of the Software Development Life Cycle.

Throughout the project, we got to work with some technologies for the first time and also improve on our existing knowledge and skills.

# **USER MANUAL FOR COMMUNITY FUND ADVISOR SYSTEM**

## Table of Contents

<b>USERMANUALFORCOMMUNITYFUNDADVISOR SYSTEM .....</b>	<b>68</b>
<b>Table of content user manual .....</b>	<b>Error! Bookmark not defined.</b>
<b>List of figures.....</b>	<b>lxx</b>
<b>List of tables.....</b>	<b>lxxi</b>
<b>8.1 How to run the system for startup .....</b>	<b>72</b>
<b>8.1.1 Recommendations.....</b>	<b>72</b>
<b>8.1.2 How to add users.....</b>	<b>72</b>
<b>8.1.3 How to create a new loan .....</b>	<b>72</b>

# List of figures

Figure 1.1: Add new users .....	72
Figure 2: Create a new loan .....	73
Figure 3: Create a new chairperson. ....	73
Figure 4: Send a message to the phone number.....	74
Figure 5: Manage VSLAs .....	74
Figure 6: Start the python server.....	75
Figure 7: Monitor VSLA performance .....	75

# List of tables

Table 1: .....**Error! Bookmark not defined.**

## 8.1 How to run the system for startup

### 8.1.1 Recommendations

- Open your favorite browser e.g chrome, firefox.
- Make sure your device is connected to the internet
- Type in "<https://cfa-system.com> " to access the system.

### 8.1.2 How to add users.

To add users to the system is done by either the Admin.

Login as an admin, navigate to users, then add users and fill in the fields and click submit.

The credentials for the created user will be sent to him via email. The information sent includes the username and the generated password.

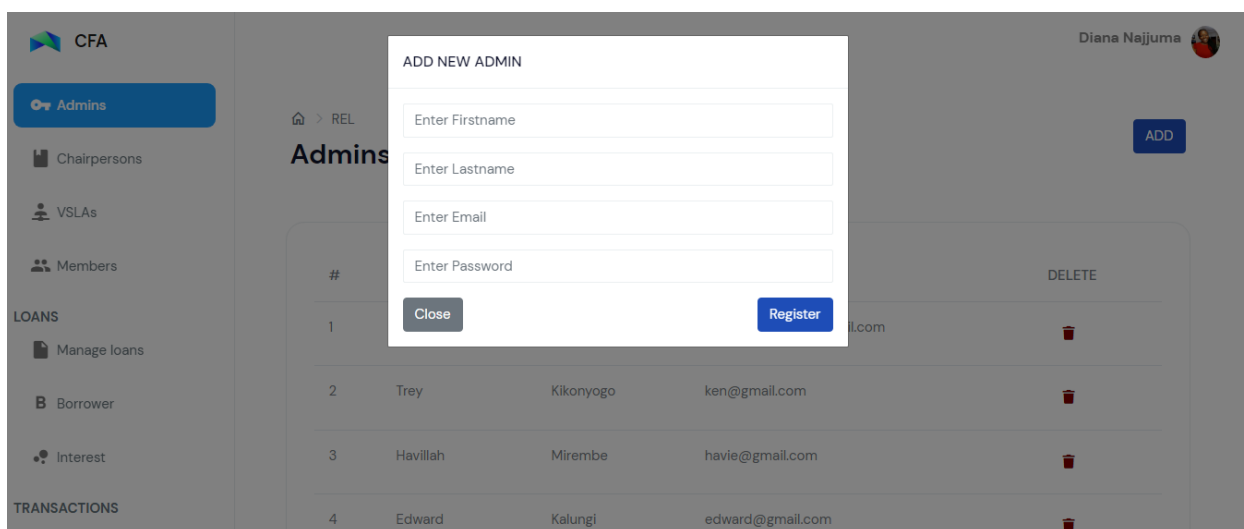


Figure 1.1: Add new users

### 8.1.3 How to create a new loan

Navigate to "Manage loans" on the sidebar, then click add to enter details and "Print report" to generate a report.



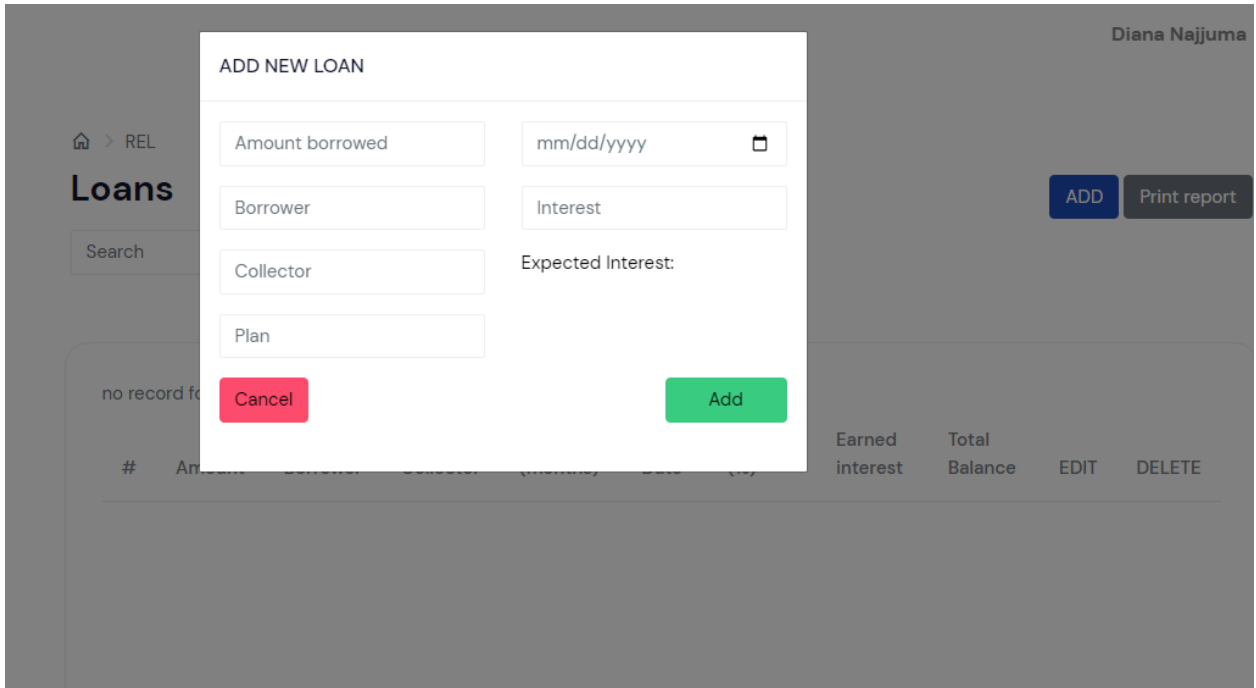


Figure 2: Create a new loan

## 8.1.4 HOW TO ADD A NEW CHAIRPERSON

Navigate to “Chairpersons”. Click “ADD” to create a new VSLA and “Print Report” to generate a report.

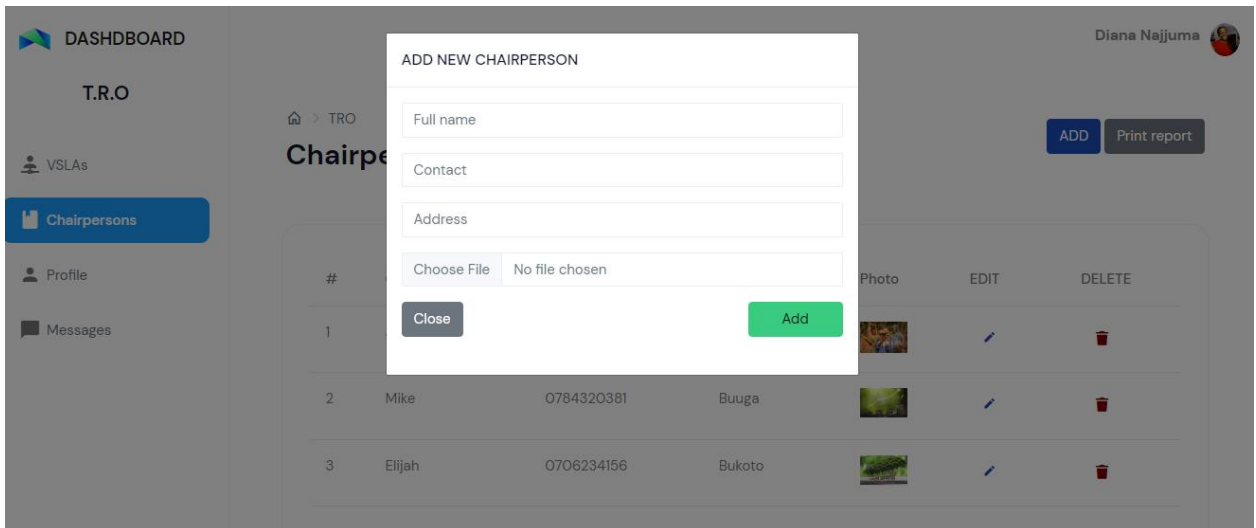


Figure 3: Create a new chairperson.

## 8.1.4 HOW TO SEND A MESSAGE OR PASS ON COMMUNICATION

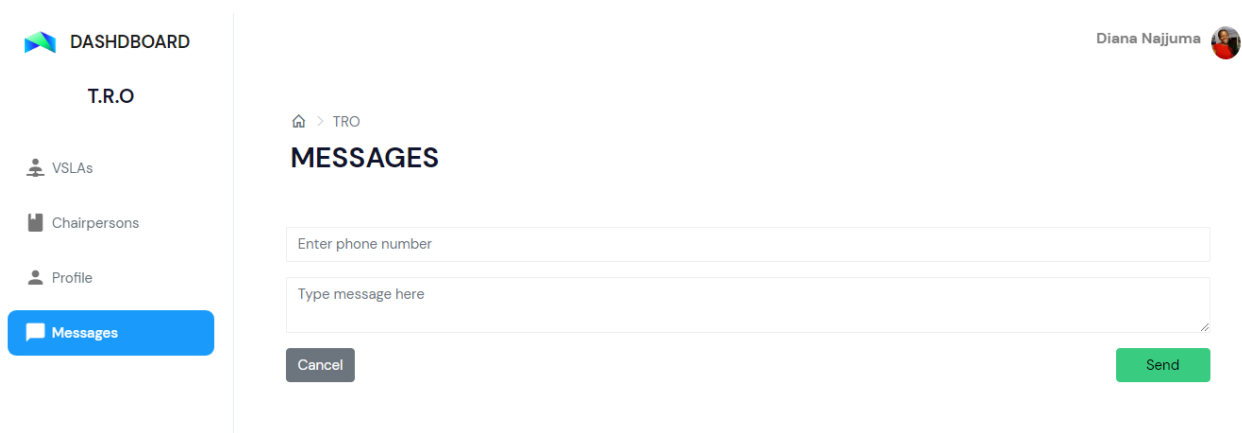


Figure 4: Send a message to the phone number

## 8.1.5 HOW TO MANAGE VSLAS

Navigate to “Manage VSLAs”. Click “ADD” to create a new VSLA and “Print report” to generate a report.

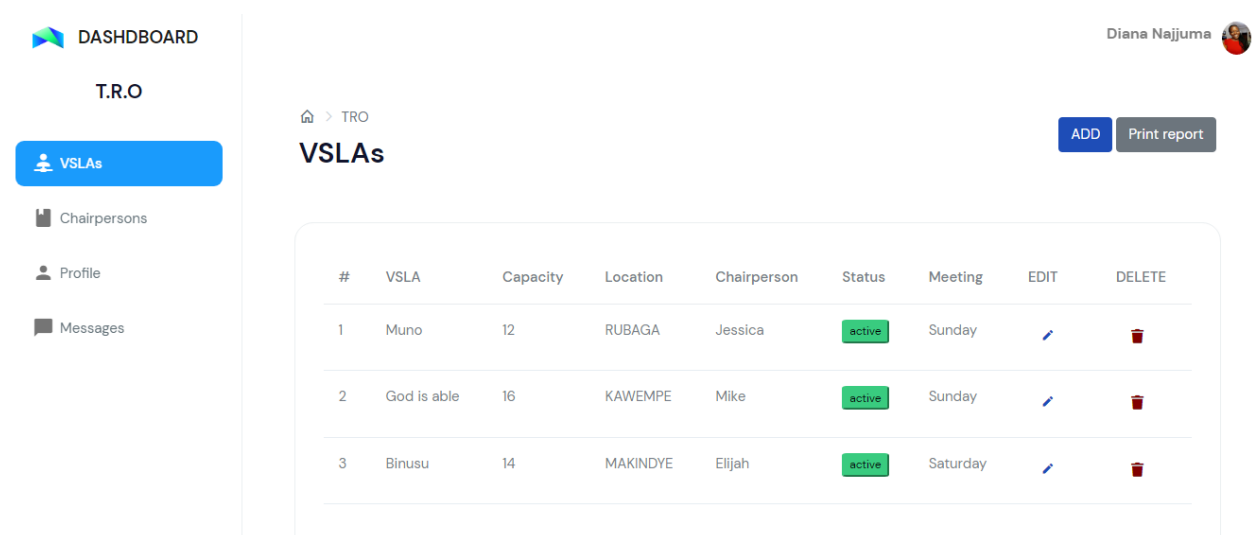


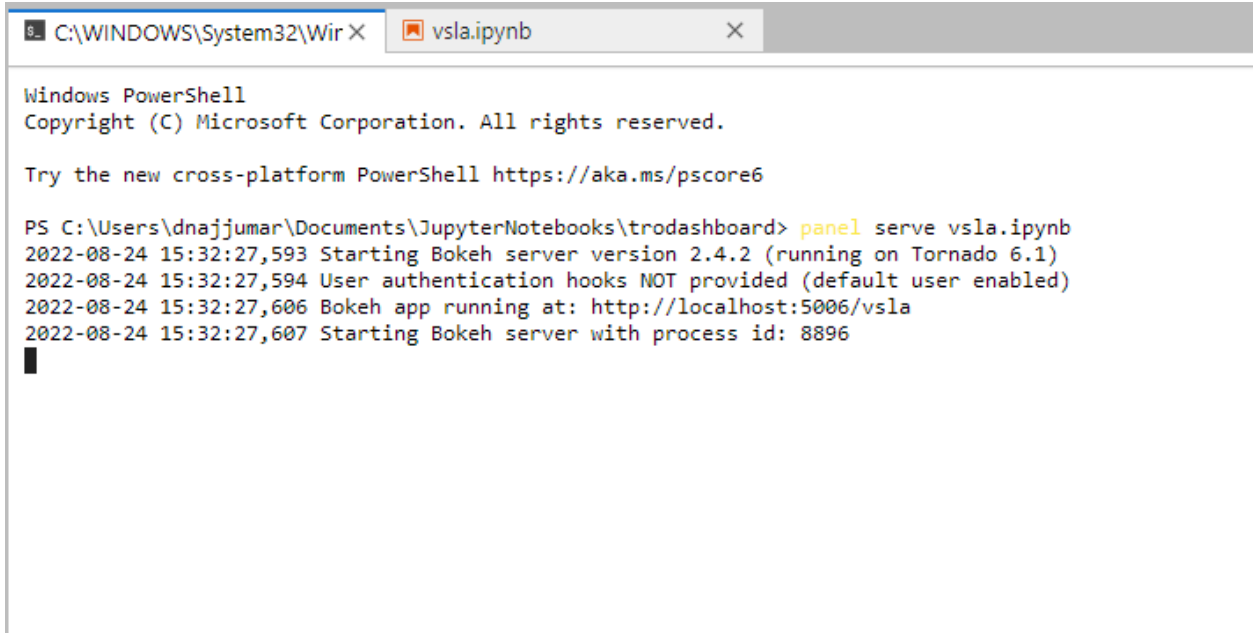
Figure 5: Manage VSLAs

## 8.1.6 HOW TO MONITOR VSLA PERFORMANCE

Navigate into the project folder, and open the terminal with anaconda.

Type `>> panel serve vsla.ipynb`

In your browser type: <http://localhost:5006/vsla>

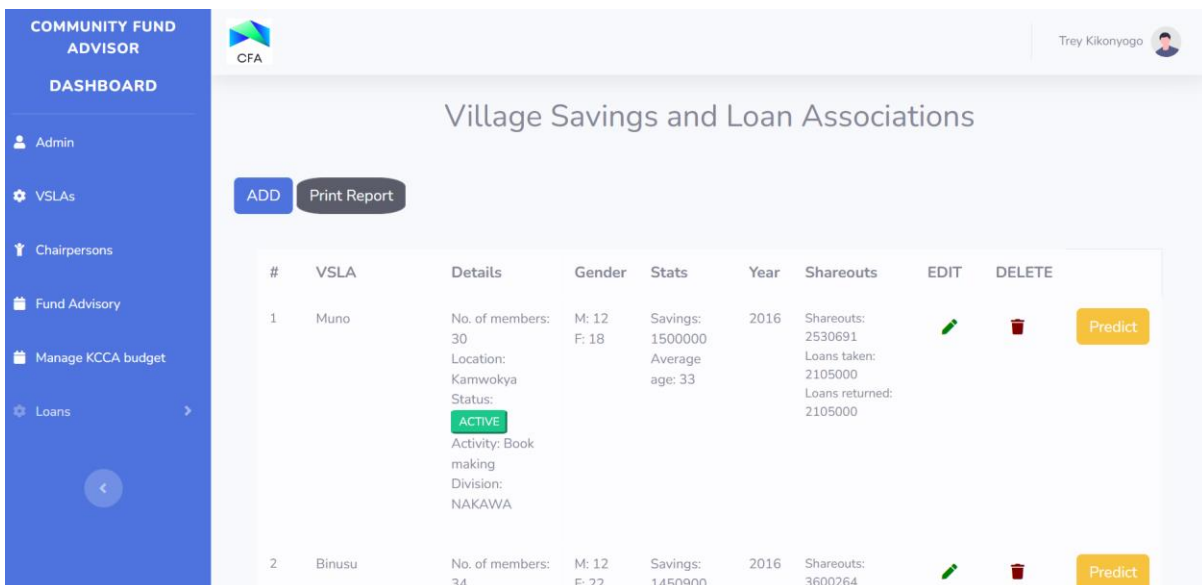


```
C:\WINDOWS\System32\Win X vsla.ipynb X
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\dnajjumar\Documents\JupyterNotebooks\trodashboard> panel serve vsla.ipynb
2022-08-24 15:32:27,593 Starting Bokeh server version 2.4.2 (running on Tornado 6.1)
2022-08-24 15:32:27,594 User authentication hooks NOT provided (default user enabled)
2022-08-24 15:32:27,606 Bokeh app running at: http://localhost:5006/vsla
2022-08-24 15:32:27,607 Starting Bokeh server with process id: 8896
```

Figure 6: Start the python server.



COMMUNITY FUND ADVISOR DASHBOARD

Village Savings and Loan Associations

ADD Print Report

#	VSLA	Details	Gender	Stats	Year	Shareouts	EDIT	DELETE	Predict
1	Muno	No. of members: 30 Location: Kamwokya Status: ACTIVE Activity: Book making Division: NAKAWA	M: 12 F: 18	Savings: 1500000 Average age: 33	2016	Shareouts: 2530691 Loans taken: 2105000 Loans returned: 2105000			
2	Binusu	No. of members: 34	M: 12 F: 22	Savings: 1450900	2016	Shareouts: 3600264			

Figure 7: Monitor VSLA performance