BACHELOR OF SCIENCE IN INFROMATION TECHNOLOGY

PROGRAMMING PROGECT: BIT 2206

SAVINGS AND CREDIT COOPERATIVE MANAGEMENT SYSTEM FOR LOCA SAVINGS AND CREDIT COOPERATIVE

BY

NJOROGE MORRIS KIMANI

16/05387

PROGECT SYSTEM REQUIREMENT SPECIFICATION IN PARTIAL FULFILMENT FOR THE REQUIREMENTS FOR THE AWARD OF A DEGREE IN INFORMATION TECHNOLOGY

PRESENTED TO: Dr. LUCY MBURU

Declaration

I declare that this project is my original work and has not been presented in any other college or university for the award of a Diploma or a Degree.

Student

Name…….. ……. Date……..

Signature…….

Table of Contents

Contents

1.0 Introduction

1.1 Purpose

Software development is a process by which one develops a system based on studies on the existing systems, their strengths and limitations and working to ensure that one produces an improved product that solves unique problems.

The purpose of the system requirements specifications document is to show system designers the needs of the clients and

1.2 Intended Audience

This document will be available to the top management of LOCA SACCO, users of the system being developed, future developers, project management committee of KCA University and will also serve as a reference to other students interested in the field of software development.

1.3 Aims of SRS

The purpose of the Software Requirements Specification is to outline the requirements for The LOCA SACCO management system. It will be built on Visual Basic.Net and MySQL .It will be operating system dependent and accessible only to the connected workstations.

1.4 Project Scope

The project is aimed at creating a platform for effectively developing a computerized system that will meet the needs for LOCA.

The new system is going to run on two servers i.e. an application server that will host the user interface and the database sever that will be used to store and retrieve data. The data will be stored in a MySQL database within the Sacco's premises.

1.5 References

https://www.t4tutorials.com/srs-software-requirement-specification/

Software Engineering, A practitioner's Approach,

Harvard Publishers, USA, 1985.

2.0 Overall Description

The LSMS is a relatively new product which will function on a client-server model. The user interface which will also hold a lot of the processing logic will be installed in individual workstations within the offices while the database will be installed in one server and will be available to all the workstations when needed.

2.1 Product Features

2.1.1 Data Capturing

Data will be drawn from members directly by the staff who will be interacting with them to obtain information for necessary transactions. The staff will enter this information directly into the system.

2.1.2 Data Processing

Data will be processed by the application immediately data is collected.

2.1.3 Data Output

After data capturing and processing, output will be in the form that the user requires.

2.1.4 Data Validation and Integrity

There will be control mechanisms that will check the validity of the data being captured in accordance to the rules provided.

2.1.1 Monitoring

The system will be able to oversee the various actions being taken within the system and documenting them. It will also inform users of any errors that may occur during the system's runtime and provide possible solutions on how to solve the error(s).

2.2 User Classes and Characteristics

The users of the systems have been using physical books hence they will require computer-literacy training for them to be able to fully and effectively use the new system. To that effect, a user guide will be provided to the users in friendly language to help them have optimal use of the system.

The system will be easy to use as it will have simple forms which will be simple to navigate. The user interface will run on specified workstations.

2.3 Operation Environment

The following will be necessary for optimal performance of the system:

2.3.1 Hardware

- Minimum hardware specifications for the workstations are **2 GB RAM**, **80 GB** hard disk,
- Minimum hardware specifications for the database server are **4 GB RAM**, **500 GB** hard disk,
- A thermal printer

2.3.1 Software

- Microsoft windows 2012 server Operating System and windows 10 for the clients/workstations.
- MySQL Database application

2.4 Design and Implementation Constraints

The current constraints are:

- Interfacing different applications is challenging.
- Current work stations do not meet the minimum requirements
-

2.5 User Documentation

Users need to be trained on how to use a new system for the project to become complete success and also maximize on how the developed software is being utilized.

The users will have the following documents at their disposal from either the vendor or developer which will be relating to the general issues that users will encounter during use of the system:

- User Manual:
    - Provided by the developer
    - Easy to use and understand
- Knowledge Base
    - Good for quick references.
    - Assumption is that the user knows what they are looking for.

    - Compiled by the vendor

- Command prompts
    - They assume knowledge of a command.
    - Can provide information about correct usage of a command when an error occurs.
    - Designed by the vendor

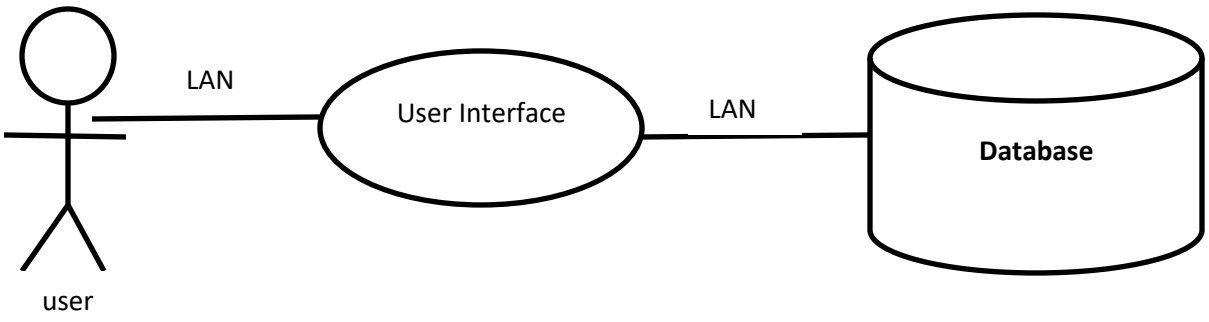
2.6 Assumptions and Dependencies

- Test data provided is assumed to be reliable.
- The MySQL database will continue to be available with existence of support and new updates.
- There will be a dedicated server for optimum performance of the database before production of the system.
- Staff will always be computer literate.
- New versions of the system will be produced to deal with any bugs that may not be captured during testing.
- It is assumed that all the clients/workstations will always run a compatible version of windows operating system.
- It is assumed that the network will always be secured such that no outsiders will gain unauthorized access to either the server or workstations in the enclosed network.

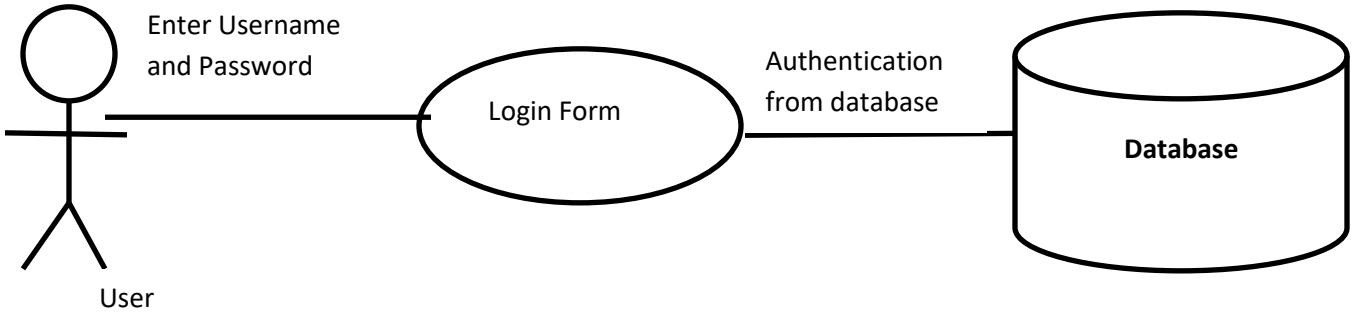
3.0 System Features

The system should be able to capture all data about the members which will be entered by staff or come automatically from other sources such as banks through a form interface and it will be stored directly into the database.

It should also be able to produce quick and detailed reports on members, loans and other necessary information very fast hence eliminating the current time wastage experience in putting together this information manually. The system can generally be depicted as the following image below.



3.1 Access to User Accounts



3.1.1 Description and Priority

To gain any access to the functionality of the system, one's identity and therefore the functions of the systems which they can utilize, this takes place as the system establishes a connection with the database as it is starting yup. The priority of this case is "High".

3.1.2 Stimulus/Response Sequences

1. The user starts the application on their workstation.
2. A connection with the database is established and the user enters their credentials.
3. The credentials entered are compared with those in the database to grant/deny access to the system.

3.1.3 Functional Requirements

REQ-1: Connection to a LAN

It ensures that the database can be accessed by the application as it runs.

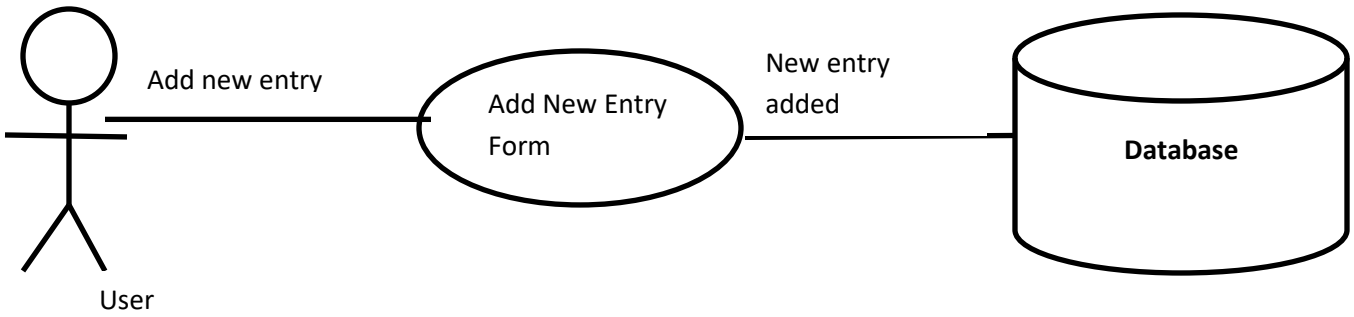
REQ-2: User Interface

It will be implemented as forms. They will be used for data input.

REQ-3: Access Control

It will be enforced by authentication of the credentials obtained from a user.

3.2 Creating New Entries



3.2.1 Description and Priority

It entails the creation of new accounts for members of the SACCO as well as the other essential staff required to use the system due to the issue of security and access control. It is of "**High**" priority because it encases the core business of the system.

3.2.2 Stimulus/Response Sequences

Connection to the database is necessary in order to use this functionality as well as access to the new entry form(s).

Attendants

1. The user clicks the "Add New Member" button after logging into their account the system.
2. A new form appears with various data fields that are to be filled with information from the new member.
3. Data entry is complete; the data will be saved in the database by clicking "save new record".

Tellers

1. The user clicks the “New Loan”
2. A form appears with fields on data that is to be collected.
3. The user then clicks “save” and the application checks if all the fields have been filled and determines if the loan can be given based on the data provided.
4. A response is given after the data has been processed and stored.

3.2.3 Functional Requirements

REQ-1: Connection to a LAN

It ensures that the database can be accessed by the application as it runs.

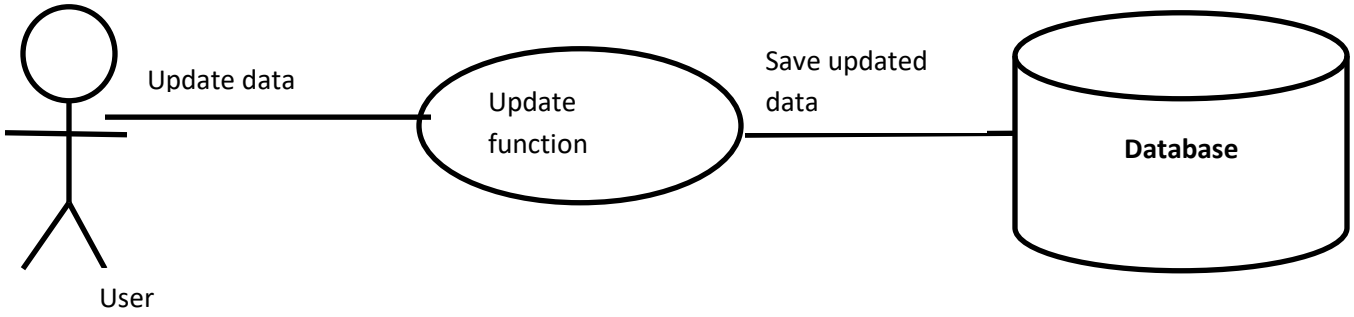
REQ-2: User Interface

It will be implemented as forms. They will be used for data input.

REQ-3: Accurate data

It will be paramount. It can however be changed at any point of time to maintain it’s timeliness and accuracy.

3.3 Editing/Updating Data



3.3.1 Description and Priority

There is need to alter the information that has been stored in the database. This may be due to a change in someone’s information or the organization’s policy that requires addition/truncation of data. It’s priority could be termed as low since the rate of having erroneous data will be very low.

3.3.2 Stimulus/Response Sequences

Connection to the database is necessary in order to use this functionality as well as access to the new entry form(s) because this is where the editing function is located.

Attendants

1. The user clicks the “Add New Member” button after logging into their account the system.
2. A new form appears with various data fields that are to be filled with information from the new member.
3. Data entry is complete; the data will be saved in the database by clicking “save new record”.

Tellers

1. The user clicks the “New Loan”
2. A form appears with fields on data that is to be collected.
3. The user then clicks “save” and the application checks if all the fields have been filled and determines if the loan can be given based on the data provided.
4. A response is given after the data has been processed and stored.

3.3.3 Functional Requirements

REQ-1: Connection to a LAN

This will enable it to connect to the database.

REQ-2: User Interface

It will be used to collect the fresh data and send it to the database to update the existing data.

4.0 External Interface Requirements

4.1 User Interfaces

It is a description of how the system will interact with its users and other software as it is being run. This system will use Visual Basic forms as an interface with the users and will be run from any workstation within the organization's premises.

4.1.1 Graphical User Interface (GUI)

Normal users will use the system through a GUI while the more experienced users will access the database directly using SQL (Structured Query Language) commands on the database's command line console.

4.1.2 Buttons

They will be easy to see and their use very clear to the user as they will be having simple labels to state their functions. They will be widely used in the system to eliminate the need of menus which take time for one to get accustomed to.

4.1.3 Messages

There will be some specific types of messages:

- *Error messages:* They will alert the user on actions that are not supposed to be taken. For example, entering an alphabetical character into a numerical input field.
- *Informational:* They will give information on the state of the system at a particular point in time. For example: if a transaction is successful.

4.2 Hardware Interfaces

4.2.1 Local Area Network (LAN)

It will provide a platform on which communication will take place between the system's components. This will include networking equipment such as: routers, cables and switches alongside the topology that will be implemented to enable the system to access the database.

4.2.2 TCP/IP Protocols

They will be used to identify the workstations within the LAN hence giving them access to the database via the server.

4.2 Software Interfaces

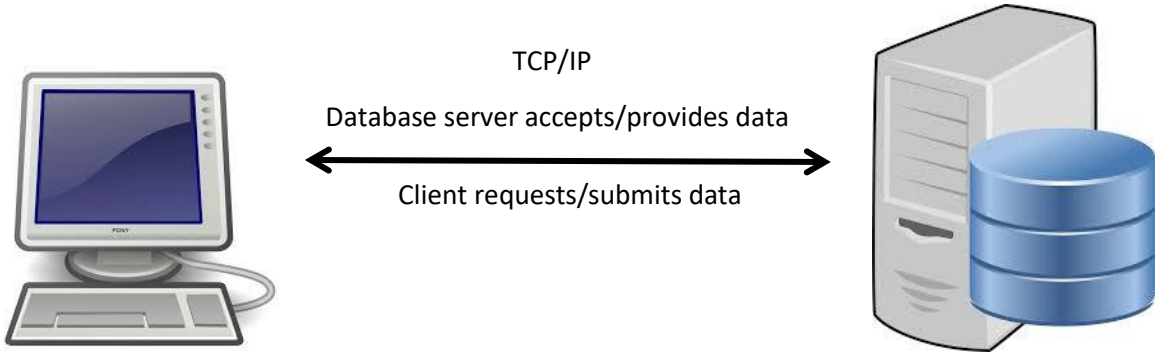
4.2.1 Operating Systems

It is software that facilitates the basic functions of a computer. In this case, it is advised that the operating system be Windows. As for the server, it will be windows 2012.

4.2.1 Database

The database that will be implemented will be MySQL.

4.3 Communication Interfaces



It will utilize a 2-tier processing architecture where there the workstations/clients will hold the user interface and business logic and the database server will be the data repository.

5.0 Other Non-Functional Requirements

5.1 Performance Requirements

The system will require the following in order to function at it's peak:

1. At least 1GB RAM for workstations and at least 4GB for the server.
2. At least windows 7 and higher for the workstations and Windows Server 2012 for the server.
3. At least 40GB internal storage for the workstations and 200GB for the server.

5.2 Safety Requirements

Using the system over time will require caution to safeguard the data. The following safety measures will be implemented.

5.2.1 Usernames and Passwords

To access the system, one must have a username and password.

5.2.1 Backup

To ensure that all the data is safe, all data must be backed up in a location that will not be accessible on the network. This is to ensure that in the event the database is corrupted, a backup mechanism is in place.

6.0 Other Requirements

6.1 Reliability

The system will always be available to users at all times.

6.2 Scalability

Upgrading the system will allow growth and able to incorporate even more workstations possibly even form a system that is distributed in different geographical areas without affecting performance.

6.3 Maintainability

The system will be open to future upgrades in both features and functionality.