# **CHAPTER 1**

# **GENERAL INTRODUCTION**

**Introduction**

Now a days technology is one best pillar of development of the world we are living today ,it has become part of our everyday lives. Information Technology is one of the best convincing scientific, technological and applicable developments for this century. Technology in the workplace allows to our daily activities to expand quickly and efficiently.

Government institutions, banks, schools, companies, etc… use those developed software in their daily activities so that workers, clients, all users in general get a quick and easy access to the information needed and then perform well all task in the time needed.

## Background of study

Rwanda Cooperative Agency (RCA) is a public institution which is in charge of promotion, Registration and Regulation of Cooperatives in the Country. Actually the Rwanda Cooperative Agency (RCA) makes registration of cooperatives and give them perpetual succession and legal personality through the entire period.

RCA shall be placed in category of regulatory Public Institutions in charge of regulation and promotion economic, social, and other activities of the general interest. RCA shall have legal personality, administrative and financial autonomy and it shall be governed in accordance with laws governing organs of public Service.

Decision makers tend to place the greatest attention or more attention on more recent information. Changing decision-making environment, managerial requests, and decision-maker limitations create a need for more and better decision support. We should consider building a computerized decision support system when good information is likely to improve the quality decisions.

IT supports new methods for precision activities of cooperatives like computerized supporting solutions to the identified problems that apply for cooperatives, to get easily the supporting information by using telecommunication components like internet or telephones. One such communication tool is the Web Site, which simply replaces the newspaper as a communication tool.

As the process goes on, online registration will be an important process which consists of guiding cooperative leader to register and get that perpetual succession and legal personality easily and actions in progress to ensure they are in process for targeting their objectives and performance. Therefore I am proposing a system to guide those cooperative leaders. I think it will be helpful.

**Statement of problem**

In Rwanda we have a problem thatin the existing system where the people want to register the cooperative they go to the nearest sector without knowing if the desired cooperative leader is available. People waste their time while going to check if the Sector Cooperative leader has reached to the sector and sometimes he/she is not even there.

Today’s RCA challenge is the increasing of Cooperatives in our Country because of Rwandan policy that establish people to group in cooperatives. This requires RCA to make more efforts for providing good training for the preparation of creating cooperatives.

* RCA faced the problem of registering cooperatives, counting cooperatives ready for perpetual succession and legal personality used take much time which is not good in delivering services.
* Another problem is that the cooperatives members don’t know in advance the requirements needed to be accepted as cooperative and this make cause the problem of doing many movement while going and coming back to get those information .
* When the Cooperatives don’t have perpetual succession and legal personality the Cooperative leaders do what they want because the members don’t have where to ask help.
* For the Cooperative which are not registered can’t get any help from Rwanda Cooperative Agency unless they have registered

**Motivation and Interest in the Study**

We believe that for each problem there is a solution when you get time to think about the problem. As many organizations need to evaluate cooperatives’ activities progress ,this gave me the motivation to develop an application named “Online Cooperatives Registration Management Information System” that will help to register cooperatives, counting cooperatives ready for perpetual succession and legal personality.

As a student, it is an opportunity to put in practice the knowledge gained throughout my academic training at Adventist University of Central Africa by solving problems and improving services using technology.

For my university, it will be a way of achieving one of AUCA’s objectives which is to help AUCA students to become useful member of society; empowered by the desire to develop the society not only with theoretical intellectual skills but also with a demonstrated desire of practical research endeavors that lead to country’s development.

**Objectives**

This study has general and specific objectives as they are stated in the following subsections:

General objectives:

The main objective of this study is to design and implement a cooperatives registration project support website which incorporates different cooperatives members’ services which are needed by Rwandan population.

**Specific objectives:**

* To analyze the existing system and its problems;
* After making assessment of the existing system then develop the needed software to solve the problem.
* To suggest the integration of a new system to provide solutions to the problems and meet the requirements of the system.
* Implementation of the system which can produce necessary reports.
* Analyze and design database where information regarding the supporting system are stored.
* To design a system this will reduce the time taken by cooperatives leaders in order to get information .

All techniques, methods and tools that will be used to accomplish this work will be described in the coming chapters.

**Challenges**

Challenges to bring this work to an end may be defined as difficulties which can prevent the system working as it is expected.

Developing a computerized system is not a simple task, even for a small organization. The process demand to be attentive and have sufficient time and experience in order to meet all cooperative needs.

Some of the challenges that we faced in bringing accurate solution to cooperative problems is the insufficient experience in developing such system.

**Project scope**

This final project is focused only on Rwandan’s cooperatives registration activities support, to provide cooperative information through the new web application; the system will be accessed using computer, browser and internet.

**Expected results**

The new system is expected to give the following results:

* The system will help the sector cooperative Officer and district cooperative Officer increase the performance and production.
* Cooperatives registration’s service will be provided quickly as possible to many cooperatives at any given time. This will drastically reduce the amount of time wasted looking for information on cooperatives and requirements.
* The system will support everyone wanting to grow his/her activities.

## 

**The structure of the work**

This study has five chapters:

Chapter One entitled General Introduction will provide basic information on the organization structure and activities and will present problems which face the management of the activities, motivation, objectives, methods and techniques of the study, delimitation of the study, expected results and organization of the report.

The Second chapter will emphasis on the analysis of the existing system where we will describe activities and how they are register the cooperatives using the organization structure , deeply discover their issues, and suggested solutions proposed on those problems. This chapter will describe the organizational environment in which the new system will be implemented.

Chapter three entitled Analysis and Design of the new System is normally the logical conception of the new system. It will portray the conceptual process of the solutions proposed to solve the problems of the existing system.

The forth chapter will focus on the technical aspect of the application and the interpretation of the results, where we will explain the new system focusing on how the application has been conceived, as well as the technologies used to build the software.

Finally chapter five will conclude our project as well as recommendations for future development.

# 

# **CHAPTER 2**

## ANALYSIS OF THE EXISTING SYSTEM

**Introduction**

A good understanding of the existing system is Manual for the conception of the new system which is automated. This analysis also allows the persons responsible to be communicating easily and understand its impact on the interaction of the services as well as on the organizational process and decision making.

Through this chapter we will describe in a clear way the organizational environment in which the new system will be implemented automatically. After defining the key words used in this project ,the current chapter consists therefore to organizational structure, mission ,vision, the present system of Rwanda Cooperative Agency, the problem faced with during usage of the current system as well as the proposed solution.

In order to well understand the existing system we will try to explain some terms that will be used, the methodologies and the environment in which the system will be implemented.

**Definitions of Terms**

Before the analysis of the existing system it is better to have the same understanding of terms which will be used. So let explain some of those terms.

**Agency:** The abstract principle that autonomous beings, agents, are capable of acting by themselves.

**Cooperative:**  is "an autonomous association of persons united voluntarily to meet their common economic, social, and cultural needs and aspirations through a jointly-owned and democratically-controlled enterprise   "

**Online:** the term online have specific meaning with respect to computer technology and telecommunication. In general online indicates a state of connectivity.

**Registration:** when a name or information is recorded on an official list.

**Legal:** Allowable or enforceable y being in conformity with law of the land and the land and the public policy ;not condemned as illegal.

**RCA:** is a public institution which is in charge of promotion, Registration and Regulation of Cooperatives in the Country.

**Information**, it means the data used in this research.

**Project:** This is a study of a particular subject done over a period of time, especially by student.

**A request:** the act or an instance of asking for something the request can be in terms of documents or emails, support etc

**Description of the Current system Environment**

In order to understand the relationship between inputs, outputs and processes, it is important to understand the environment in which all process of cooperative registration for Rwanda Cooperative Agency occurs.

At this time an individuals or cooperative s who wish to register cooperative need to approach nearest Sector.

The entire process involves the Rwanda Cooperative Agency where cooperative leader who register the cooperative needs to be identified to sector.

**DESCRIPTION OF CURRENT SYSTEM**

Actually the current system that are used to register cooperatives and get that perpetual succession and legal personality through sector ,District and then goes to Rwanda Cooperative Agency where the leader of cooperative in village goes to sector cooperative Officer to tell him to register the Cooperative and then when that leader see that cooperative fulfill all requirement approve it and when that cooperative doesn’t fulfill the requirements go and search what are missing then he/she return back.

Some cooperatives should either live or work far from the sector. This may create long journey to go to the sector , in that journey of course the money will be needed for tickets of transport to achieve there. Also some cooperatives work far from the sector .

**Mission**

To develop the cooperative sector such as it serves its members equitably, efficiently and empowers them economically

**Vision**

To promote an automous and economically viable cooperative movement founded on the cooperative values and principles and is able to enhance social integration and uplifting the standard of living of its members.

## Methodology and Data Collection Techniques

Data collection can be gathered from a number of sources, which include documents, the workplace, the Internet surveys, focus groups, field notes, questionnaires and social interaction or interviews. The followings are techniques used in the analysis of the existing system.

## Observation

Observation is the act of careful monitoring of phenomena, without desire to change, with means of investigation and study appropriate. To observe, it is necessary to go to the location of user activity, at least by influencing the situation.

Observation was a big tool for us in developing this research as it is in a scientific research, we cannot do anything without relying on observation as it is among the tools that help us in mastering the existing system.

As mentioned in the definition above one of ways we used in understanding the flow of work of existing system we did an observation in company (RCA) looking on how they work during registering cooperatives and look for techniques they use during analysis of their data.

**Documentation**

The documentation is process of consulting works (articles, books, reports, and websites) on the topic of interest. This method was used when collecting data from reports, files and official documents of Rwanda Cooperative Agency relevant to this work. The documentation allows us to familiarize ourselves with the various tools and models that we used to develop the application.

Consulting documentations about cooperatives registration in our Country, news about cooperatives on Internet, reading library and books and different thesis related to the use of ICT in managing cooperatives and in analyzing of recorded data(data mining), has been one of the preferable techniques used to carry out this research work.

#### Interviews

Interview is verbal questioning .in research is defined as a two person conversation ,initiated by the interviewer for the specific purpose of obtaining research relevant information and focused by him on the content specified by research objectives of description and explanation.

To be able for catching everything about the existing system, we do not only need documentation.

I had an interview with many people who are cooperative members and director of cooperative at Rwanda Cooperative Agency in order to get a general view of how cooperative s are currently positioned in our country.

**Problems of Current System**

**Time Management**: The cooperative leader takes couple of time for going to the Sector to register the cooperative . And time he/she waits to be served because there are many cooperative leaders who come to register their cooperatives .

**Long Journey:** Sometimes most of cooperative leaders are located in long distance from where the Sector where they can go to register their cooperatives, so they must take journey to go there because there is no other choice to register them.

**Transport:** Some Cooperative leaders spend their money, where they pay the tickets of transport. This issue occurs on the Cooperative leaders who live far from the Sector.

**Availability:** Sometimes the cooperative leader goes to sector and then they don’t find the sector cooperative Officer.

**Proposed Solutions**

**Functional Requirements:**

* The system will provide user interfaces that will help them during recording data in an easy manner.
* System will allow admin ,Sector cooperative officer and district cooperative officer also cooperative leader to create account..
* The system will register admin ,Sector cooperative officer and district cooperative officer also cooperative leader.
* System will allow admin to lock and unlock user.
* It will allow the registration of cooperatives
* The system will allow Sector cooperative officer, cooperative leader and district cooperative officer to view and approve cooperatives.
* The system will allow Admin ,Sector cooperative officer, cooperative leader and district cooperative officer to login and logout
* The system will provide report.
* The system will allow Admin ,Sector cooperative officer, cooperative leader and district cooperative officer to share information.
* The system will allow admin to send message to Cooperative leader.

**Non-Functional Requirements:**

**Definition:**  is a requirement that specifies criteria that can be used to judge the operation of a system, such as performance, speed, Usability, security, etc.

The nonfunctional requirements are categorized based on three different criteria:

**Usability**

The application must be available for cooperatives leader, Sector cooperative officer (SCO) , district cooperative officer (DCO) and Administrator.

**Performance**

* The web server shows the registration form through the browser, and then it registers the cooperatives.
* The system should be available to all users 24 hours a day.
* The system shows all the cooperatives which are registered.

**Security**

* All user must use username and password in order to access on the system.
* Other users or unauthorized users should not have access on the system.

**CHAPTER 3**

**ANALYSIS AND DESIGN OF NEW SYSTEM**

## Introduction

In order to build a strong and successful system that meets the user needs, a deep analysis and clear design are required so as a strong foundation is important for a strong superstructure.

System development generally involves two major components:

* System Analysis.
* System Design.

Systems Analysis and Design is an active field in which analysts repetitively learn new approaches and different techniques for building the system more effectively and efficiently. The primary objective of systems analysis and design is to improve organizational systems (Tutorial Point, 2015).

## Analysis and Design Methodology

### Objects oriented methodology

Object Oriented Methodology (OOM) is a system development approach encouraging and facilitating re-use of software components. With this methodology, a computer system can be developed on a component basis which enables the effective re-use of existing components and facilitating the sharing of its components by other systems (Grady, 2009).

They are two main basic building blocks that are useful in development of the system by using Object Oriented Methodology those are: Classes and Objects.

An Object is an instance of a class; it is defined as a discrete entity with a well-defined boundary that encapsulates states and behavior expressed as a software representation. An Object can be a thing, a concept, or an event. By knowing what an object means, simply a class is a description of a collection of objects with common attributes and behaviors (Alan, 2005)

Object-oriented modeling is typically done via use cases and abstract definitions of the most important objects. The most common language used to do object-oriented modeling is the Unified Modeling Language (UML).

### Unified Modeling Language (UML)

The Unified Modeling Language (UML) is a general-purpose modeling language in the field of software engineering, which is designed to provide a standard way to visualize the design of a system. It is a standard notation for modeling the Object-oriented systems.

It was developed by Grady Booch, Ivar Jacobson and James Rumbaugh at Rational Software in the 1990s. It was adopted by the Object Management Group (OMG) in 1997, and has been managed by this organization ever since. In 2000 the Unified Modeling Language was accepted by the International Organization for Standardization (ISO) as a standard for modeling software-intensive systems.

Unified Modeling Language (UML) combines techniques from data modeling (entity relationship diagrams), business modeling (work flows), object modeling, and component modeling. It can be used with all processes, throughout the software development life cycle, and across different implementation technologies (Louis Rivest, 2002).

The Unified Modeling Language (UML) offers a standard way to visualize a system's architectural blueprints, including elements such as:

* activities
* actors
* business processes
* database schemas
* (logical) components
* programming language statements
* Reusable software components.

UML has synthesized the notations of the Booch method, the Object-modeling technique (OMT) and Object-oriented software engineering (OOSE) by fusing them into a single, common and widely usable modeling language. UML aims to be a standard modeling language which can model concurrent and distributed systems (Jeffrey L. & et al, 1998).

**Analysis of the New System**

**Requirement Analysis**

After getting all customer needs or requirements we have to analyze them, doing so different models or diagrams are used. At this particular point we will focus on use case diagram. Use case diagram describes the functionality provided by a system in terms of actors, their goals represented as use cases, and relationships between actors and use cases. The followings are components of a use case model:

**Actor**

An actor specifies a role that some external entity adopts when interacting with a system directly. It may represent a user role or a role played by another system that touches a system.

For performing system analysis activities it is necessary to have a modeling language. One of the most used languages today is unified modeling language. This language is used to generate several diagrams. Some diagrams generated by unified modeling language are the following:

* Use case diagram
* Class diagram
* System architecture diagram
* Sequence diagram
* Database diagram

**Use cases**

A use case describes a sequence of actions that provide something of measurable value to an actor and is drawn as a horizontal ellipse.

**Relationships**

Kram, K. E. (1988), Meaningful relationships between actors and use cases, which is a UML association symbol.



**System boundary boxes (optional)**

By Van Oorschot, P. C. (2003, January), A rectangle is drawn around the use cases, called the system boundary box, to indicate the scope of system. Anything within the box represents functionality that is in scope and anything outside the box is not.



**Actors**

An actor is a person, organization, or external system that plays a role in one or more interactions with the system.



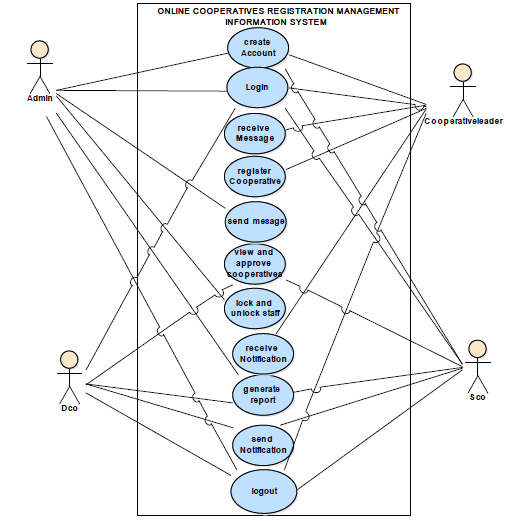
**Table**

A table is a set of data (values) that is organized according to a model of vertical columns (that are identified by their name) and of horizontal lines. A table has a number specified of columns, but can have any number of lines.



**Use Case Diagram**

Use case diagrams are behavior diagrams used to describe a set of actions (use cases) that some system or systems should perform in collaboration with one or more external users of the system (actors).



**Figure 2: RCA use case diagram**

**Use-case description**

Use Case description details what a use case do, and what it requests in order to be well executed. Each use case looks like this:

**Name :** a name of a use case

**Description:** what a system intends to do

**Actor:** the actor involved in the use case

**Pre-condition:** the system state before the use case can begin

**Post-condition:** the system state when the use case is over

**Normal flow:** the actual steps of the use case

**Alternative flow**: steps which may happen in case a normal flow fails.

**1.Create new account**

|  |
| --- |
| **Name:**  Create account |
| **Actor:** Admin, Dco,Sco, Cooperative leader. |
| **Description:** Help for Admin, Dco , Sco and Cooperative leader to create an account for accessing the system. |
| **Pre-condition:** the users have to have valid needed identification information |
| **Post-condition:** the system should send welcome message which contain his/her names |
| **Main Flow:** 1.User requests System to create an account,  2. System displays a form to fill the identification information,  3. User fills in the requested data,  4. User sends the filled form to System,  5. System validates data,  6. System adds data in database,  7. System sends a welcome message to the User. |
| **Alternative flow:**  If the information are not valid the system shows the error where is not correct. |

**2.Use case name: Login**

|  |
| --- |
| **Name:**  Login |
| **Actor:** Admin, Dco,Sco, Cooperative leader. |
| **Description:** Help for Admin, Staff and Cooperative leader to enter in the system and access |
| **Pre-condition:** Being logged in to the system. |
| **Post-condition:** the system should send welcome message which contain his/her names |
| **Main Flow:** The actor presses the login button. |
| **Alternative flow:**  If the information are not valid the system shows the error where is not correct. |

**3.Use case name: Register Cooperatives**

|  |
| --- |
| **Name:**  Register Cooperative |
| **Actor:** Cooperative leader. |
| **Description:** allows a cooperative leader to a register the cooperatives into the system. |
| **Pre-condition:** the cooperative leader has to login into the system as the as a cooperative leader. |
| **Post-condition:** success message is displayed and request is sent already; |
| **Main Flow:**   1. The cooperative leader requests the system a form to register cooperative; 2. The system displays the requested form; 3. The cooperative leader send and receive their need and the system allow him to send and to receive to whom it concerns ; 4. The systems display a success message and send it the site sector cooperative officer. |
| **Alternative flow:**  If the request made by cooperative leader is failed, the system displays an error messages. |

**4.Use case name: view and Approve cooperatives**

|  |
| --- |
| **Name:**  View and Approve Cooperatives |
| **Actor:** Dco,Sco |
| **Description:** allows to Dco,Sco to view and approve cooperative information. |
| **Pre-condition:** Login as Dco and Sco. |
| **Post-condition:** success message is displayed and request is sent already; |
| **Main Flow:**   1. The Dco and Sco requests the system to display what is needed. 2. The system displays the requested information. 3. The Dco and Sco check the requested accurate information. 4. The Dco and Sco approve the requested information by reading them, |
| **Alternative flow:**  If the request made by Dco and Sco to accept cooperative is failed, the system displays an error messages. |

**5.Use case name: Lock and Unlock** Dco and Sco **.**

|  |
| --- |
| **Name:**  lock and unlock Dco and Sco . |
| **Actor:** Admin |
| **Description:** allows to Admin Lock or unlock Dco and Sco |
| **Pre-condition:** The actor has to login to the system as Admin. |
| **Post-condition:** success message is displayed . |
| **Main Flow:**   1. The Admin requests the system to view the record done ; 2. The system gives a record form; 3. Admin open the record made; 4. The system validates information; 5. Admin lock and unlock user. |
| **Alternative flow:**  If Dco and Sco is locked , the system displays an error messages. |

**6. Use case name: send message.**

|  |
| --- |
| **Name:** send message |
| **Actor:** Admin. |
| **Description:** Allows Admin to send message. |
| **Pre-condition:** Being logged in to the system. |
| **Post-condition:** the system sends a message showing the success or the failure operation. |
| **Main Flow:**   1. The system gives him/her to send message; 2. The system validates send message; 3. The system provides him/her with a message; 4. System prints/displays message; |
| **Alternative flow:**  The system sends a message indicating an error. |

**7. Use case name: receive message.**

|  |
| --- |
| **Name:** receive message |
| **Actor:** Cooperative leader. |
| **Description:** Allows Cooperative leader to receive message. |
| **Pre-condition:** Being logged in to the system. |
| **Post-condition:** the system sends a message showing the success or the failure operation. |
| **Main Flow:**   1. The system gives him/her to receive message; 2. The system validates receive message; 3. The system provides him/her with a message; 4. System prints/displays message; |
| **Alternative flow:**  The system sends a message indicating an error. |

**8. Use case name: receive Notification.**

|  |
| --- |
| **Name:** receive notification |
| **Actor:** Cooperative leader. |
| **Description:** Allows Cooperative leader to receive Notification. |
| **Pre-condition:** Being logged in to the system. |
| **Post-condition:** cooperative leader will receive a Notification showing the success or the failure operation. |
| **Main Flow:**   1. The system gives him/her to receive Notification; 2. The system validates receive Notification; 3. The system provides him/her with a Notification; 4. System prints/displays Notification; |
| **Alternative flow:** |

**9. Use case name: send Notification.**

|  |
| --- |
| **Name**: Send notification |
| **Actor**: Dco and Sco |
| **Description**: This allows the Dco and Sco to notify other users about the progress of their requests. |
| **Pre-condition**: Dco and Sco has to login with his credentials |
| **Post-condition**: the system should display the successfully message |
| **Normal flow:**  **Sending notification**   1. Dco and Sco request system to display all cooperatives. 2. System allows staff to access the table of all cooperatives. 3. Dco and Sco checks all requirements, if they are all met he notifies the sender with a success message or failure message otherwise. 4. The system stores that message in the database. |
| **Alternative flow:** |

**10.Use case name : Generate report**

|  |
| --- |
| **Name:**  Generate Report |
| **Actor:** Dco ,Sco and Admin |
| **Description:** allows a Dco and Sco and Admin to generate report . |
| **Pre-condition:** the Dco ,Sco and Admin has to login into the system |
| **Post-condition:** success message is displayed report; |
| **Main Flow:**   1. Dco , Sco and Admin requests the system to generate report ; 2. The system gives a menu of reports available; 3. The system validates Report information; 4. The system provides him/her with a report; 5. The Dco , Sco and Admin print/display report; |
| **Alternative flow:**  If report information is not valid, the system sends a message indicating an error. |

**11.Use case name: Logout**

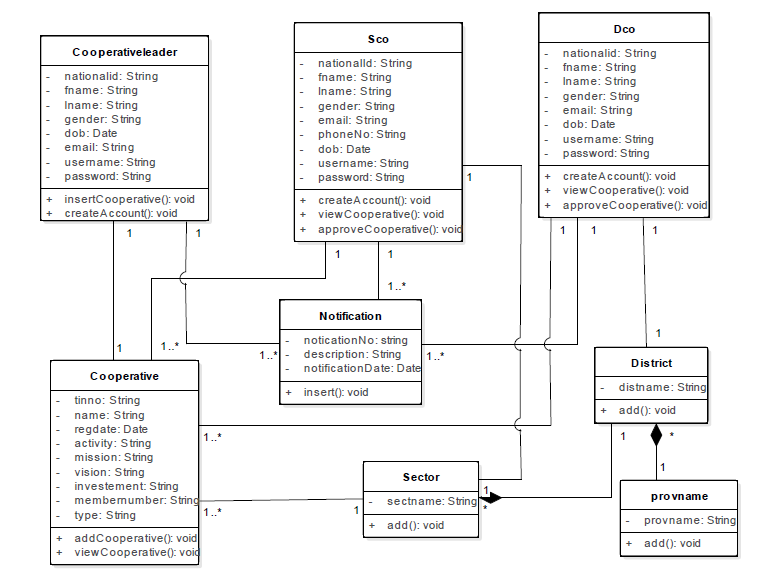
|  |
| --- |
| **Name:**  Logout |
| **Actor:** Admin, Dco and Sco Cooperative leader. |
| **Description:** Help for Admin, Dco and Sco and Cooperative leader to logout to the system . |
| **Pre-condition:** Being logout to the system. |
| **Post-condition:** the system should send message if you are sure to logout to the system. |
| **Main Flow:** The actor presses the logout button. |
| **Alternative flow:**  If you are not need to logout the system click Cancel. |

**Domain Analysis**

In software engineering, domain analysis is the process of analyzing related software to determine the operations, properties appropriate for designing solutions to problems in a given domain. The output of this process is a class diagram. Class diagram in the Unified Modeling Language (UML) is a structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes.

**Class diagram**

By Cantone, G. (2003, A class diagram is the structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. The figure below shows the class diagram of ONLINE COOPERATIVES REGISTRATION MANAGEMENT INFORMATION SYSTEM.

****

**Design of new system**

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirement. Systems design could be seen as the application of systems theory to product development. During system design, we identify design goals, decompose the system into subsystems, and refine the subsystem decomposition until all design goals are addressed.

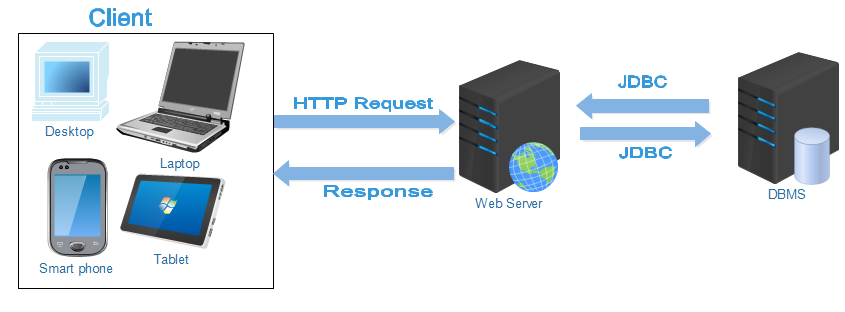
However, the impact is similar: one design goal is examined at the time, influencing the system decomposition and resulting in the change of the subsystem decomposition or its interfaces.

**System architecture diagram**

By (Griffin, W. J. 1998), The important step of design is the creation of the physical architecture layer design, the plan for how the system will be distributed across the computers and what hardware and software will be used for each computer.Most systems are built to use the existing hardware and software in the organization, so often the current architecture and hardware and software infrastructure restricts the choice. Other factors, such as corporate standards, existing site-licensing agreements, and product vendor relationships also can mandate what architecture, hardware, and software the project team must design. However, many organizations now have a variety of infra-structures available or are openly looking for pilot projects to test new architectures, hardware, and software, which enables a project team to select an architecture on the basis of other important factors.

The main element of the architectural design

1. The user interface, which runs on the user's computer (the client)
2. The functional modules that actually process data. This middle tier runs on server and is often called the application server.
3. A database management system (DBMS) that stores the data required by middle tier. This tier runs on a second server called the database server



**Figure 4: System architecture diagram**

**Sequence Diagram**

A sequence diagram shows object interactions arranged in time sequence, in a Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another in a given order. Sequence diagrams typically are associated with use case realizations in the logical view of the system under development.(Merseguer, J. 2002, July)

The symbols below are used in sequence diagram:

|  |  |
| --- | --- |
| **Term and definition** | **Symbol** |
| **An actor:**   * It can be a person or system that derives benefit from and is external to the system. * It participates in a sequence by sending or receiving messages. * It is placed across the top of the diagram. | C:\Users\Niyigena Clemence\Desktop\actor.PNG |
| **An object lifeline:**   * It participates in a sequence by sending or receiving messages. * It Is placed across the top of the diagram. |  |
| **An activation**:   * It is a long narrow rectangle placed on top of a lifeline. * It denotes when an object is sending or receiving messages |  |
| **A message**:   * It conveys information from one object to another one. * An operation call is labeled with the message being sent and a solid arrow, whereas a return is labeled with the value being returned and shown as a dashed arrow. | :Message send:asynchronous  Message return  : Message call |

**Sequence Diagram for create account to the system**

 **Figure 5: Sequence diagram for create account**



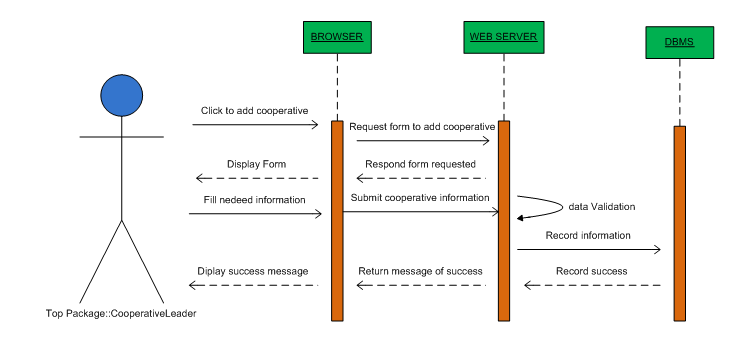
**Figure 6: Sequence Diagram for login to the system**

**Sequence diagram for Record Dco and Sco**



**Figure 7: Sequence diagram for Record Dco and Sco**

**Sequence diagram to record Cooperative**

****

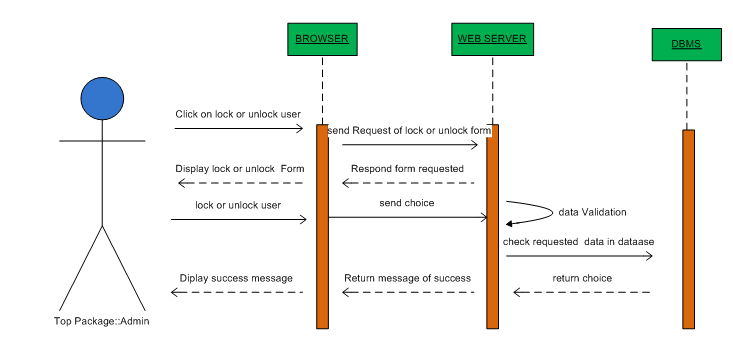
**Figure 8: Sequence diagram for Record cooperatives**

**Sequence Diagram for view and approve Cooperatives**

## 

**Figure 9: Sequence diagram for view and approve cooperatives**

**Sequence Diagram for locking /unlocking user**

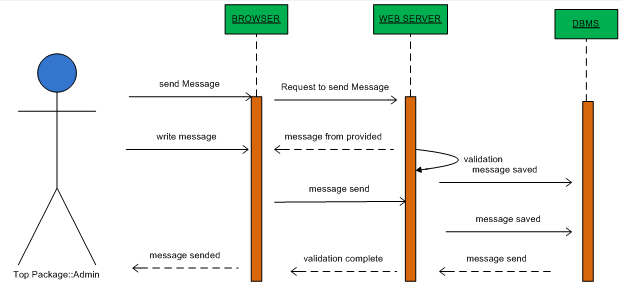


**Figure 10: Sequence Diagram for locking/unlocking user**

**Sequence diagram for generating Report**

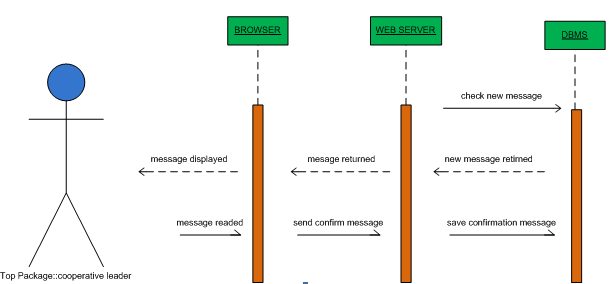
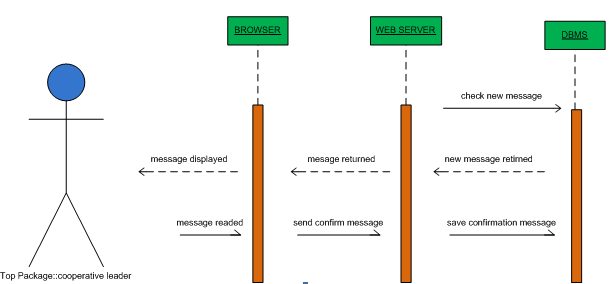
**Figure 10:** Sequence Diagram for Generating Report

Sequence Diagram for Send message

****

**Figure 11:** Sequence Diagram for Send message

Sequence Diagram for Receive message

****

**Figure 12:** Sequence Diagram for Receive message

**Database design**

**A database** is a collection of related data. By data, we mean known facts that can be recorded and that have implicit meaning. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a Data Definition Language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity.

The term database design can be used to describe many different parts of the design of an overall database system. Principally, and most correctly, it can be thought of as the logical design of the base data structures used to store the data. In the relational model these are the tables and views. In an object database the entities and relationships map directly to object classes and named relationships. However, the term database design could also be used to apply to the overall process of designing, not just the base data structures, but also the forms and queries used as part of the overall database application within the database management system (DBMS).

**Database schema diagram**

A **database schema** of a database system is its structure described in a formal language supported by the database management system (DBMS) and refers to the organization of data as a blueprint of how a database is constructed (divided into database tables in case of Relational Databases. The formal definition of database schema is a set of formulas (sentences) called integrity constraints imposed on a database. These integrity constraints ensure compatibility between parts of the schema.



# CHAPTER 4

# IMPLEMENTATION OF THE NEW SYSTEM

## Introduction

This chapter describes the development of “ONLINE COOPERATIVES REGISTRATION MANAGEMENT INFORMATION SYSTEM”. It includes a brief overview of the technologies used to make the application, operational, the presentation of screenshots for the new system, tests that have been applied. Last but not least, software and hardware compatibility requirements.

## Technologies used

To develop this application we have used different technologies and tools namely:

* + **NetBeans IDE 8.0.2** for the conception of the graphic interfaces allowing the users to interact with the new system.
  + **MySQL** used for the creation of the database.
  + **JSP Technology** used for programming.
  + **CSS** used for the fonts and web pages styles
  + **iText:** used for reports printing.

**Netbeans IDE 8.0.2**

A programming tool or software development tool is a program or application that software developers use to create, debug, maintain, or otherwise support other programs and applications.

NetBeans IDE is a software development tool which is an open-source integrated development environment which supports development of all Java application types (Desktop Application, web application, and Applets).

Java is a programming language originally developed by James Gosling at Sun Microsystems and released in 1995 as a core component of Sun Microsystems' Java platform. One characteristic of Java is its portability, which means that computer programs written in the Java language can run similarly on any hardware/operating-system platform. Java is a pure object-oriented language which means that everything in a Java program is an object and everything is descended from a root object class.

**JSF Technology**

**JavaServer Faces** (**JSF**) is a Java specification for building component-based user interface for web applications and was formalized as a standard through the Java Community Process being part of the Java Platform, Enterprise Edition. It is also a MVC web framework that simplifies to construct user interfaces (UI) for server-based applications by using reusable UI components in a page.

**MySQL**

MySQL is relational database management system which is widely used around the globe due to it being open- sourced. It stores data in separate tables rather than putting all the data in one big storeroom. This adds speed flexibility.

Officers a rich and useful set of functions its connectivity, speed and security make it highly suited for accessing databases on the internet. MySQL is also mostly platform independent which means it can run on most operating system such as Windows and Linux. We have used this tool for the creation of the database (Oracle, 2010)

**iText Reports**

A report is a nicely formatted way of presenting the data that you have entered, reports are all about querying a database and displaying the results in a nice format. ItextReport is an open source Java reporting tool that can be used in Java enabled applications, including web applications, to generate dynamic content. It has the ability to deliver rich content in various formats such as PDF, HTML, XML files, or directly on the screen or printer.

ItextReport is a content-rendering library, not a standalone application. It cannot run on its own and must be embedded in another client or server-side Java application. ItextReport is a pure Java library and can be used on any platform that supports Java. Because ItextReport is a library and cannot run on its own, you do not really install it.“Installing” ItextReport simply means downloading its JAR file and putting it into the Class path of your application along with the other required JAR files.

Generating reports is a common, if not always glamorous, task for programmers. In the past, report generation has largely been the domain of large commercial products. Today, the open source ItextReport report generating library gives Java developers a viable alternative to commercial software. ItextReport provides the necessary features to generate dynamic reports, including data retrieval using JDBC (Java Database Connectivity), as well as support for parameters, expressions, variables, and groups (Vogel, 2008).

**Cascading style sheets**

Cascading style sheets are used to format the layout of Web pages. They can be used to define text styles, table sizes, and other aspects of Web pages that previously could only be defined in a page's HTML.CSS helps Web developers create a uniform look across several pages of a Web site. Instead of defining the style of each table and each block of text within a page's HTML, commonly used styles need to be defined only once in a CSS document. Once the style is defined in cascading style sheet, it can be used by any page that references the CSS file.

CSS makes it easy to change styles across several pages at once. While CSS is great for creating text styles, it is helpful for formatting other aspects of Web page layout as well. For example, CSS can be used to define the cell padding of table cells, the style, thickness, and color of a table's border, and the padding around images or other objects. CSS gives Web developers more exact control over how Web pages will look than HTML does. This is why most Web pages today incorporate cascading style sheets (Håkon Wium Lie and Bert Bos, 1999).