Web Based Management System

For

Department Of Agrarian Development, Uduvil

Kavitha S

2017

Web Based Management System

For

Department Of Agrarian Development, Uduvil

Kavitha S

BIT Registration Number: R101238

Index Number: 1012381

Name of the supervisor: Mr.K.Venugoban

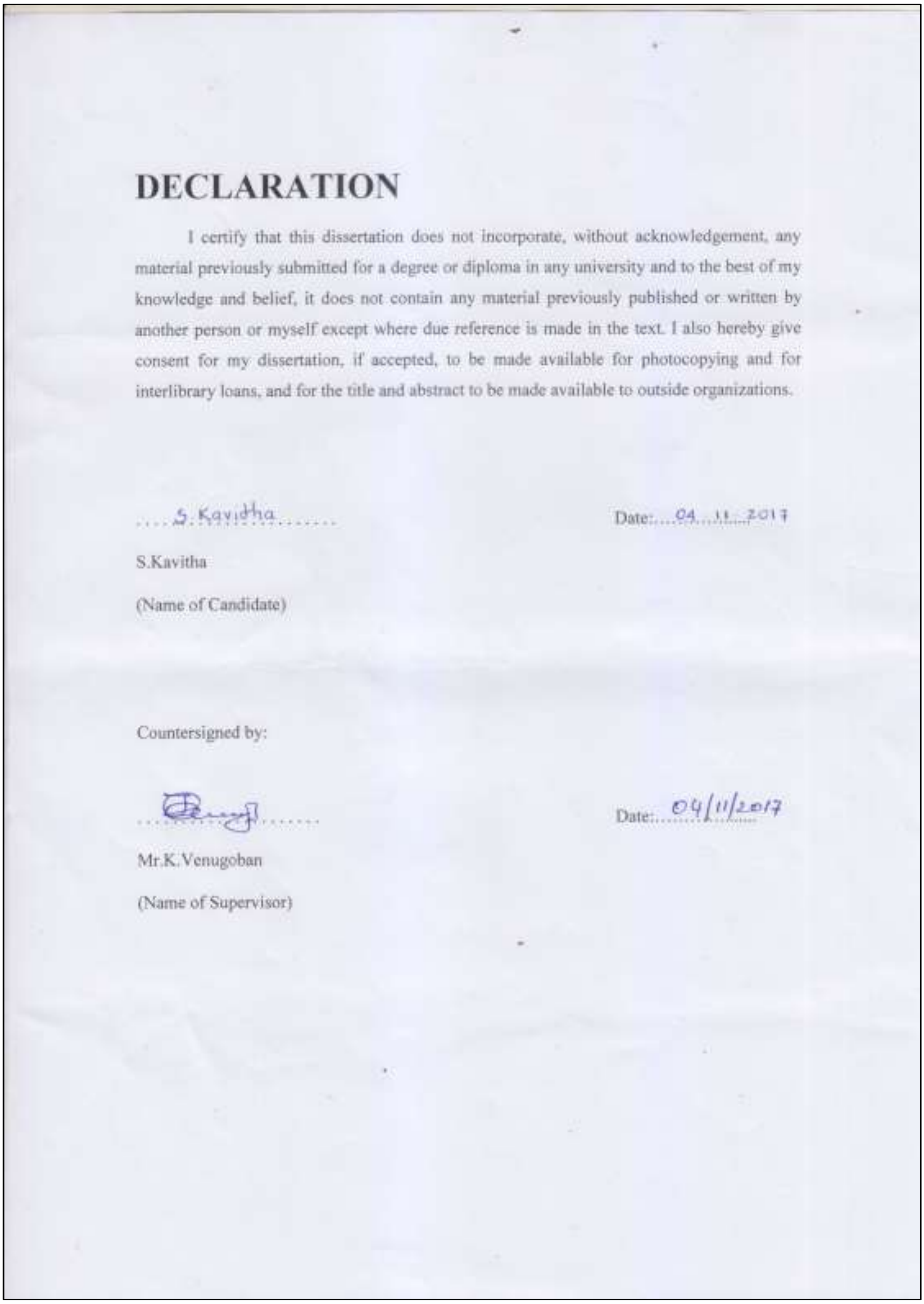
2017

This dissertation is submitted in partial fulfillment of the requirement of the

Degree of Bachelor of Information Technology (external) of the

University of Colombo School of Computing

DECLARATION



DECLARATION

I certify that this dissertation does not incorporate, without acknowledgement, any material previously submitted for a degree or diploma in any university and to the best of my knowledge and belief, it does not contain any material previously published or written by another person or myself except where due reference is made in the text. I also hereby give consent for my dissertation, if accepted, to be made available for photocopying and for interlibrary loans, and for the title and abstract to be made available to outside organizations.

....S.Kavitha....... Date:....04..11..2017

S.Kavitha

(Name of Candidate)

Countersigned by:

.... Date:..04/11/2017

Mr.K.Venugoban

(Name of Supervisor)

ABSTRACT

Valikamam South Divisional Secretariat Division is the Centre part of Jaffna District in Sri Lanka. Particularly agriculture development, agro-based industrial developments are suitable for this division. Uduvil is an agricultural village of about 32.7 square kilometers situated along the KKS Road at about five miles to the North of Jaffna city, Sri Lanka. The Information System is developed for the Uduvil Agrarian Department to manage their activities.

The Information System is developed for the Uduvil Agrarian Department to manage their activities. In this competitive world the Uduvil Agrarian Department mostly occupied with paper work. The department management faces many difficulties in manual works such as manage records of day to day routine activities, manage farmer details, manage farmer bank accounts and manage relief details. So they would like to implement web based business management system to manage services to the farmers. This project provides effective and efficient system to the web based business management system for the department. This web based business management system help to minimize the workload, increase efficiency of work, identify the day to day activities and view the daily, monthly and annual reports.

The system consists of the web site for Agrarian Service Centre and its management information system. It is possible for authorized personnel to gain access to the management information system via the Agrarian department web site. This project enables the management of the department to update the farmer’s details with ease. This can enable the management to keep track of farmer’s details effectively and run the department in a better way. This web based information system was developed using PHP: Hypertext preprocessor, which is a powerful server side scripting language, and Apache as the webserver. My SQL was used as the Database management system. This developed system is thoroughly tested to guarantee as a validated system.

ACKNOWLEDGEMENT

I take this space to acknowledge and extend my heartfelt gratitude to those who have helped me in various ways throughout the project work to make this project a reality. First and foremost I owe my deep gratitude to the University Of Colombo School Of Computing for offering us this precious degree program and all the staff who guided us from the beginning. A very special recognition needs to be given to my project supervisor Mr. Kanesh Venugoban for his extensive assistance, without which the completion of this project would have been extremely complicated.

I must thank Manager and all the staff of Agrarian Service Center, Uduvil who gave me the opportunity to develop this system for their center and gave me an enormous support for the successful completion of the project. I would like to thank Mr. Subramaniam Theivamainthan for extending his support to write this dissertation. I dedicate this dissertation to my family for their unconditional love and support given in every way possible throughout the process of this degree program of three years. I would like to memorize my friends who have given me advice and encouragement throughout the project. I’m much grateful to them too.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ACRONYMS

1NF / 2NF / 3NF – First / Second / Third Normal Form

BIT – Bachelor of Information Technology

CSS – Cascading Style Sheet

DVD – Digital Versatile Disc

ER – Entity Diagram

GB – Gigabyte

GUI – Graphical User Interface

HTTP – Hypertext Transfer Protocol

OO – Object Oriented

PHP – Hypertext Pre-Processor

RAM – Random Access Memory

SMS – Short Message Service

UCSC – University of Colombo School of Computing

CHAPTER 1: INTRODUCTION

Valikamam South Divisional Secretariat Division is the Centre part of Jaffna District in Sri Lanka. Particularly agriculture development, agro-based industrial developments are suitable for this division. Uduvil is an agricultural village of about 32.7 square kilometers situated along the KKS Road at about five miles to the North of Jaffna city, Sri Lanka. The Information System is developed for the Uduvil Agrarian department to manage their activities.

1.1 MOTIVATION FOR PROJECT

Sri Lanka, affectionately called as the pearl of the Indian Ocean, is a tear drop shaped island of 65,000sq KM surrounded by sparkling blue green sea. Moderate climate throughout the year, fertile soil, freely available ground water with fairly distributed river network in most parts and two monsoons which bring rains for the two main cultivation seasons are the key ecological factors of the country. With these ideal conditions, agriculture sector has always performed as a major economic force in Sri Lanka, making a significant contribution to the national economy, food security and employment. At the same time agriculture is the livelihood of the majority in the rural sector and plays a key role in alleviating rural poverty. This has been well recognized from the time of independence and there has always been a cabinet portfolio set aside for the agriculture sector

The Information System is developed for the Uduvil Agrarian Department to manage their activities. In this competitive world the Uduvil Agrarian Department mostly occupied with paper work. The department management faces many difficulties in manual works such as manage records of day to day routine activities, manage farmer details, manage farmer bank accounts and manage relief details. So they would like to implement web based business management system to manage services to the farmers. This project provides effective and efficient system to the web based business management system for the department. This web based business management system help to minimize the workload, increase efficiency of work, identify the day to day activities and view the daily, monthly and annual reports.

1.2 OBJECTIVES OF THE PROJECT

- Manage farmer and farming place details.
- Manage the farmer details and their farming place details and get the details in finger tips.
- Manage farming organization, organization member details
- Every Girama Nilathari has a farming organization and it has a community group, this organization provides a service to farmer.
- Manage relief details of recycling paddy cultivation, give money to farmer for their farming.
- Manage farmer bank account and transaction details
- Keep records of banking details and manage daily routine transaction details.
- Manage disaster relief , loan details
- Supply money, agriculture inputs and agriculture instruments based on their disaster evaluation or insurance or loan to farmer.
- Calculating loan and crediting term interest for farmer
- Calculate the interest for loan and send alert to farmer about loan details.
- Quickly and fully efficient Reporting
- Generate various categories in daily or monthly or yearly report
- Communicate within system through message via SMS or E-Mail.
- Send alert SMS to farmer regarding loan, interest, relief details.
- By this system maintain the time management

1.3 SCOPE

The scope of this project is manage the routine activities of Department of Agrarian Development, Uduvil such as farmers details, farming place details, paddy cultivation relief details, loan details, supply agriculture input details and etc. This web based management system to provide without difficulties to manage from their place and provide their day to day activities.

1.4 STRUCTURE OF THE DESSERATION

The rest of this report is organized as follows. Chapter 2 provides the Analysis, This chapter describes the existing system, requirements and feasibility study for the proposed system and the process model used to develop the system. Chapter 3 provides a design including Use case diagram, ER Diagram, Activity diagram and class diagram. In chapter 4, the implementation of the project work. In chapter 5 4 describe evaluation of project work. Finally, chapter 6 concludes this work with a discussion of our findings towards future extensions.

After the main chapters there is a Reference section where all the materials referred to write the dissertation are given. Furthermore in the appendices, System Documentation, Design Documentation, User Documentation, Management Reports, Test Results, Code Listing and the Client Certificate are provided. Finally, a glossary of terms is provided.

CHAPTER 2: ANALYSIS

The system analysis chapter explains about the requirements gathering techniques and collected functional, non-functional requirements to design and develop the system. Top level Use-Case diagrams are used to make it easier to understand the system.

2.1 EXISTING MANUAL SYSTEM

Management System for Agrarian department is done by manually. Since this system fully engaged in paper work. It fines many difficulties in manual work such as managing day to day records and activities, generating reports, keeping farmers details, and using many paper documents etc. The manual Administration system contains human errors as calculation errors. In this manual method is a very time consuming process with the need of considerable amount of man powers.

2.1.1 DRAWBACKS OF THE SYSTEM

The following major drawbacks have been identified in the existing manual management system.

- Difficult to Manage the farmer details and their farming place details and get the details in finger tips.
- Difficult to Manage farming organization, organization member details, farmer bank account and transaction details
- Difficult to Generate various categories in daily or monthly or yearly report
- Time wasting paper work
- Critical calculations are done manually
- No backups for the confidential data
- Poor communication methods between the farmers

Manual Use case diagram for “Agrarian Department” as follows. The use-case diagram shown as following Figure 2.1 Existing Use-case Diagram

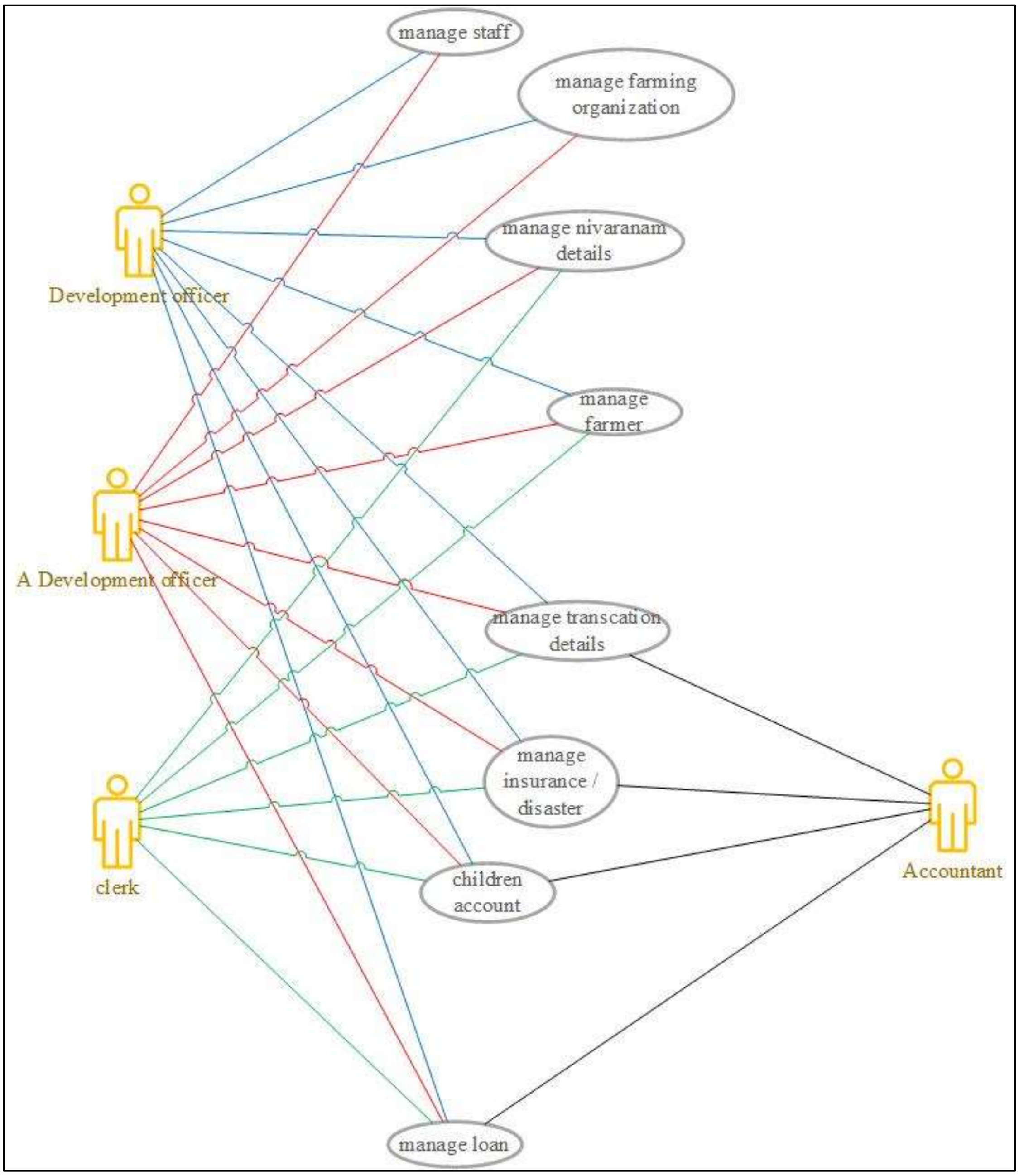


Figure 2.1 Existing Use case Diagram

2.2 INFORMATION GATHERING TECHNIQUES

In the software development, analysis part is very important to collect correct and accurate requirements for this system and analysis the requirements and identify the solution. Some information gathering techniques use to collect requirements are available such as, interviews, observation, questionnaires, site visit and etc. but in this study interviews, observations and site visit are used to find requirements.

Site visit & Observation is the best technique to gather valuable data. This technique is used to fulfill the requirements of the "Agrarian department" day to day activities. Client's office has been visited to observe the problems they are facing on the manual system. This site visit gave us many ideas to develop the new system in an effective manner.

2.2.1 FUNCTIONAL REQUIREMENTS

Functional requirements are capture the intended behavior of the system. This behavior may be expressed as services, tasks or functions the system is required to perform. In product development, it is useful to distinguish between the baseline functionality necessary for any system to compete in that product domain, and features that differentiate the system from competitors.

REQUIREMENTS OF THE AGRARIAN DEPARTMENT

- Manage farming organization, organization member details, the details of farmers and their farming place details.
- Manage paddy cultivation relief details.
- Manage farmer bank account and transaction details; Obtaining periodical transaction report view and printouts.
- Calculating loan and crediting term interest for farmer
- Quickly and fully efficient Reporting in various category such as yearly or monthly or daily.
- Alert news regarding loan, relief, pending payment, disaster details and etc. to the farming organization's members and farmers.
- Manage and publish latest new and events and upload the photos and videos of the event in web.
- Get users feedback regarding web site and farming organization's services..

2.2.2 NON FUNCTIONAL REQUIREMENTS

Nonfunctional requirements in systems engineering and requirement engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors.

- Accuracy and Consistency

  These are very important nonfunctional requirements that should be considered when storing the details of customers and company calculating values.

- Security

  There should be superior security mechanisms since the system stores very confidential information such as details of user profile, password of the customers and company account.

- Usability

  Usability should be very much important in this project when developing the online user profiles to the customers.

- Reliability

  This is a nonfunctional requirement of the system users. There should be trustworthiness between the users and the system.

- Reusability and Maintainability

  In a case where the system needs any changes in the future, it should not be a risky task. Proper documentation and using standard methods when developing the system will ensure this nonfunctional requirement.

  The following requirements were also nonfunctional requirements:

  - Authorized person only login into the system.
  - Reduce the labor cost and time.
  - Easy to access the system and should be user friendliness.
  - System should be accurate and fast.
  - Computer Resources should be used in an effective manner.

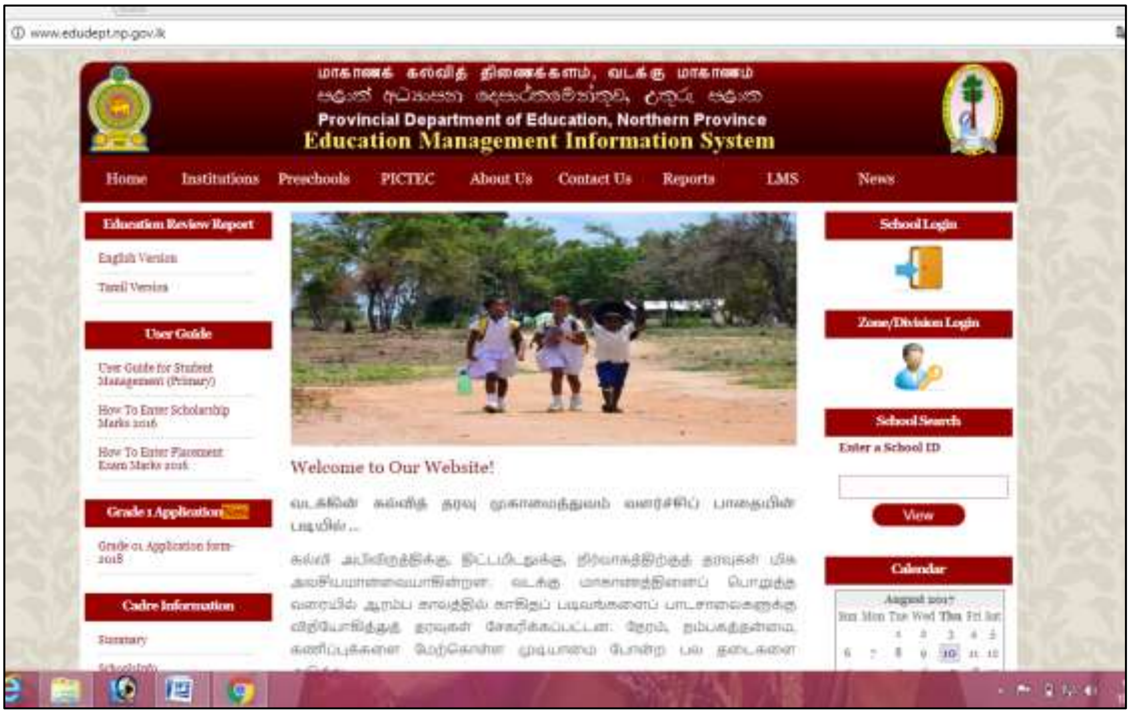
2.3 SIMILAR SYSTEM

This section summarizes various techniques that have been employed closely related with this system. This system has some literature reviews that are on agrarian department systems. The following similar systems were used for detailed study.

NORTHERN EDUCATION MANAGEMENT SYSTEM

Data and information are very vital factors of efficient education planning, development and management processes. The earlier practice of collecting data through printed forms had its drawbacks such as delays, data losses and time consuming processing methods. Therefore a standalone digital data base was designed and the data collection was administered through compact discs which were distributed among Education Zones to enter their raw data at their offices and return in a specified time interval. Even though this method was advantageous than the earlier methods it still had several shortfalls such as technical difficulties in handling, different approaches of raw data collection from schools and processing delays. Considering these challenges it was attempted to maintain an offline locally distributed data base in two locations to which the officials should physically travel and enter data. As the Divisional Education Officials are the nearest managers to schools they were entrusted with the responsibility of collecting data and updating these data bases. They could update the system bimonthly. Based on this data base, a bimonthly publication of vital statistics has been issued since 2010. And now this online data base is introduced with the view to progress towards a comprehensive education management information system.

The Northern education management system [1] home page is shown in the following Figure 2.2



www.edudept.np.gov.lk

மாகாணக் கல்வித் திணைக்களம், வடக்கு மாகாணம்
පළාත් අධ්‍යාපන දෙපාර්තමේන්තුව, උතුරු පළාත
Provincial Department of Education, Northern Province
Education Management Information System

Education Review Report

English Version

Tamil Version

User Guide

User Guide for Student Management (Primary)

How To Enter Scholarship Marks 2016

How To Enter Placement Exam Marks 2016

Grade 1 Application New

Grade 01 Application form-2018

Cadre Information

Summary

SchoolsInfo

Welcome to Our Website!

வடக்கின் கல்வித் தரவு முகாமைத்துவம் வளர்ச்சிப் பாதையின் படியில்...

கல்வி அபிவிருத்திக்கு, திட்டமிடலுக்கு, நிர்வாகத்திற்குத் தரவுகள் மிக அவசியமானவையாகின்றன. வடக்கு மாகாணத்தினைப் பொறுத்த வரையில் ஆரம்ப காலத்தில் காகிதப் படிவங்களைப் பாடசாலைகளுக்கு விநியோகித்துத் தரவுகள் சேகரிக்கப்பட்டன. நேரம், நம்பகத்தன்மை, கணிப்புக்களை மேற்கொள்ள முடியாமை போன்ற பல தடைகளை

School Login

Zone/Division Login

School Search

Enter a School ID

Calendar

August 2017

Sun Mon Tue Wed Thu Fri Sat
1 2 3 4 5
6 7 8 9 10 11 12

Figure 2.2 Northern education management system home page

CHARITY NAVIGATOR SYSTEM

One of the famous charity organization "Charity Navigator" [2] good service organization in America. They facilities education, environment, health, human services, human and civil rights, religion, arts culture humanities and community development and etc. The front page of the Charity Navigator shown in the following Figure 2.3 Charity Navigator:

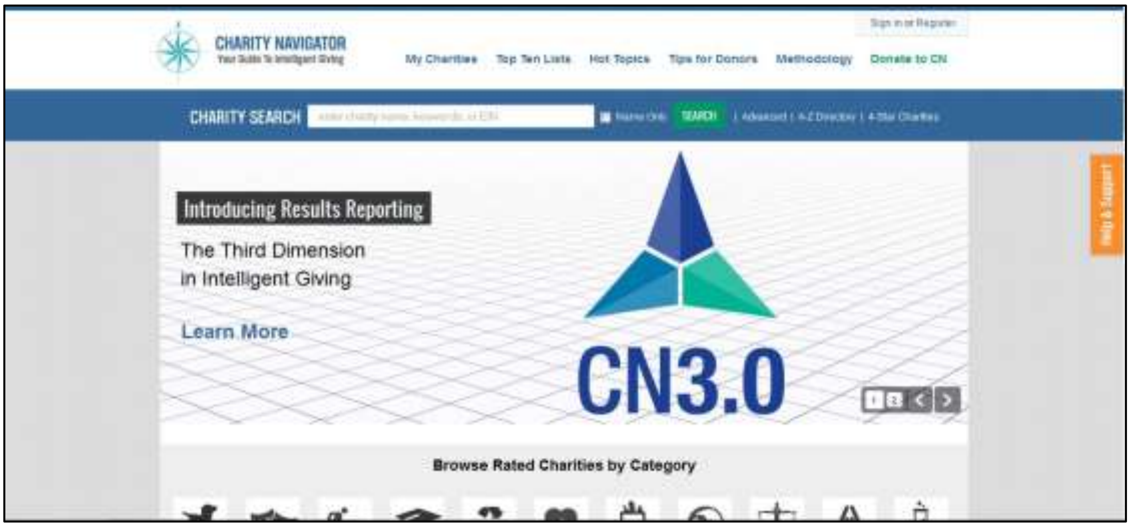


Figure 2.3 Charity Navigator

AGRICULTURE DEPARTMENT SYSTEM

In Sri Lanka, there is a web site for Agriculture department [3] but it’s containing only news, events and projects details. From this site we cannot manage agriculture department details. The Figure 2.4 is shown the homepage of Agriculture department:

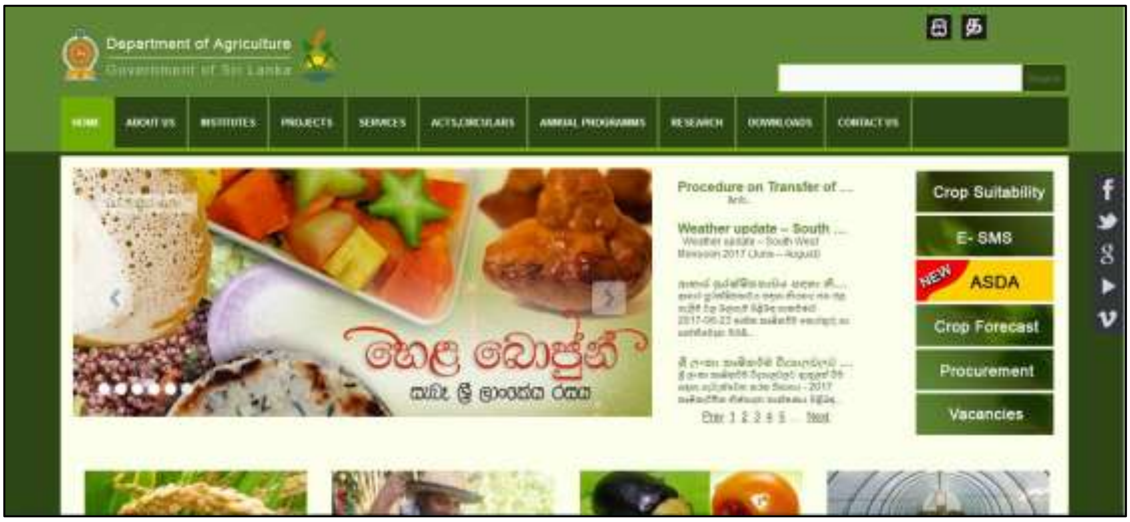


Figure 2.4 Agriculture Department

CHAPTER 3: DESIGN

System design focused on the technical or implementation aspect of the system development project. System analysis phase of the system development project is implementation independent. Design is much more creative process than analysis. System designer converts the requirements from the requirement analysis phase into technical solutions. System design considers the software architecture, database design and interfaces design. There are several techniques exist to design a system, Such as:

- Structured design techniques
- Object oriented design techniques
- Rapid Application techniques
- Joint Application Development
- Prototyping

3.1 PROCESS MODELS

STRUCTURED PROGRAMMING

Structured programming is a programming paradigm aimed at improving the clarity, quality, and development time of a computer program by making extensive use of subroutines, block structures, for and while loops—in contrast to using simple tests and jumps such as the go to statement, which could lead to "spaghetti code" that is difficult to follow and maintain.

OBJECT ORIENTED PROGRAMMING

Object-oriented programming (OOP) is a programming paradigm based on the concept of "objects", which may contain data, in the form of fields, often known as attributes; and code, in the form of procedures, often known as methods. A feature of objects is that an object's procedures can access and often modify the data fields of the object with which they are associated. In OOP, computer programs are designed by making them out of objects that interact with one another. There is significant diversity of OOP languages, but the most popular ones are class-based, meaning that objects are instances of classes, which typically also determine their type.

RAPID APPLICATION TECHNIQUES

Rapid Application Development (RAD) is both a general term, used to refer to adaptive software development approaches, as well as the name for James Martin's approach to rapid development. In general, RAD approaches to software development put less emphasis on planning and more emphasis on an adaptive process. Prototypes are often used in addition to or sometimes even in place of design specifications. RAD is especially well suited for (although not limited to) developing software that is driven by user interface requirements. Graphical user interface builders are often called rapid application development tools. Other approaches to rapid development include the adaptive, agile, spiral, and unified models.

JOINT APPLICATION DESIGN

Joint application design (JAD) is a process used in the life cycle area of the dynamic systems development method to collect business requirements while developing new information systems for a company. The JAD process also includes approaches for enhancing user participation, expediting development, and improving the quality of specifications.

PROTOTYPING MODEL

Prototyping is used to allow the users evaluate developer proposals and try them out before implementation. It also helps understand the requirements which are user specific and may not have been considered by the developer during product design.

Software Prototyping is most useful in development of systems having high level of user interactions such as online systems. Systems which need users to fill out forms or go through various screens before data is processed can use prototyping very effectively to give the exact look and feel even before the actual software is developed.

The prototyping model [4] is shown in the following Figure 3.1:

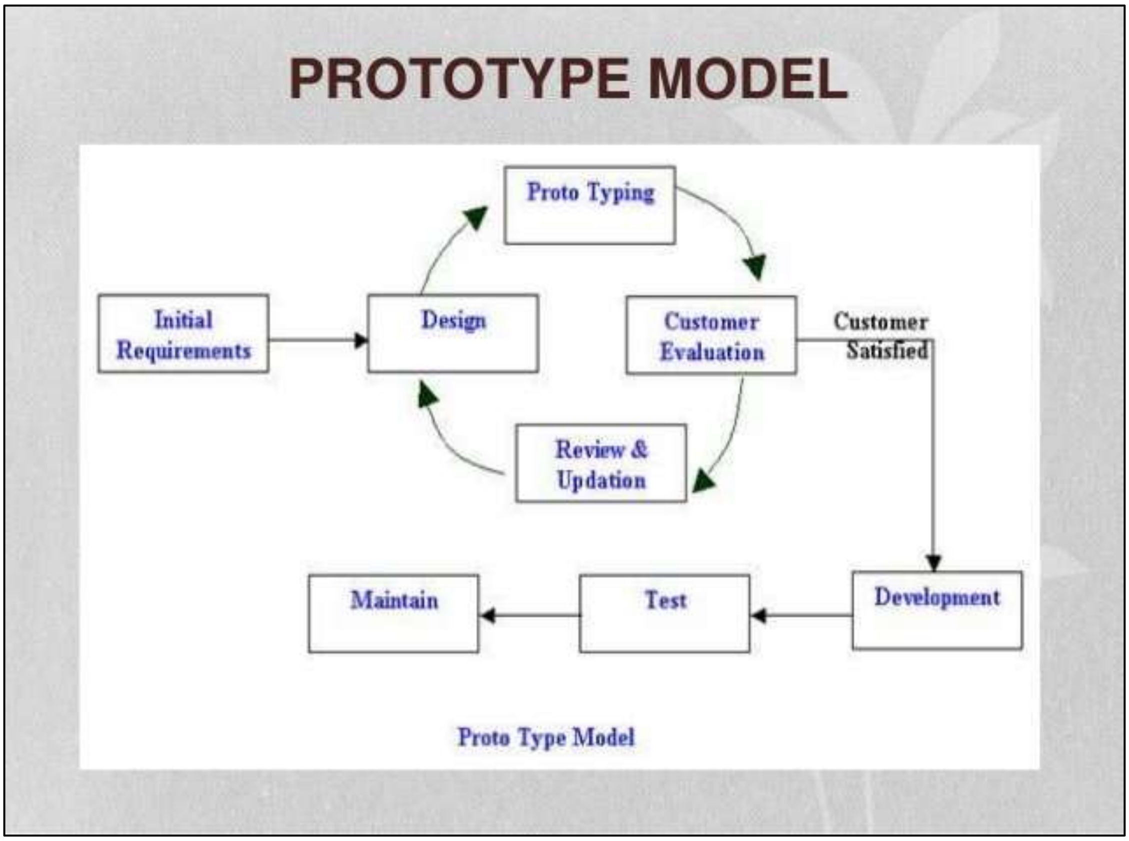


Figure 3.1 Prototype model

THE ADVANTAGES OF THE PROTOTYPING MODEL ARE AS FOLLOWS

- Increased user involvement in the product even before its implementation.
- Since a working model of the system is displayed, the users get a better understanding of the system being developed.
- Reduces time and cost as the defects can be detected much earlier.
- Quicker user feedback is available leading to better solutions.
- Missing functionality can be identified easily.
- Confusing or difficult functions can be identified.

THE DISADVANTAGES OF THE PROTOTYPING MODEL ARE AS FOLLOWS

- Risk of insufficient requirement analysis owing to too much dependency on the prototype.
- Users may get confused in the prototypes and actual systems.
- Practically, this methodology may increase the complexity of the system as scope of the system may expand beyond original plans.

- Developers may try to reuse the existing prototypes to build the actual system, even when it is not technically feasible.
- The effort invested in building prototypes may be too much if it is not monitored properly.

WHY USE TO PROTOTYPE MODEL?

The Agrarian Department was government sector that will chance to change their requirements so this will suitable for unstable requirements SDLC so prototype is suitable for this system. Prototype model should be used when the desired system needs to have a lot of interaction with the end users. Typically, online systems, web interfaces have a very high amount of interaction with end users, are best suited for Prototype model. It might take a while for a system to be built that allows ease of use and needs minimal training for the end user. Prototyping ensures that the end users constantly work with the system and provide a feedback which is incorporated in the prototype to result in a useable system. They are excellent for designing good human computer interface systems.

3.2 ALTERNATIVE SOLUTIONS

The software development systems can be developed into three basic categories, which are standalone, network based and web based system. When discussing an alternative solution for this system, it is considered about the system requirements, users and departments those are use this system.

Reasons	**Web Based**	**Windows Based**	**Network Model**
Easy to link the branches	✓	-	✓
Apply the applications in online	✓	-	-
Easy to view the progress of application	✓	-	✓
Generate the reports based on user levels	✓	-	-
Easy to transfer the data to other branches	✓	-	-
Get the reminder or alert news via email or SMS	✓	-	-
User friendly facility	✓	✓	✓
Database is centralized, maintenance is easy to carry out	✓	✓	✓

Table 3.1 Alternative solution

Further web based applications are the better way to take advantage of today's technology to enhance organizations efficiency. Web based application gives an opportunity to access information from anywhere and at any time. It also facilitates to save time, money and improve the interactivity among the staffs. Web-based applications are easy to use and can be implemented with minimal cost of the hardware. And the Table 3.1 helps to justify the appropriate alternate solution.

Therefore it was decided to develop a web based application as a solution for this system.

3.3 PROCESS DESIGN

Based on the requirement analysis made initially. It has been designed in three stages as follows.

1. Database Design
2. Application & Architecture Design
3. Interface Design

3.3.1 DATABASE DESIGN

Data modeling technique is used to design the database. Database normalization is done to minimize the data redundancy. Though there are normalization can be done up to 6th normalization. Up to 3rd normalization is enough to remove unwanted columns that are not depended on primary key.

First normal form (1NF) is done to eliminate duplicate columns in a same table. It divides the single table into double as it eliminates the columns. Second normal form (2NF) is done to determine any non-function keys that are partially depend on just a part of the primary key. Then the separated tables are joined with the foreign key. Third normal form (3NF) is done to remove the columns that are not depended on the primary key. After done the third normal form (3NF) tables without any data redundancy as follows in the E.R diagram

The E.R diagram shown as following Figure 3.2E.R diagram

.

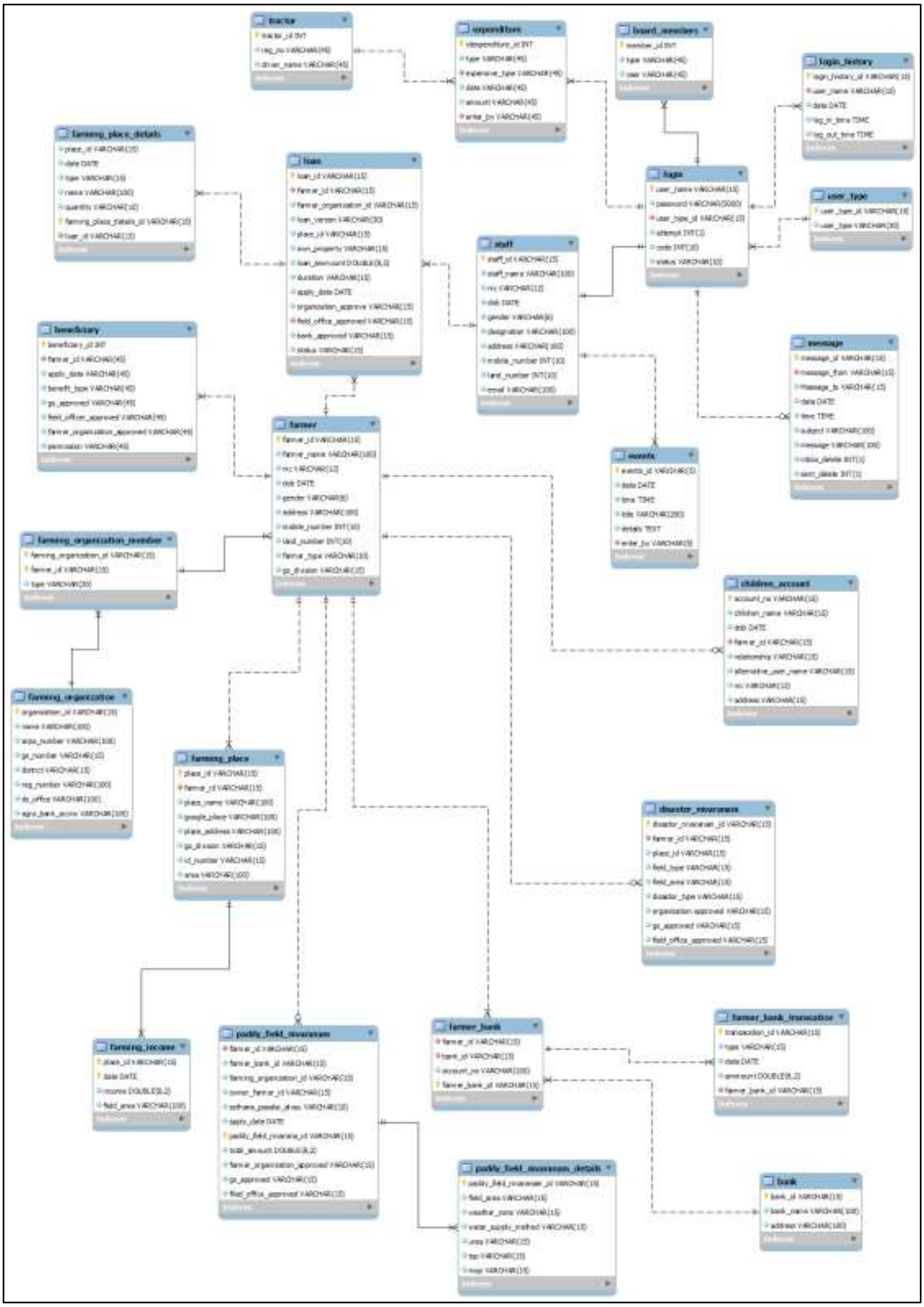


Figure 3.2 ER Diagram

3.3.2 APPLICATION & ARCHITECTURE DESIGN

Application architecture design describes the structure and behavior of this system. Activity diagram, Class diagram, Sequence diagram, Use case diagrams are drawn to illustrate the structure and behavior of the system. Microsoft Visio 2013 is used to design the above diagrams

USE CASE DIAGRAM

Use case diagram is a simplest that illustrates the users' interaction with the system. It portrays the different type of users in the system and various ways that they interact with the system. Use case diagram for “Agrarian Department” as follows. The use-case diagram shown in the following Figure 3.3 Use-case Diagram:

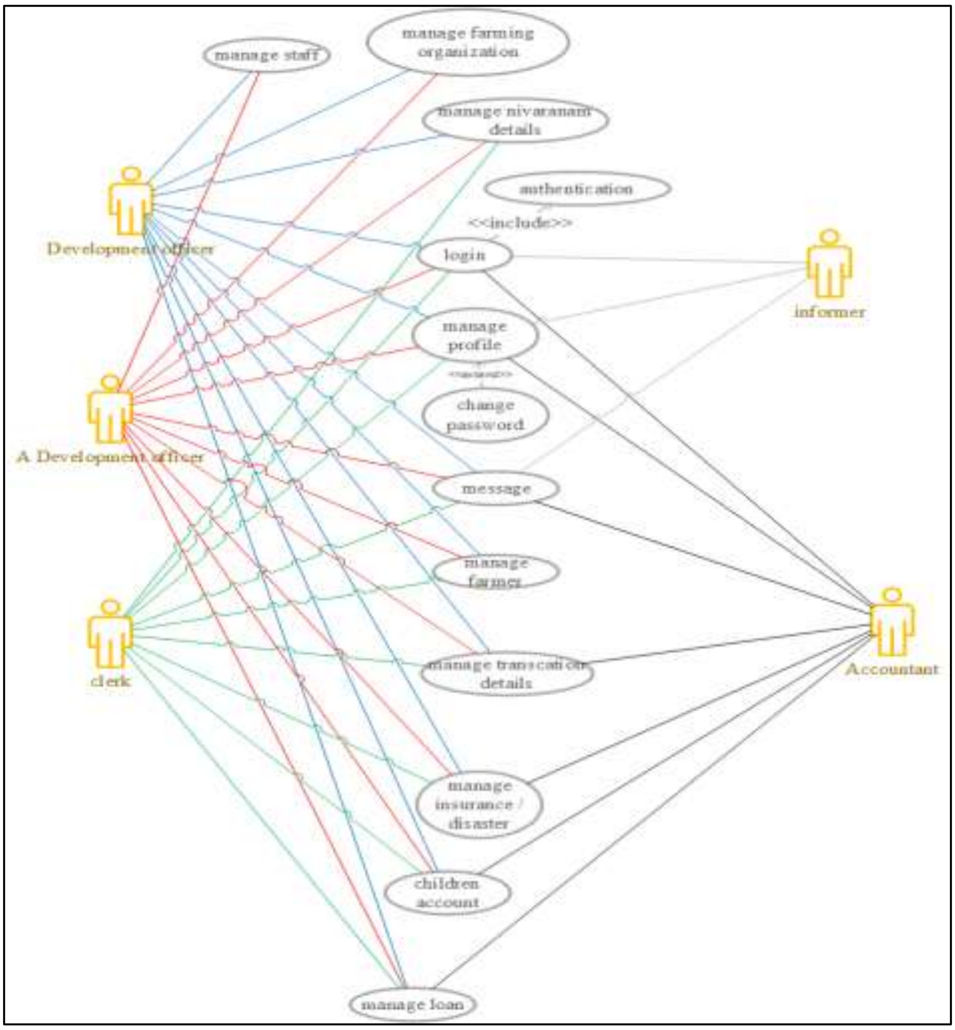


Figure 3.3 Use case Diagram

ACTIVITY DIAGRAM

Activity diagrams are graphical representation of workflows of stepwise activities and actions. Activity diagram for login process as shown in Figure 3.4 Activity diagram

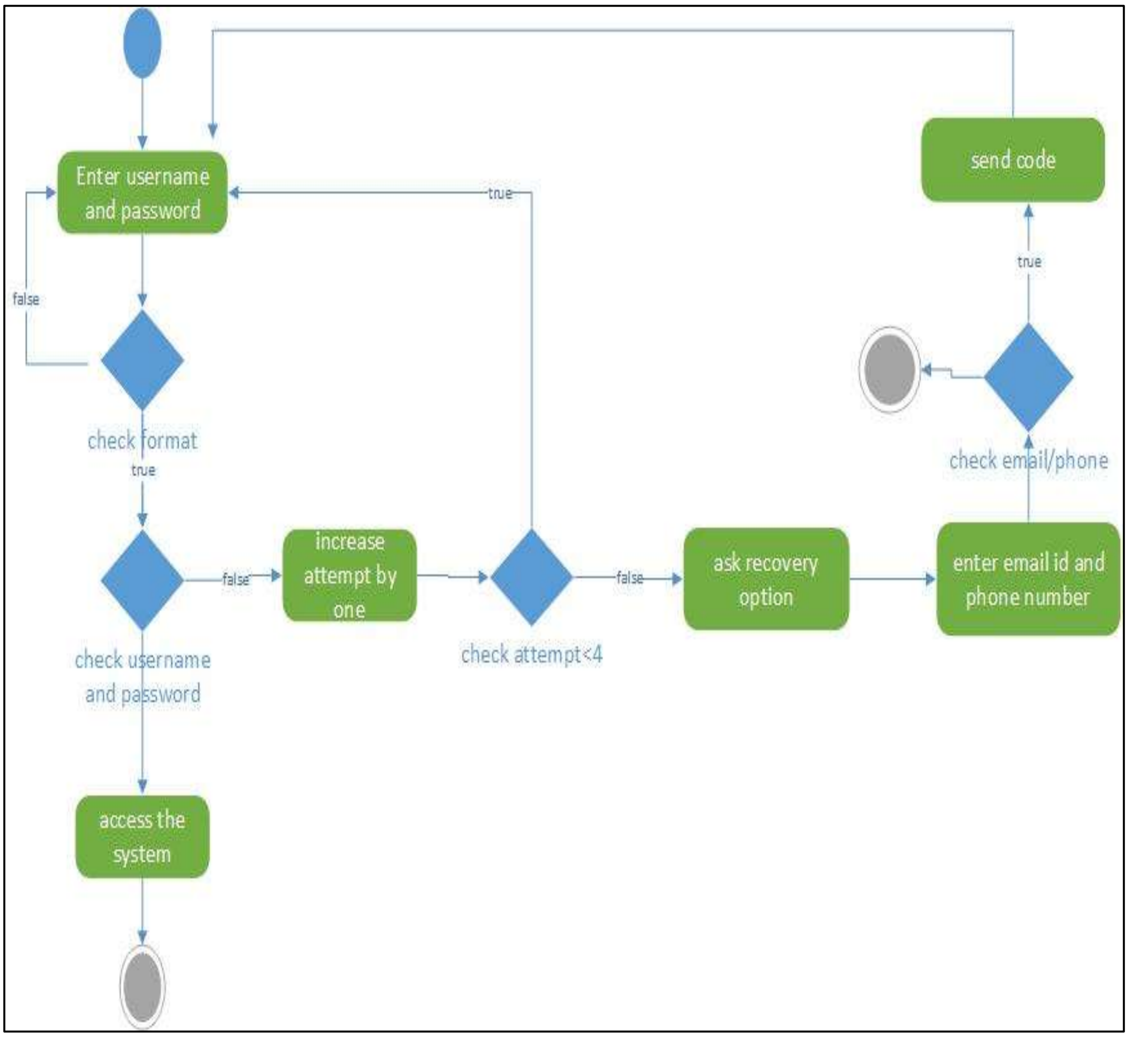


Figure 3.4 Activity Diagram for Login

CLASS DIAGRAM

Class diagram in the UML is a type of static structure diagram that describes the structure of the system by showing the system's classes. This diagram shows their attributes, operations and the relationships between them. Class diagram for "Agrarian Department" as follows.

The class diagram shown as following Figure 3.5 Class Diagram

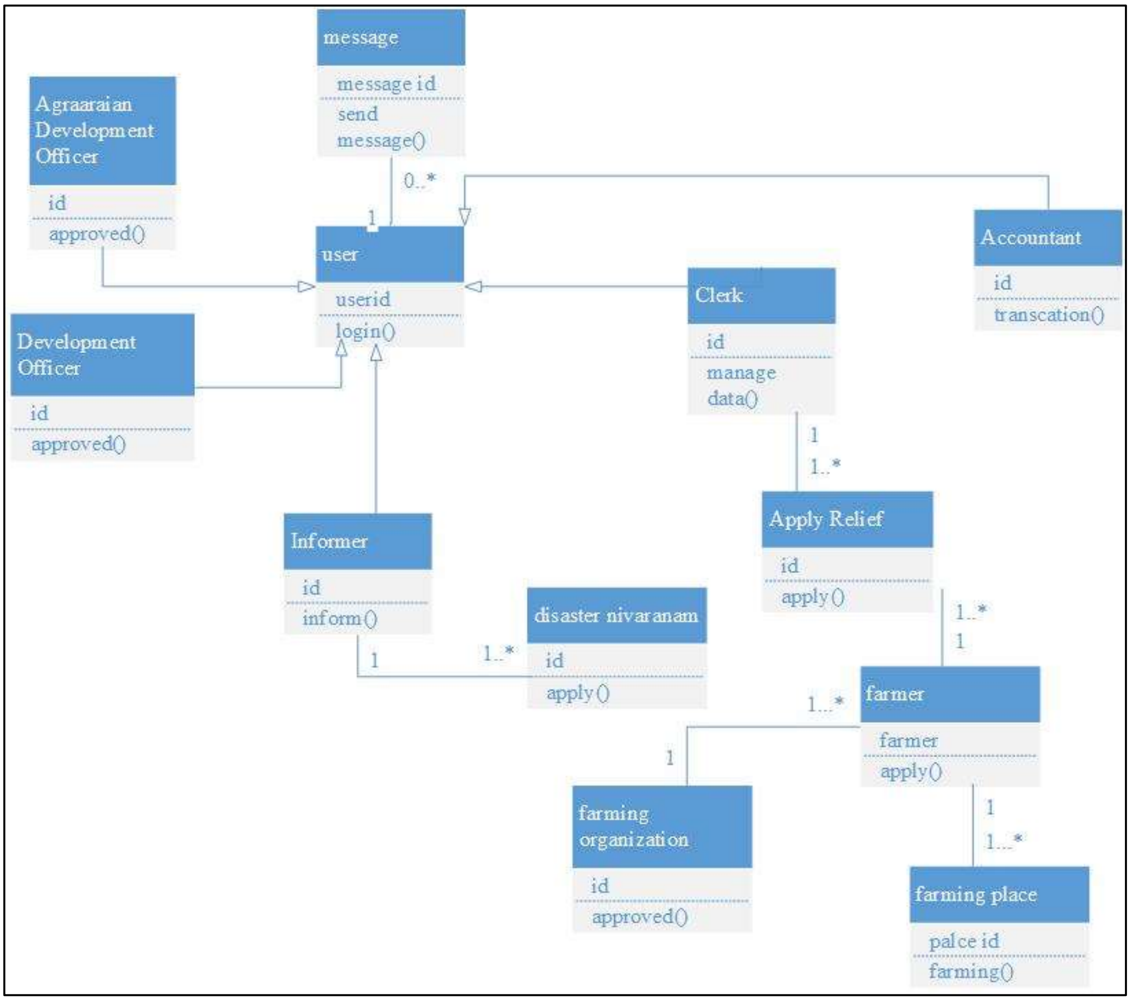


Figure 3.5 Class Diagram

SEQUENCE DIAGRAM

A sequence diagram is an interaction diagram that shows how processes operate with one another and what order. It shows object interaction in a time sequence.

Sequence diagram for login process as shown in Figure 3.6Sequence diagram

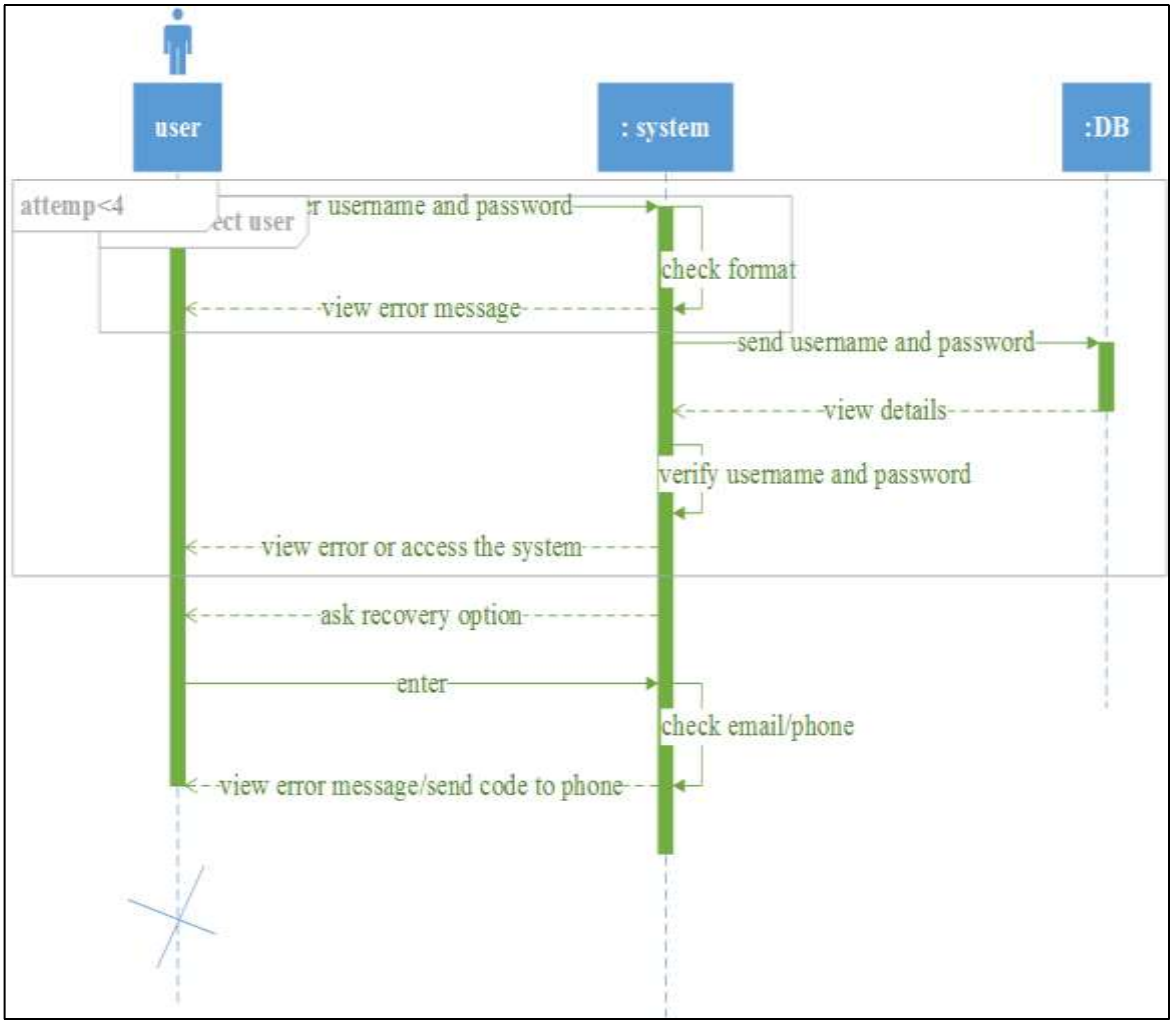


Figure 3.6 Sequence Diagram for Login

3.4 USER INTERFACE DESIGN

User interfaces are the most relevant part of the system development. Designing simple and user-friendly interfaces helps users to use the system easily and work with the system without bothering about the backend processes.

Since the system is mainly used by the staffs of this department. The following interface designing rules were followed while the system was developed.

Design should organize the user interface purposefully, in a meaningful and a useful way.

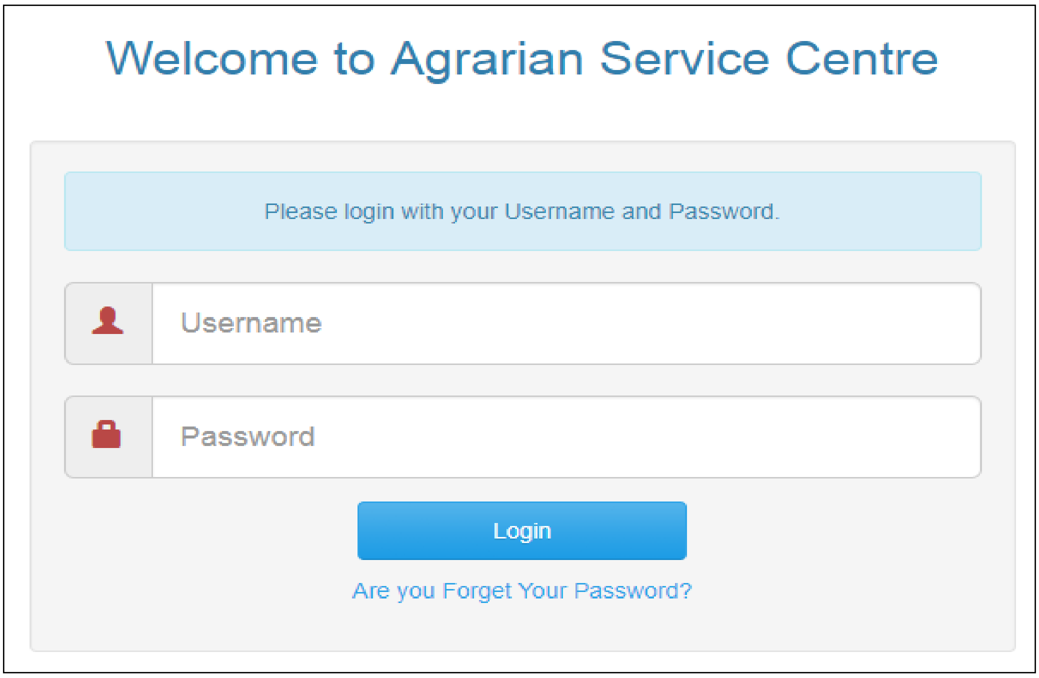
- The design should be simple, common tasks easy, communicating clearly.
- The design should make all needed options and materials for a given task visible without distracting the user with extraneous or redundant information.

- The design should keep users informed of actions or interpretations, changes of state or condition, and errors or exceptions that are.
- The design should be flexible and tolerant, reducing the cost of mistakes and misuse by allowing undoing and redoing, while also preventing errors wherever possible by tolerating varied inputs.
- The design should reuse internal and external components and behaviors.
- Choose a font and the text-size which is clear and readable.
- Be flexible, auto select options as well as let the users select options.
- Use images or icons to make the user to feel comfortable while using the system.
- Display meaningful error messages and proper instructions when the user encounters any errors.

This section is providing few essential interfaces of the Web Based management System for Agrarian Department to show the structure and design of the system. Please refer Appendix C – User Documentation for the rest of the interface designs.

LOGIN PAGE

This page allows users who already registered with this system to login. And this page also provides facility to request for a password if the user forgets it. The Figure 3.7 shown on login page



Welcome to Agrarian Service Centre

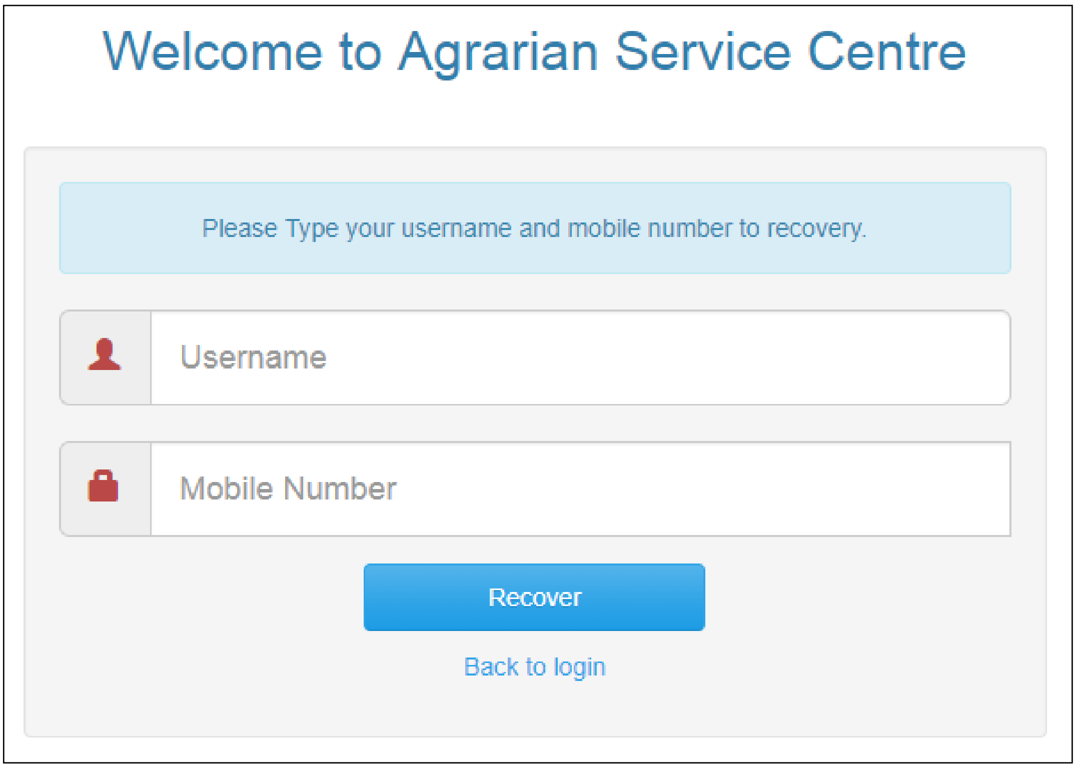
Please login with your Username and Password.

Are you Forget Your Password?

Figure 3.7 Login Page

FORGET PASSWORD

The forget password screen is shown on following Figure 3.8 Forget Password



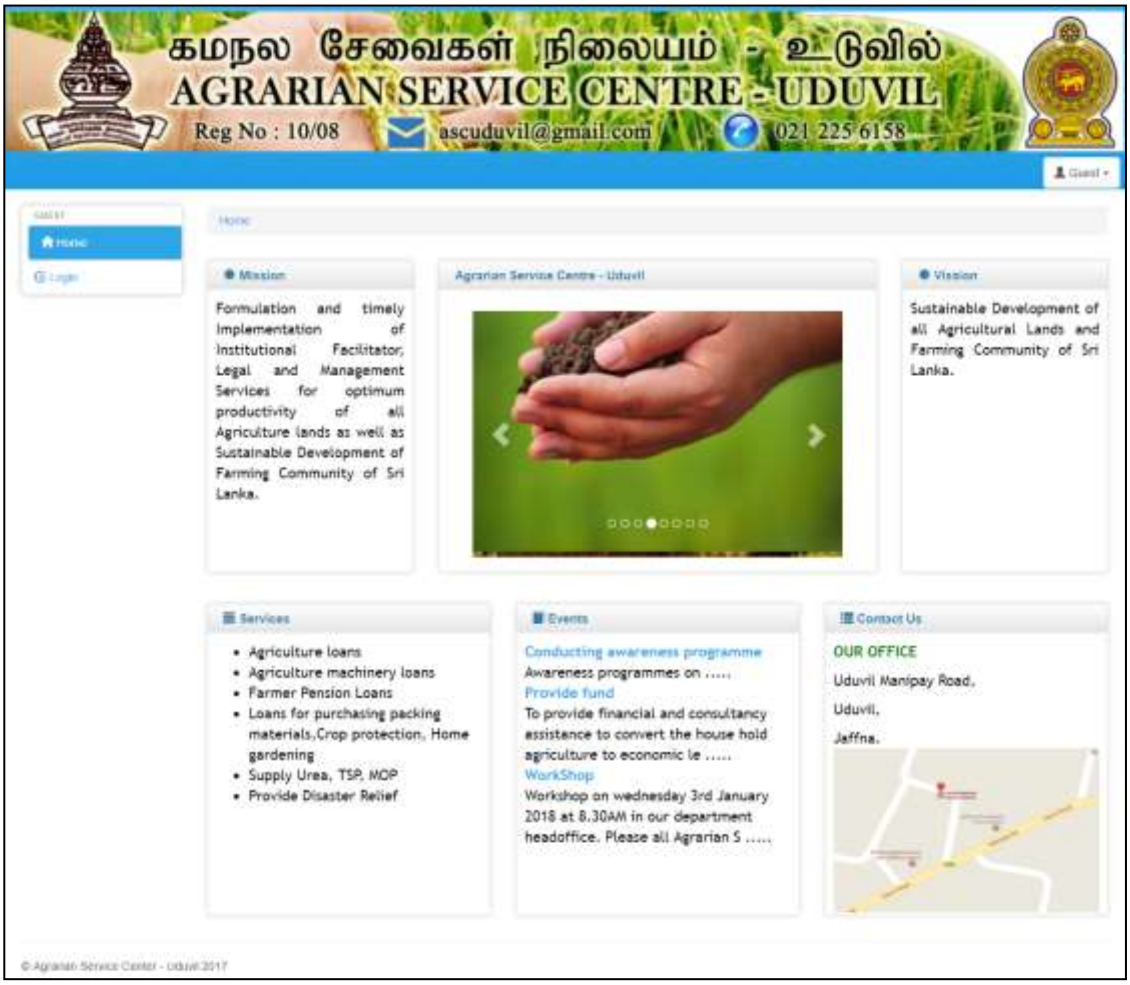
Welcome to Agrarian Service Centre

Please Type your username and mobile number to recovery.

Figure 3.8Forget Password

HOME PAGE

The Figure 3.9 Home page shows the home page of the Web Based management system there are customized links according to the user type logged in order to give different types of privileges to each user type. The main window contains some important data that will be usable for all user categories. User can view details about their Vision, Mission, Services and contact details.



கமநல சேவைகள் நிலையம் - உடுவில்
AGRARIAN SERVICE CENTRE - UDUVIL
Reg No : 10/08 ascuduvil@gmail.com 021 225 6158

Mission

Formulation and timely Implementation of Institutional Facilitator, Legal and Management Services for optimum productivity of all Agriculture lands as well as Sustainable Development of Farming Community of Sri Lanka.

Agrarian Service Centre - Uduvil

Vision

Sustainable Development of all Agricultural Lands and Farming Community of Sri Lanka.

Services

- Agriculture loans
- Agriculture machinery loans
- Farmer Pension Loans
- Loans for purchasing packing materials,Crop protection, Home gardening
- Supply Urea, TSP, MOP
- Provide Disaster Relief

Events

Conducting awareness programme
Awareness programmes on .....
Provide fund
To provide financial and consultancy assistance to convert the house hold agriculture to economic le .....
WorkShop
Workshop on wednesday 3rd January 2018 at 8.30AM in our department headoffice. Please all Agrarian S .....

Contact Us

OUR OFFICE

Uduvil Manipay Road,

Uduvil,

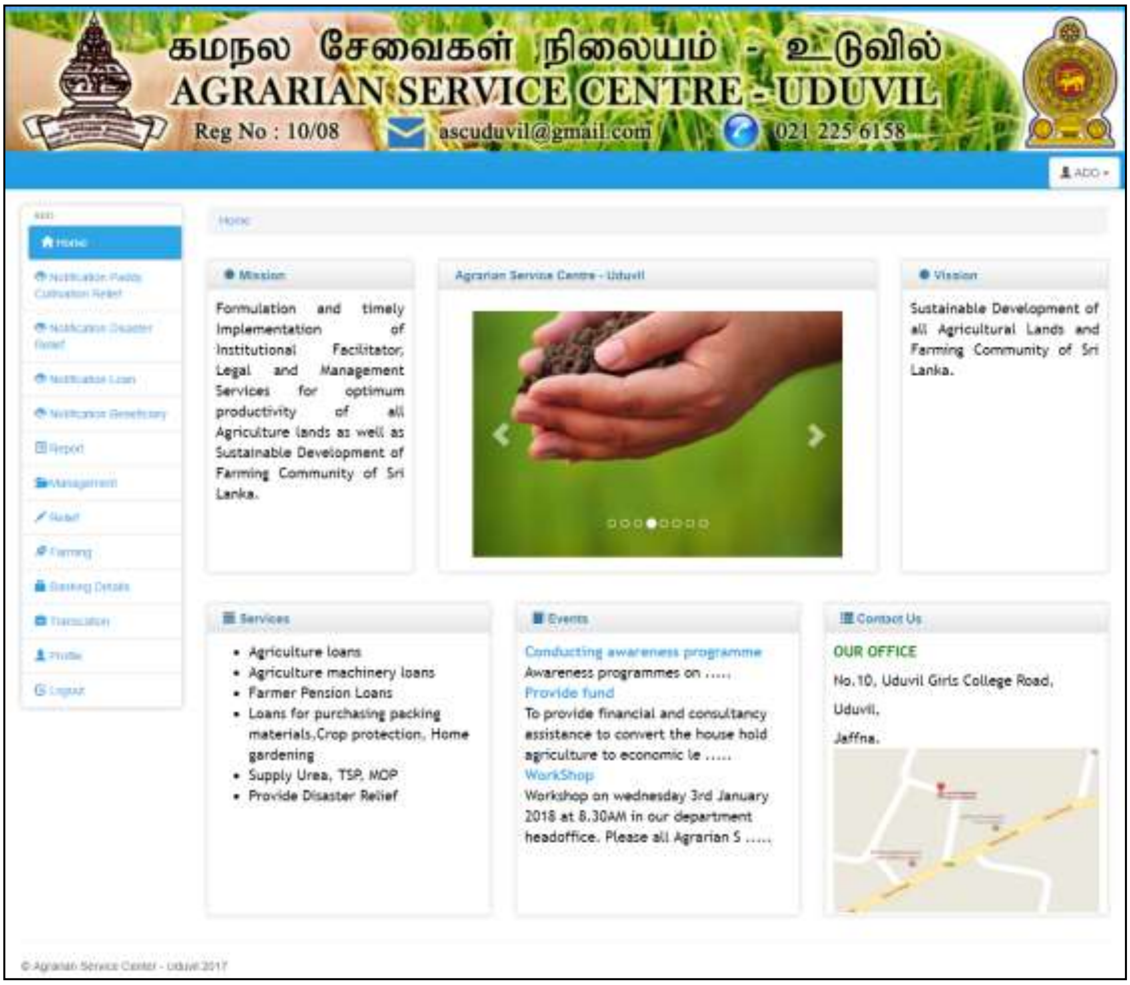
Jaffna.

© Agrarian Service Center - Uduvil 2017

Figure 3.9Homepage

ADMINISTRATION INTERFACE

After user login as admin the Administration interface show the ADO menu page and common body page. The administration interface is shown on the following Figure 3.10 Admin Page



Mission

Formulation and timely Implementation of Institutional Facilitator, Legal and Management Services for optimum productivity of all Agriculture lands as well as Sustainable Development of Farming Community of Sri Lanka.

Agrarian Service Centre - Uduvil

Vission

Sustainable Development of all Agricultural Lands and Farming Community of Sri Lanka.

Services

- Agriculture loans
- Agriculture machinery loans
- Farmer Pension Loans
- Loans for purchasing packing materials,Crop protection, Home gardening
- Supply Urea, TSP, MOP
- Provide Disaster Relief

Events

Conducting awareness programme
Awareness programmes on .....
Provide fund
To provide financial and consultancy assistance to convert the house hold agriculture to economic le .....
WorkShop
Workshop on wednesday 3rd January 2018 at 8.30AM in our department headoffice. Please all Agrarian S .....

Contact Us

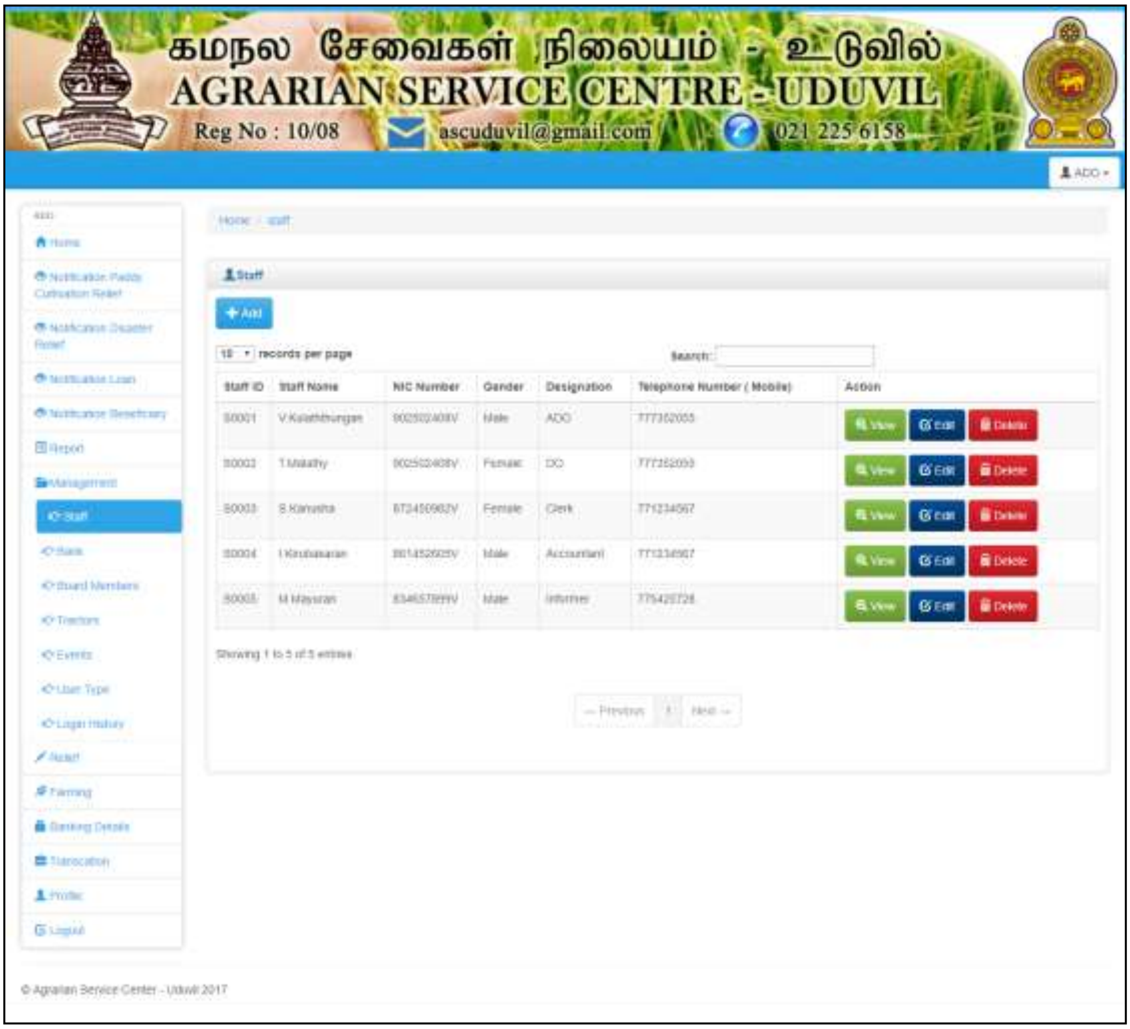
OUR OFFICE
No.10, Uduvil Girls College Road,
Uduvil,
Jaffna.

© Agrarian Service Center - Uduvil 2017

Figure 3.10 Admin page

STAFF INFORMATION

This page displays the staff basic details. And this page also contains action buttons such as view, edit, delete, Add new staff button and print staff. The staff information screen is shown on following Figure 3.11 Staff Details



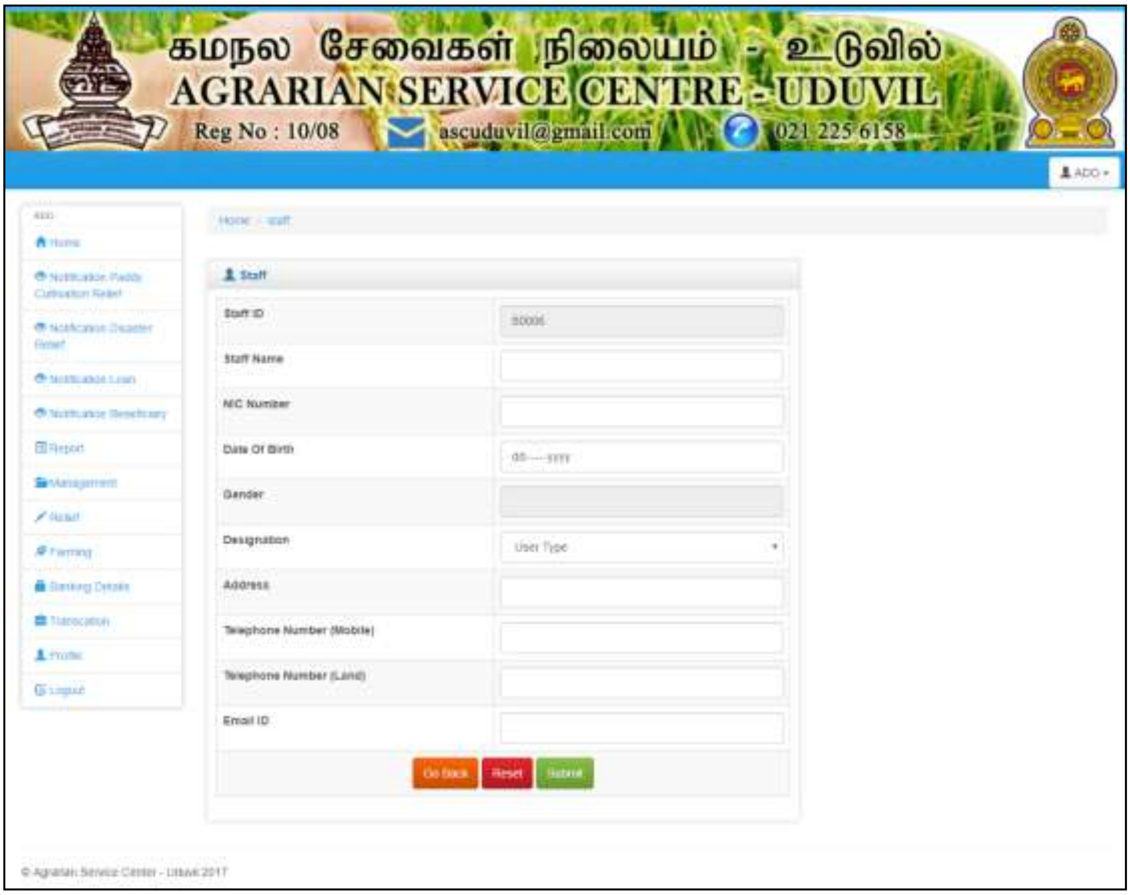
Staff ID	Staff Name	NIC Number	Gender	Designation	Telephone Number (Mobile)	Action
S0001	V.Kalaththungan	902502408V	Male	ADO	777352055	View Edit Delete
S0002	T.Malathy	902502408V	Female	DO	777352053	View Edit Delete
S0003	S.Karusha	872450982V	Female	Clerk	771234567	View Edit Delete
S0004	I.Kirubaharan	801452605V	Male	Accountant	771234567	View Edit Delete
S0005	M.Mayuran	834657899V	Male	Informer	775420728	View Edit Delete

Showing 1 to 5 of 5 entries

Figure 3.11Staff Details

ADD NEW STAFF FORM

System user can easily add a new staff details throw add new staff button. If user clicks this button there will be a form appear. The Figure 3.12 Staff Form shows this staff information entering interface.



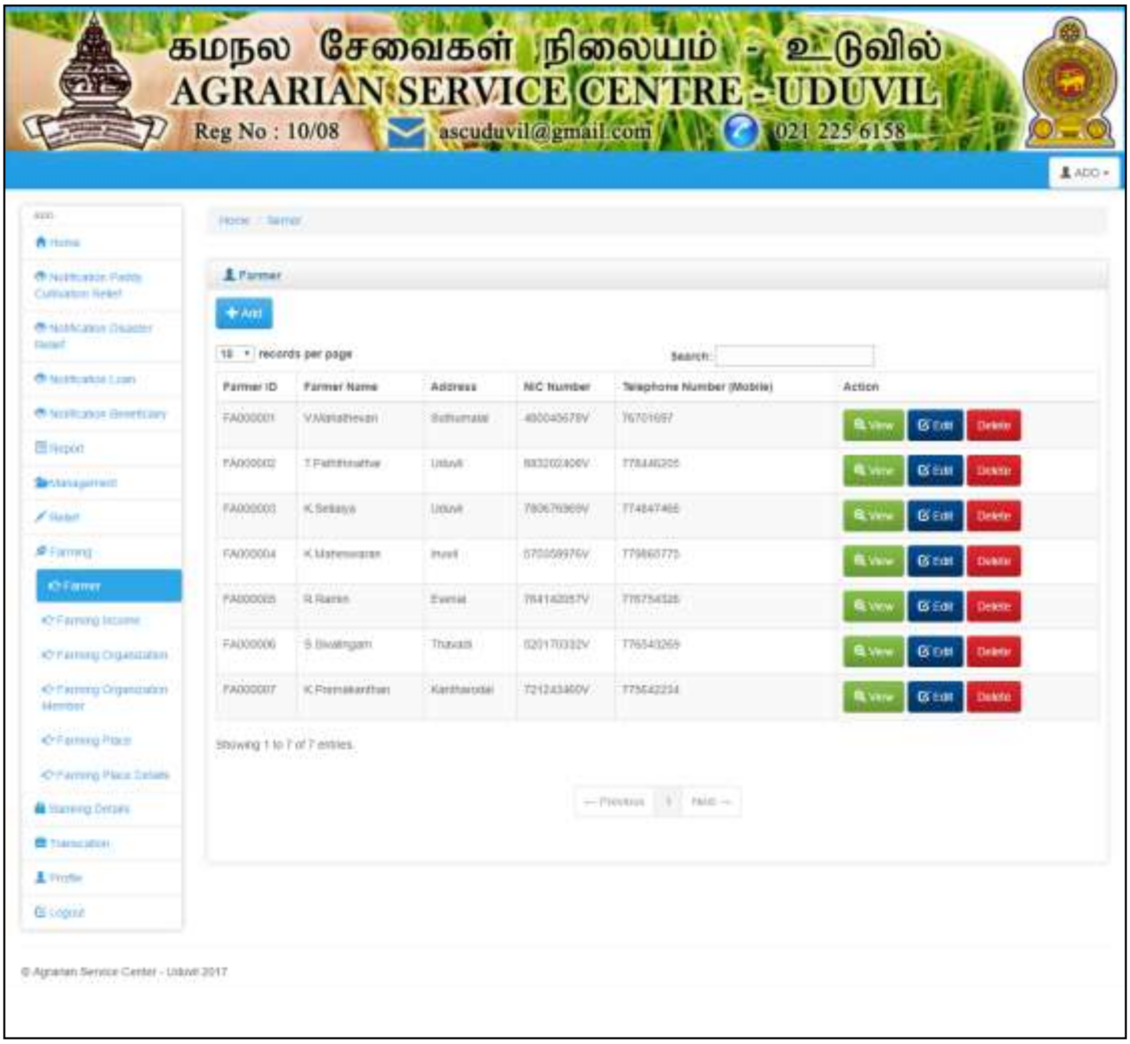
Staff

Staff ID	80006
Staff Name	
NIC Number	
Date Of Birth	dd-----yyyy
Gender	
Designation	User Type
Address	
Telephone Number (Mobile)	
Telephone Number (Land)	
Email ID	

Figure 3.12 Add New Staff Details

FARMER INFORMATION

This page displays the farmer basic details. And this page also contains action buttons such as view, edit, delete, Add new farmer button these buttons are visible only staff. The farmer information screen is shown on following Figure 3.13FarmerDetails.



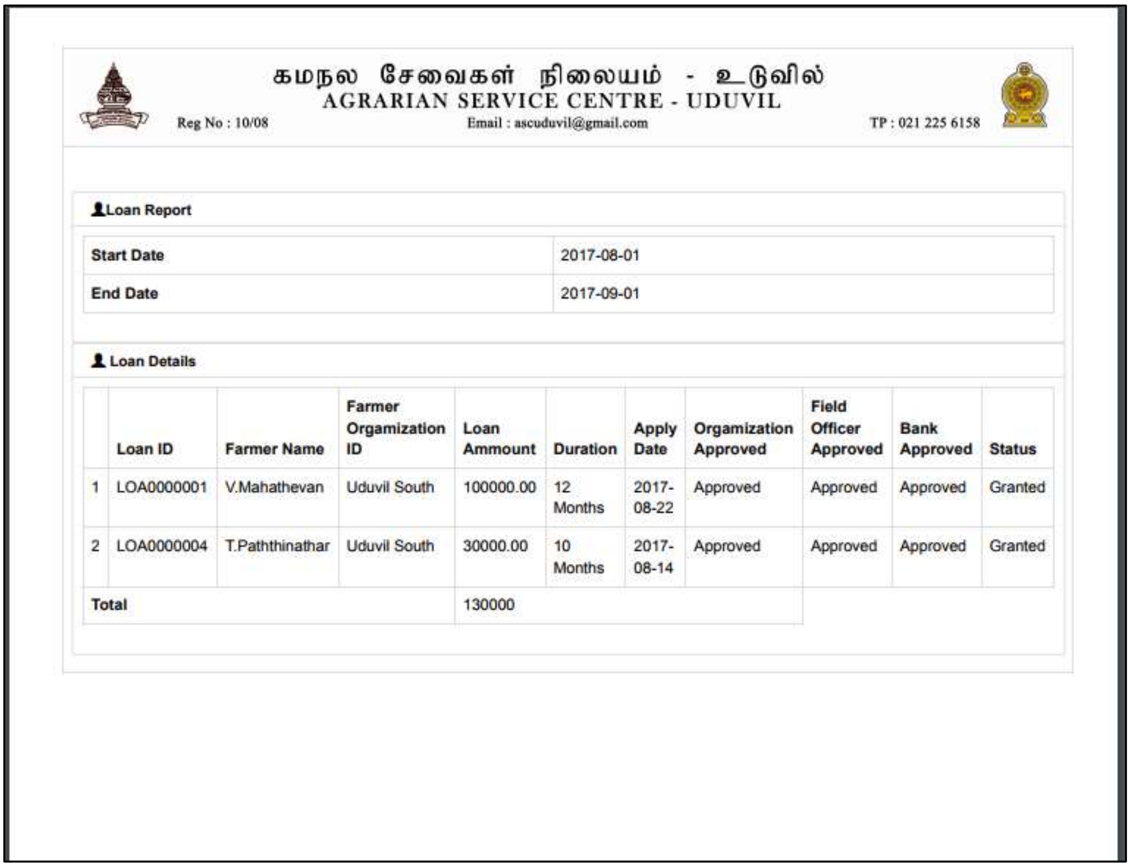
Farmer ID	Farmer Name	Address	NIC Number	Telephone Number (Mobile)	Action
FA000001	V.Mahathevan	Suthumalai	490040578V	76701697	View Edit Delete
FA000002	T.Paththinathan	Uduvil	893202406V	778446205	View Edit Delete
FA000003	K.Sellaiya	Uduvil	780676969V	774847466	View Edit Delete
FA000004	K.Maheswaran	Inuvil	570058976V	779860775	View Edit Delete
FA000005	R.Ramin	Evenai	784142057V	776754326	View Edit Delete
FA000006	S.Sivalingam	Thavadi	820170332V	776540265	View Edit Delete
FA000007	K.Premakanthan	Kantharodai	721243400V	775642234	View Edit Delete

Showing 1 to 7 of 7 entries

Figure 3.13 Farmer Details

REPORT GENERATOR

This system allows to generating report according to various conditions given by the user. Users are also able to print the generated report for their documentation. Figure 3.14 Report is an example of report of loan information of given date generated by the administrator.



கமநல சேவைகள் நிலையம் - உடுவில்
AGRARIAN SERVICE CENTRE - UDUVIL
Reg No : 10/08 Email : ascuduvil@gmail.com TP : 021 225 6158

Loan Report

Start Date	2017-08-01
End Date	2017-09-01

Loan Details

	Loan ID	Farmer Name	Farmer Orgamization ID	Loan Ammount	Duration	Apply Date	Orgamization Approved	Field Officer Approved	Bank Approved	Status
1	LOA0000001	V.Mahathevan	Uduvil South	100000.00	12 Months	2017-08-22	Approved	Approved	Approved	Granted
2	LOA0000004	T.Paththinathar	Uduvil South	30000.00	10 Months	2017-08-14	Approved	Approved	Approved	Granted
Total				130000						

Figure 3.14 Loan Report

CHAPTER 4: IMPLEMENTATION

4.1 INTRODUCTION

An implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through computer programming and deployment. Many implementations may exist for a given specification or standard.

PHP and MySQL is chosen in the development and coding. Some tools are also used to develop the system effectively. Dreamweaver, Notepad ++, Bootstraps are some examples of tools used in development. All coding were written in an understandable format along with comments. This will help to produce the system with more functionality in the future [5].

4.2 THE DEVELOPMENT ENVIRONMENT

Resource requirements for develop the project are list down below.

4.2.1 HARDWARE REQUIREMENTS

- Pentium 4 computers
- Basic Printer for report printing

4.2.2 SOFTWARE REQUIREMENTS

FOR IMPLEMENTATION PURPOSE:

- Operating system
- Browsers

FOR DEVELOPMENT PURPOSE:

- Windows operating system
- WampServer Version 2.4 for Windows
- Apache Web Server Version 2.4.4
- PHP Script Language Version 5.4.16
- MySQL Database Version 5.6.12
- PhpMyAdmin Database Manager Version 4.0.4

- MySQL Workbench 6.0.8 CE
- Microsoft Visio 2013
- Adobe Dreamweaver Version 13.0
- Adobe Photoshop Version 14.0
- Internet Explore/Google Chrome/Mozilla Firefox

4.2.3 SPECIAL REQUIREMENTS

- Internet Facility
- Web hosting and Domain registration

TECHNOLOGIES USED FOR DEVELOPMENT

- PHP was the main development language used to develop the main system and its logics.
- MySQL was used to handle all the development related to the database.
- HTML 5 was used to build the base Interfaces of the system.
- CSS was used to make the plain HTML 5 interfaces more attractive and user friendly, which also decided the look and feel of the system.
- JavaScript was used to code all the client-side validation.
- JQuery which is also based on JavaScript was used to implement the pre-coded time picker module, Date Picker module, and password validate.

4.3 CODE AND MODULES STRUCTURE

In this system, code and modules divided into User management, ADO management, DO management, Clerk management and Accountant management. User management considers about login, change password, forget password. ADO management considers about Relief Approval.

DO management considers about manage staff, manage farming details and manage loan approval. Clerk management considers mange farmer and mange staff details. Accountant management considers maintains account.The Module structure screen is shown on following Figure 4.1

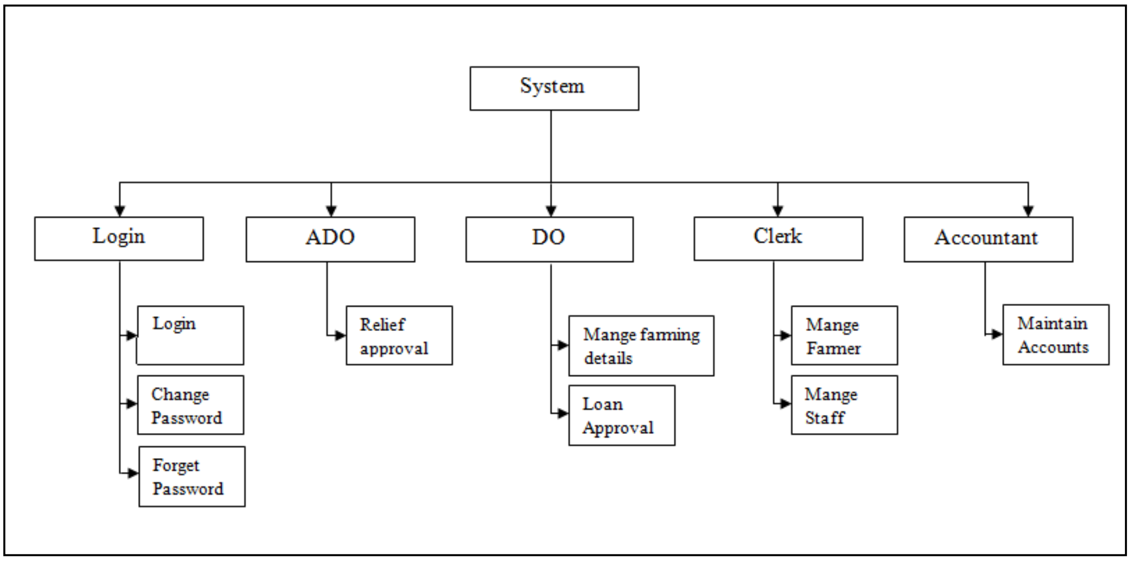


Figure 4.1 Module Structure

4.4 REUSED MODULES AND COMPONENTS

In this system some CSS, JavaScript of the charisma [6] template is reused to only for the design part of table, form, button and user interface. This template is available in web in free of charge and anyone can download it.

4.5 NETWORK IMPLEMENTATION

Since the system is web based and available to public users. It could be installed on a web server and a dedicated database server is used to manage the system's database. The above Figure 4.2 shown a Dedicated Web server and a Database Server is installed within the organization but public servers also can be used to implement the system. It will reduce much initial cost and maintenance host.

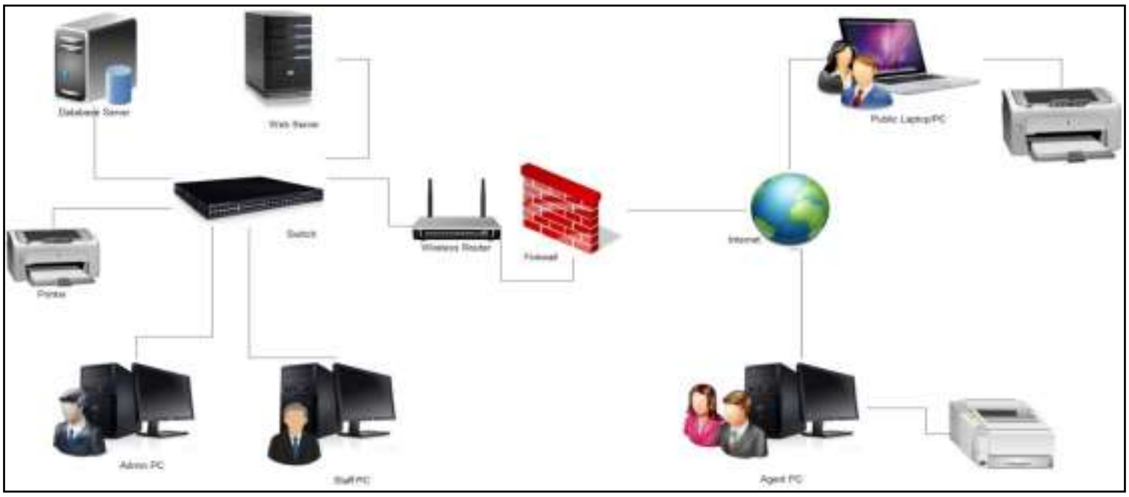


Figure 4.2 Network Implementation

4.6 CODES OF THE MAIN MODULES

As the system is developed using PHP, MySQL for the programming and HTML, CSS, JavaScript are used to designing and validation purpose.

mysql_connect: this used to connect the server. Here three parameters such as hostname, server name and password will pass to connect.

mysql_select_db: this used to select the database from server. Here one parameter database name will pass to connect database.

mysql_query: this used to execute the query such as insert or update or delete or select.

mysql_num_rows: this used to count the how many rows will collect when query is executed.

mysql_real_esace_string: this used to provide a basic secure when insert data into database.

4.6.1 DATABASE CONNECTIVITY

When develop the system first we want to connect the system with database. Used the mysql_connect code to connect the local host. After successful of the connection of local host we want to connect the database. We used the mysql_select_db code for connect the database

DATABASE CONNECTION CODE

```
<?php
$con=mysql_connect("localhost","root","");
if(!$con)
{
die("sever error");
}
$db=mysql_select_db("agra");
if(!$db)
{die("database error");
}
?>
```

4.6.2 LOGIN TO SYSTEM

This login has to used authenticate the users. When we enter username and password correctly system displays their user interface. If we enter username or password wrong the system will provide a message box with meaningful message. If we enter wrong username or password more than three then systems automatically go to forget password page.

LOGIN CODE

```
<?php
if(isset($_POST["btnsubmit"]))
{
$entererusername=$_POST["txtusername"];
$enterpwd=$_POST["txtpassword"];
$sqlusername="SELECT * FROM login WHERE user_name='$entererusername'";
$resultusername=mysql_query($sqlusername)ordie("sql error in sqlusername ".mysql_error());
if(mysql_num_rows($resultusername)>0)
{
//username is correct
$rowusername=mysql_fetch_assoc($resultusername);

$sqlpassword="SELECT * FROM login WHERE user_name='$entererusername' AND password='$enterpwd'";
$resultpassword=mysql_query($sqlpassword)ordie("sql error in sqlpassword ".mysql_error());
if(mysql_num_rows($resultpassword)>0)
{
//username and password is correct
$sqlusertype="SELECT * FROM user_type WHERE user_type_id='$rowusername[user_type_id]'";
$resultusertype=mysql_query($sqlusertype)ordie("sql error in sqlusertype ".mysql_error());
$rowusertype=mysql_fetch_assoc($resultusertype);
$_SESSION["username"]=$entererusername;
$_SESSION["usertype"]=$rowusertype["user_type"];
//set attempt to zero

$sqlupdate="UPDATE login SET attempt='0' WHERE user_name='$entereruname'";

$resultupdate=mysql_query($sqlupdate)ordie("sql error in sqlupdate ".mysql_error());

header("location:index.php");

}

elseif($rowusername["attempt"]<3)

{

//attempt less than three

$sqlupdate="UPDATE login SET attempt=attempt+1 WHERE user_name='$entereruname'";

$resultupdate=mysql_query($sqlupdate)ordie("sql error in sqlupdate ".mysql_error());

echo'<script>alert("Your password is wrong");</script>';

}

else

{

//forget password redirect

$_SESSION["forgetusername"]=$entereruname;

echo'<script>alert("You attempt more than three time; please recover your password!");

window.location.href="forget.php";</script>';

}

}

else

{

//username is wrong

echo'<script>alert("There is no such username!");</script>';

}

}

?>
```

4.6.3 AUTO NUMBER-INCREMENT

This coding used to increment item no automatically when adding new items to database. This type of coding is also used to messages to increment Message ID automatically when adding to database

```
<?php
$sqlstaffid="SELECT staff_id FROM staff ORDER BY staff_id DESC LIMIT 1";
$resultstaffid=mysql_query($sqlstaffid)ordie("sql error in sqlstaffid ".mysql_error());
if(mysql_num_rows($resultstaffid)>0)
{
//second or more
$rowstaffid=mysql_fetch_assoc($resultstaffid);
$staffid=++$rowstaffid["staff_id"];
}
else
{
//first time
$staffid="S0001";
}
?>
```

4.6.4 NIC VERIFICATION CODE

This code used to validate NIC number. User has entering data with mistakes wrong data there will be messages according their mistakes.

This coding used to increment item no automatically when adding new items to database. This type of coding is also used to messages to increment Message ID automatically when adding to database.

```
functionnicnumber()//nic validation start
{
varnic=document.getElementById("txtnic").value;
if(nic.length==10)//for nic length 10
```

```
{
var gender=nic.substring(2,5);
var nicformat1=/^[0-9]{9}[a-zA-Z0-9]{1}$/;
if(nic.match(nicformat1))// validate first 9 character are number
{
var nicformat2=/^[0-9]{9}[vVxX]{1}$/;
if(nic.match(nicformat2))//validate first 9 character are number and last v/x
{
if(gender>500)
{
document.getElementById("txtgender").value="Female";
}
else
{
document.getElementById("txtgender").value="Male";
}
calculatedob(nic);//if correct goto this function
}
else
{
alert("last character must be V/v/X/x");
document.getElementById("txtnic").value="";
document.getElementById("txtnic").focus();
document.getElementById("txtdob").value="";
}
}
else
{
alert("First 9 characters must be numbers");
document.getElementById("txtnic").value="";
document.getElementById("txtnic").focus();
document.getElementById("txtdob").value="";
}
}
elseif(nic.length==12)//for nic length 12
{
var gender=nic.substring(4,7);
var nicformat3=/^[0-9]{12}$/;
if(nic.match(nicformat3))
{
if(gender>500)
{
document.getElementById("txtgender").value="Female";
}
else
{
document.getElementById("txtgender").value="Male";
}
calculatedob(nic);//if correct goto this function
}
else
{
alert("All 12 characters must be number");
document.getElementById("txtnic").value="";
document.getElementById("txtnic").focus();
document.getElementById("txtdob").value="";
}
}
elseif(nic.length==0)//nic field blank
{}
else//nic is not 10 or 12
{
```

```
alert("NIC No must be 10 or 12 Characters");

document.getElementById("txtnic").value="";

document.getElementById("txtnic").focus();

document.getElementById("txtdob").value="";

}

}
```

4.6.5 INSERT CODE

```
if(isset($_POST["btnsubmitnew"]))

{

$sqlinsert=" INSERT INTO
staff(staff_id,staff_name,nic,dob,gender,designation,address,mobile_number,land_number,em
ail)

VALUES('".mysql_real_escape_string($_POST["txtstaffid"])."',

        '".mysql_real_escape_string($_POST["txtstaffname"])."',

        '".mysql_real_escape_string($_POST["txtnic"])."',

        '".mysql_real_escape_string($_POST["txtdob"])."',

        '".mysql_real_escape_string($_POST["txtgender"])."',

        '".mysql_real_escape_string($_POST["txtdesignation"])."',

        '".mysql_real_escape_string($_POST["txtaddress"])."',

        '".mysql_real_escape_string($_POST["txtmobile"])."',

        '".mysql_real_escape_string($_POST["txtland"])."',

        '".mysql_real_escape_string($_POST["txtemail"])."')";

$resultinsert=mysql_query($sqlinsert)ORdie("sql error in sqlinsert".mysql_error());


$sqlinsert=" INSERT INTO login(user_name,password,user_type_id,attempt,code,status)

VALUES('".mysql_real_escape_string($_POST["txtstaffid"])."',

        '".mysql_real_escape_string($_POST["txtnic"])."',

        '".mysql_real_escape_string($_POST["txtdesignation"])."',

        '".mysql_real_escape_string(0)."',

        '".mysql_real_escape_string(0)."',

        '".mysql_real_escape_string("Active")."')";
$resultinsert=mysql_query($sqlinsert)ORdie("sql error in sqlinsert".mysql_error());

if($resultinsert)

{

echo'<script>alert("successfully insert");</script>';

}

}
```

4.6.6 GENERATE REPORT CODE

```
functionload_loan_report()

{

varstartdate=document.getElementById("txtstartdate").value;

varenddate=document.getElementById("txtenddate").value;



varxmlhttp=newXMLHttpRequest();

xmlhttp.onreadystatechange=function()

{

if(xmlhttp.readyState==4&&xmlhttp.status==200)

{

document.getElementById("load_ajax_loan").innerHTML=xmlhttp.responseText;

document.getElementById("btnprint_loan_report").disabled=false;

}

};

xmlhttp.open("GET","report_ajax.php?option=loan_report&startdate="+startdate+"&enddate=
"+enddate,true);

xmlhttp.send();

}



functionprint_load_loan_report()//print the loan report

{

varstartdate=document.getElementById("txtstartdate").value;

varenddate=document.getElementById("txtenddate").value;
varurl="print.php?pr=report.php&option=loan_report&startdate="+startdate+"&enddate="+enddate;

window.open(url,"_blank");

}
```

4.7 SECURITY

Since the system is accessed via web by public and carryout financial transactions it is important to implement good security mechanisms. The administration have full privilege to access the system, in other hand other users have less privilege than administration. From our system only registered user only access the system, the registered user uses their user ID and password to login and access the system. After register in our system the new user can access the system

If user try to login with wrong password, our system allow only three time after three time system automatically load the forget password page. From forget password web page the system verify the user ID and registered hand phone number then system send a password to user’s registered hand phone number.

CHAPTER 5: EVALUATION

5.1 INTRODUCTION

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Furthermore, this testing can provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Testing techniques are included, but not limited to the process of executing a program or application with the intent of finding software bugs (errors or other defects) [7].

5.2 TESTING PROCEDURE

Testing procedures are the testing process and techniques used to ensure that the system is tested and validated before handing over to the client. Testing is done to detect failures so that defects may be discovered and corrected.

Formal testing techniques were used to find out the error on this system. System testing was done in different testing levels. Unit testing, Integration testing, Component interface testing, System testing and Acceptance testing.

5.2.1 UNIT TESTING

Unit testing, which is known as component testing, refers to tests that verify the functionality of a specific section of code, usually at the function level. In an object-oriented environment, this is usually at the class level, and the minimal unit tests consist of constructors and destructors.

5.2.2 INTEGRATION TESTING

Integration testing is any type of software testing that seeks to verify the interfaces between components against a software design. Software components may be integrated in an iterative way or all together ("Big Bang"). Normally the former is considered a better practice since it allows interface issues to be located more quickly and fixed.

5.2.3 COMPONENT INTERFACE TESTING

The practice of component interface testing can be used to check the handling of data passed between various units, or subsystem components, beyond full integration testing between those units.

5.2.4 SYSTEM TESTING

System testing, or end-to-end testing, tests a completely integrated system to verify that it meets its requirements. For example, a system test might involve testing a logon interface, then creating and editing an entry, plus sending or printing results, followed by summary processing or deletion (or archiving) of entries, then logoff.

5.2.5 ACCEPTANCE TESTING

At last the system is delivered to the user for Acceptance testing.

5.3 TESTING PLANS AND TESTING CASES

A test specification is called test plan. Testing specifications were designed before the implementation phrase to ensure all testing were done before the implementation on client side.

A test case normally consists of a unique identifier, requirement references from a design specification, preconditions, events, a series of steps to follow, input, output, expected result.

Our test modules were designed as follows,

- ADO module (Table 5.1)
- DO module (Table 5.2)
- Accountant module (Table 5.3)
- Clerk module (Table 5.4)
- Common function module (Table 5.5)

5.3.1 ADO MODULE

The Agrarian Development Officer (ADO) has full privilege to modify the data. ADO can manage staff, account, and loan and generate the reports in various categories.

Test No	**Test Description**	**Step to Test**	**Expected Result**	**Status**
01	Manage staff	Enter parameters for the field	should be able to manage customer	pass
02	Add new Account	Enter parameters for the field	Able to add new account details	pass
03	Manage loan	Enter parameters for the field	should be able to manage loan	pass
04	Generate Staff details	Select staff in the report menu click report	Administrator should be able to generate staff details report	pass
05	Generate farmer details	Select farmer in the report menu click report	Administrator should be able to generate farmer details report	pass
06	Generate loan details	Select loan in the report menu click report	Administrator should be able to generate loan details report	pass

Table 5.1 ADO Module

5.3.2 DO MODULE

The Development Officer (DO) can manage the staff and loan details.

Test No	**Test Description**	**Step to Test**	**Expected Result**	**Status**
01	Manage staff	Enter parameters for the field	should be able to manage customer	pass
02	Manage loan	Enter parameters for the field	should be able to manage loan	pass

Table 5.2 DO Module

5.3.3 ACCOUNTANT MODULE

Accountant can manage loan and transaction details and has authorized to approve the loan, disaster relief and transaction details.

Test No	**Test Description**	**Step to Test**	**Expected Result**	**Status**
01	Manage loan	Enter parameters for the field	should be able to manage loan	pass
02	Make Transaction	Enter parameters for the field	should be able to make transaction or visible message	pass

Table 5.3 Accountant Module

5.3.4 CLERK MODULE

Clerk can modify the farmer details and do the data entry of the details.

Test No	**Test Description**	**Step to Test**	**Expected Result**	**Status**
01	Manage farmer details	Enter parameters for the field	should be able to manage farmer	pass
02	Create Account	Enter parameters for field	Can create an Account	pass

Table 5.4 Clerk Module

5.3.5 COMMON FUNCTION MODULE

All the users can able to login and access the system. In this module login, forget password, database connection and form validation are included.

Test No	**Test Description**	**Step to Test**	**Expected Result**	**Status**
01	Login (Positive)	Enter a valid registered username and password	Redirect to appropriate page based on user type	Pass
02	Login	Enter wrong	Display error	Pass

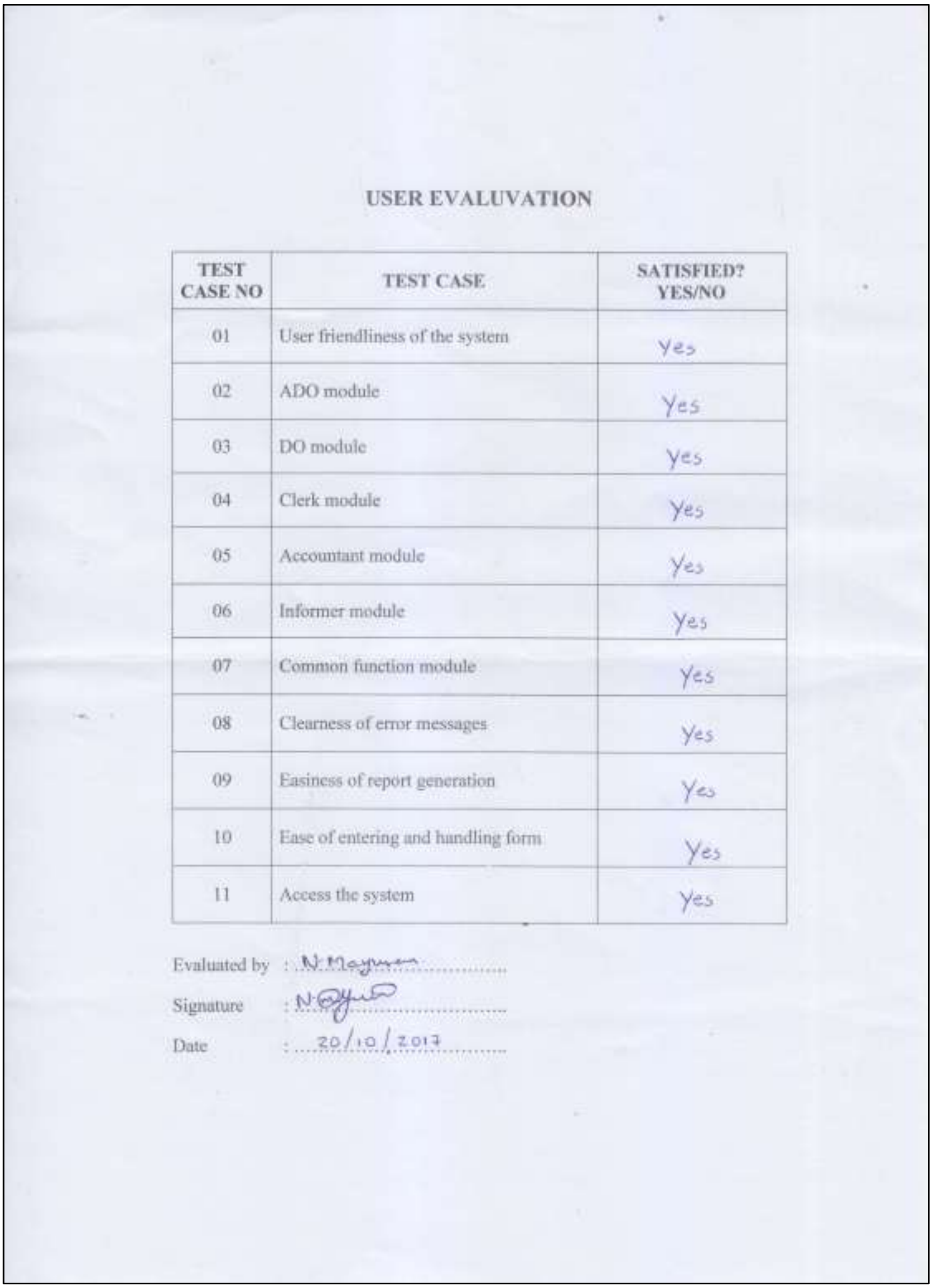
	(Negative)	username and password	message as follows (Error E-mail ID or Password)	
03	Forgot Password	Click forgot password on the login page	Ask username and phone number for verification if provided correctly ask answers for the security question if provided correctly send password to registered email	Pass
04	Database connection	Want to connect	The system connect with database successfully	pass
05	Form validation	Enter wrong parameters for the field	Display error message	pass
06	Required field	Do not enter data required field	Display error message and couldn’t move next field	pass

Table 5.5 Common Module

5.4 TEST DATA

The sample data were entered to the system to all modules tested in order to test the function. We test the all form, view, manage and report with sample data.

Acceptance Result (Figure 5.1)



USER EVALUVATION

TEST CASE NO	TEST CASE	SATISFIED? YES/NO
01	User friendliness of the system	Yes
02	ADO module	Yes
03	DO module	yes
04	Clerk module	Yes
05	Accountant module	Yes
06	Informer module	Yes
07	Common function module	Yes
08	Clearness of error messages	yes
09	Easiness of report generation	Yes
10	Ease of entering and handling form	Yes
11	Access the system	Yes

Evaluated by : N. Mayuran

Signature : N.

Date : 20/10/2017

Figure 5.1 Acceptance Result

The sample data were entered to the system to test the all modules. All the form, error message, success message, user privilege and etc. were tested. The user evaluation done from various user and client accept this system and acceptance result shown in the following Figure 5.2 is for user evaluation.

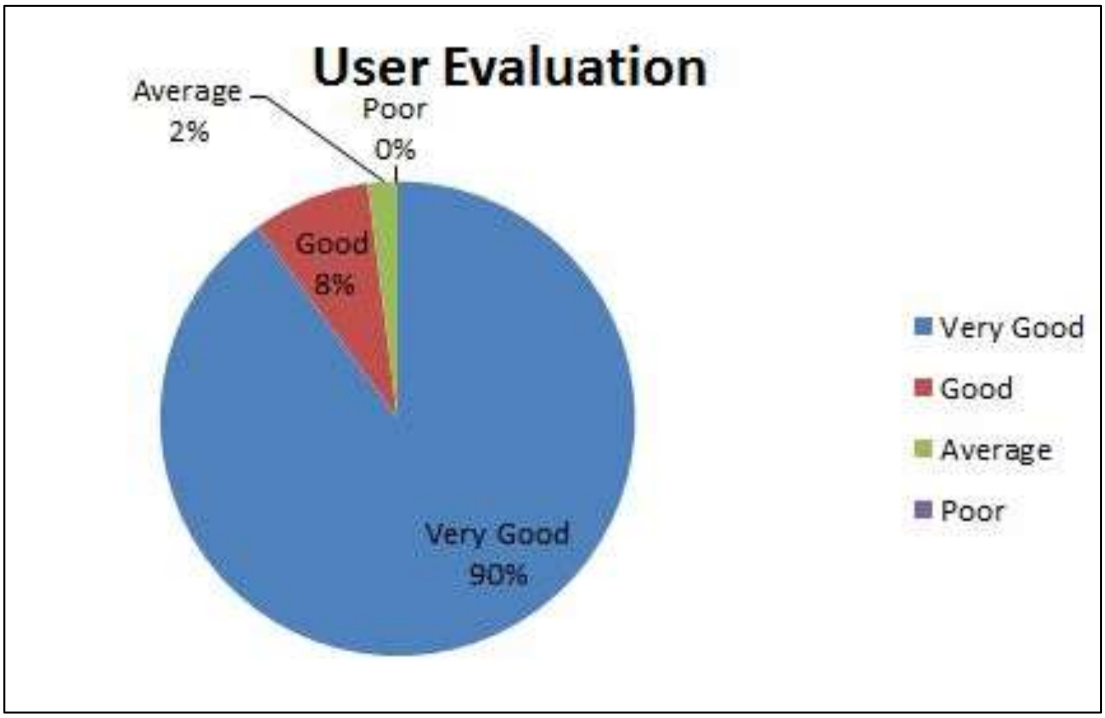


Figure 5.2 User Evaluation

CHAPTER 6: CONCLUSION

In this system, web based management system was give good knowledge to us and we go to next level of our career successfully. This project gave knowledge and how to implement our knowledge what gain throughout from previous semester of BIT (Bachelor of Information Technology) degree.

6.1 LESSON LEARNT

The knowledge gained throughout this project is really valuable and giving me an opportunity to work on a whole system development life cycle from the initial phase to implementation.

This project gave me a chance to practice and implement the theories which I have learnt throughout the BIT degree program. It also helps me to find out and get good knowledge in the latest development technologies such as HLML5, PHP, JavaScript, Ajax and MySQL. In the designing I got clear knowledge about reused modules like twitter bootstrap. Furthermore developing this project helped me to improve my technical and communication skills.

6.2 PROJECT ASSESSMENT

This system was built using the technologies Object Oriented (OO) concept, HTML, CSS, PHP, Ajax and JavaScript with back end MySQL.

The system was successfully satisfied with user requirements including their functional and non-functional requirements and client was accepting our project with full fill of satisfaction. From this system the management of the Agrarian Service Centre facilitates to manage farmer, staff of the department, Easy to manage branch, get up to date reports to help make better management decision and support to management.

Manage farmer and farming place details, farming organization, organization member details, Every GiramaNilathari has a farming organization and it has a community group, this organization provides a service to farmer. Manage paddy cultivation relief details, Manage farmer bank account and transaction details, Keep records of banking details and manage daily routine transaction details, Manage disaster relief , loan details, Supply money, agriculture inputs and agriculture instruments based on their disaster evaluation or insurance or loan to farmer. Calculating loan and crediting term interest for farmer, Calculate the interest for loan and send alert to farmer about loan details, Quickly and fully efficient Reporting.

Generate various categories in daily or monthly or yearly report, Communicate within system through message via SMS or E-Mail, Send alert SMS to farmer regarding loan, interest, relief details. By this system maintain the time management.

6.3 FUTUTRE WORK

Our system satisfy client requirement but further more if we change some function out system make more quality and our country now a day’s grow up in technology field so we want to do some future work.

These are some future work:

- Improve the security by using encryption methods.
- Allow to farmers login
- Implement online payment methods
- Generate more reports in graphical view

REFERENCE

[1] Northern education management system,[Online]

Available: http://www.edudept.np.gov.lk/[Accessed: 08.07.2017]

[2] Charity Navigator, [Online]

Available: http://www.charitynavigator.org/ [Accessed: 08.07.2017]

[3] Department of Agriculture, [Online]

Available: https://www.doa.gov.lk/index.php/en/ [Accessed: 08.07.2017]

[4] Prototyping model,[Online]

Available:https://www.tutorialspoint.com/sdlc/sdlc_software_prototyping.htm

[Accessed: 10.07.2017]

[5] Implementation, [Online]

Available:https://en.wikipedia.org/wiki/Implementation#Computer_science

[Accessed: 20.09.2017]

[6] Charisma – Free Template, [Online]

Available:https://usman.it/themes/charisma/ [Accessed: 20.07.2017]

[7] Software testing, [Online]

Available:
https://www.princeton.edu/~achaney/tmve/wiki100k/docs/Software_testing.html [Accessed: 03.10.2017]

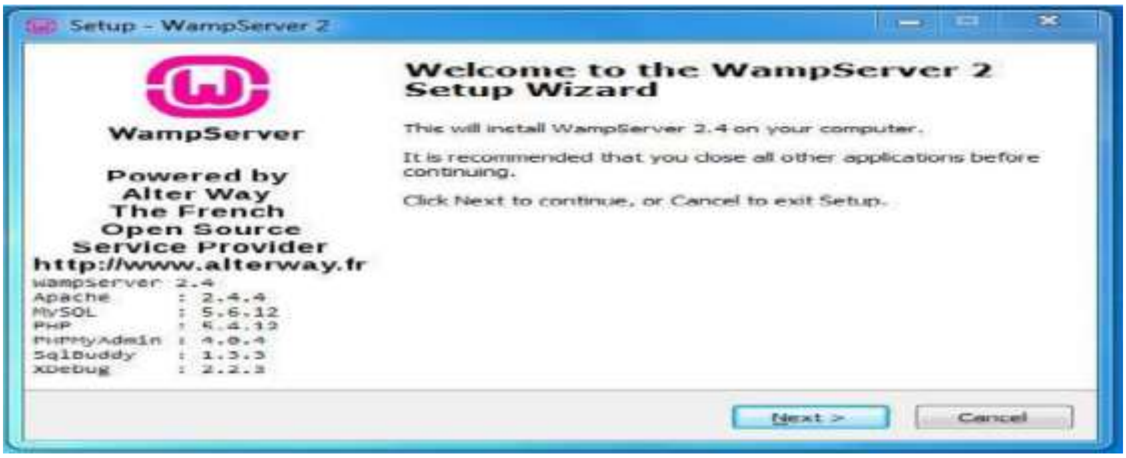
APPENDIX

APPENDIX A – SYSTEM DOCUMENTATION

INTRODUCTION

We give instruction how to install our system and database, and it is help to client to installation and maintenance.

Step 1: Double click on download file WampServer 2.4 and just follow the instruction. Everything is automatic. Figure A.1 Setup wizard



Welcome to the WampServer 2 Setup Wizard

This will install WampServer 2.4 on your computer.

It is recommended that you close all other applications before continuing.

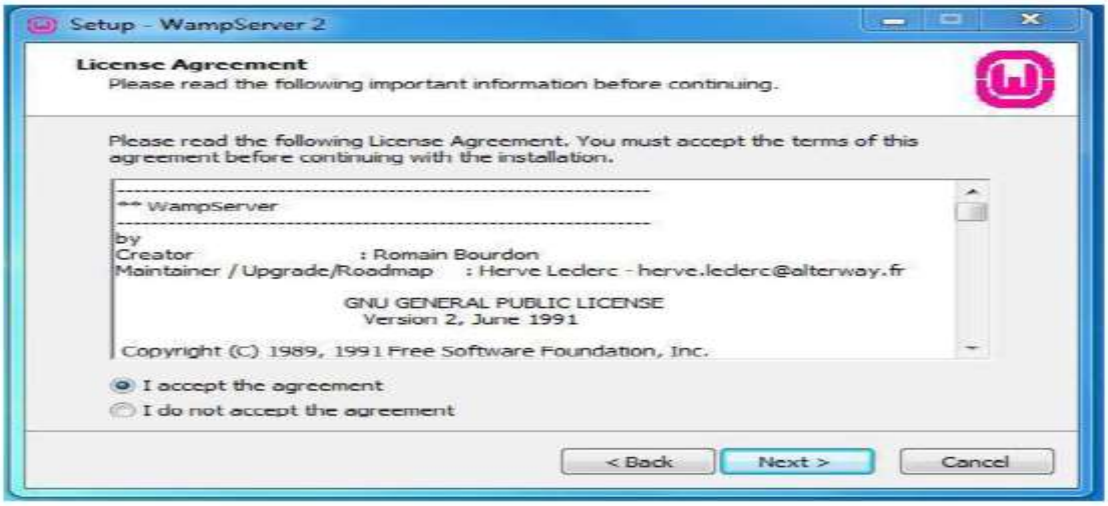
Click Next to continue, or Cancel to exit Setup.

Powered by
Alter Way
The French
Open Source
Service Provider
http://www.alterway.fr

```
wampserver 2.4
Apache       : 2.4.4
MySQL        : 5.6.12
PHP          : 5.4.12
PhpMyAdmin   : 4.0.4
SqlBuddy     : 1.3.3
XDebug       : 2.2.3
```

Figure A.1 Setup Wizard

Step 2: After click next, accept the agreement and click next. Figure A.2 Agreement



License Agreement

Please read the following important information before continuing.

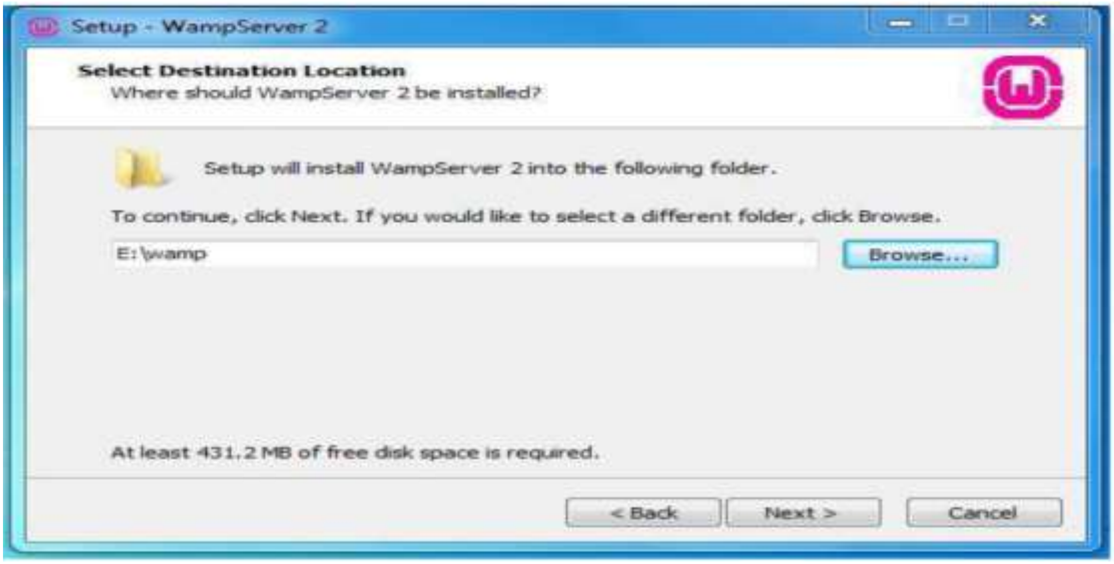
Please read the following License Agreement. You must accept the terms of this agreement before continuing with the installation.

```
-----------------------------------------------------------
** WampServer
-----------------------------------------------------------
by
Creator                     : Romain Bourdon
Maintainer / Upgrade/Roadmap    : Herve Leclerc - herve.leclerc@alterway.fr

                GNU GENERAL PUBLIC LICENSE
                   Version 2, June 1991

 Copyright (C) 1989, 1991 Free Software Foundation, Inc.
```

● I accept the agreement
○ I do not accept the agreement

Figure A.2 Agreement

Step 3: Select the installation location of WampServer. Figure A.3 Folder Location



Select Destination Location

Where should WampServer 2 be installed?

Setup will install WampServer 2 into the following folder.

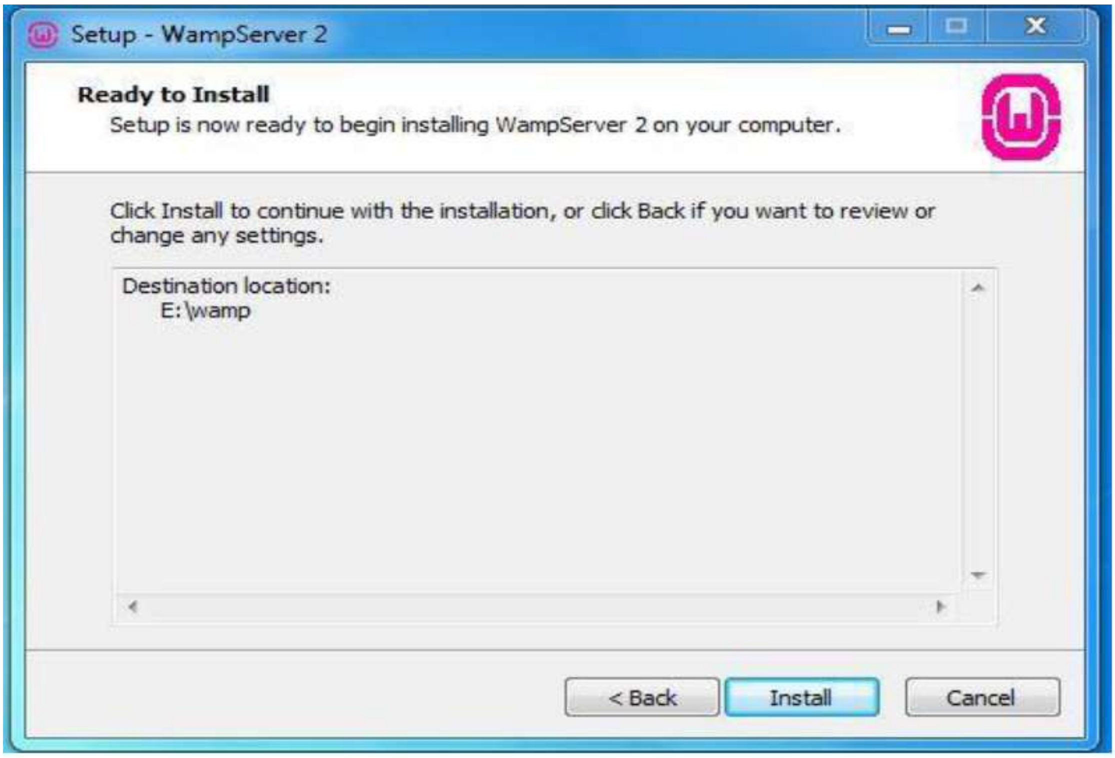
To continue, click Next. If you would like to select a different folder, click Browse.

E:\wamp

At least 431.2 MB of free disk space is required.

Figure A.3 Folder Location

Step 4: Click Install and It finish automatically. Figure A.4 Install



Ready to Install

Setup is now ready to begin installing WampServer 2 on your computer.

Click Install to continue with the installation, or click Back if you want to review or change any settings.

Destination location:
E:\wamp

Figure A.4 Install

Step 5: After Install, start the WampSever in taskbar WampServer visible in green color. After green color, go and type in browser's address bar "local host or 127.0.0.1".

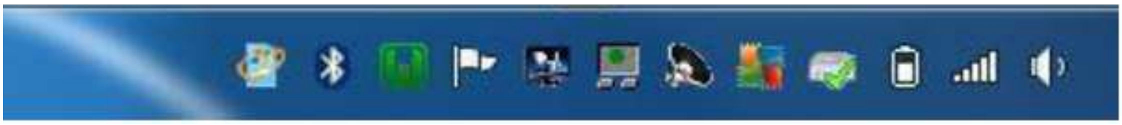
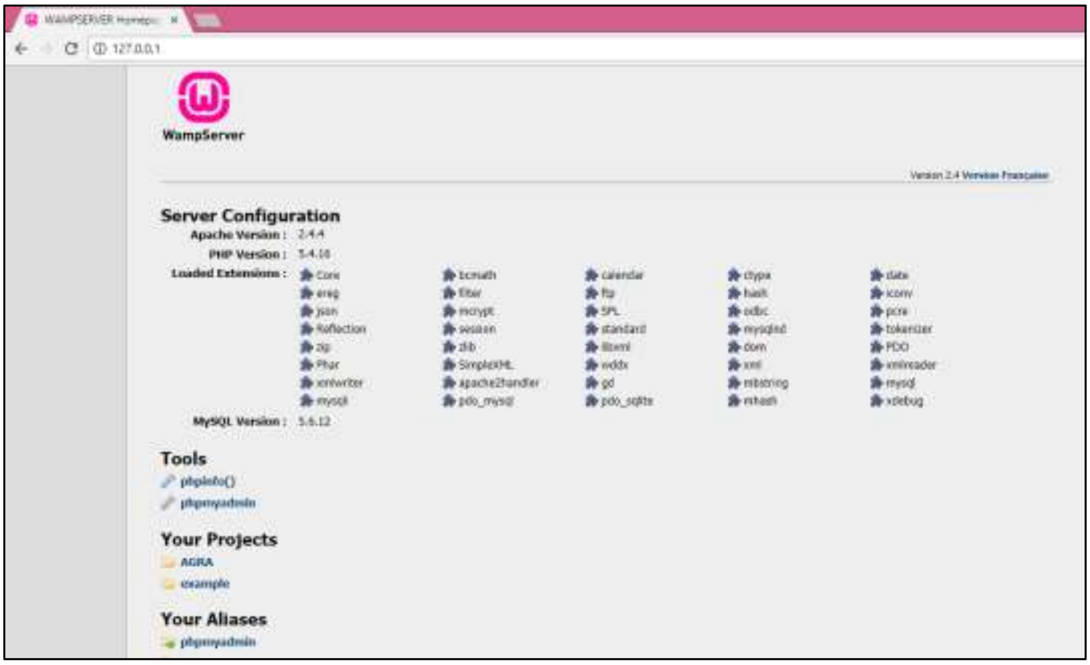


Figure A.5 Task Bar

Step 6: In browser, there is phpmyadmin under Tools heading, click that link and type username as "root" and password is blank. And click go button



Server Configuration

Apache Version : 2.4.4

PHP Version : 5.4.16

Loaded Extensions :

Core	bcmath	calendar	ctype	date
ereg	filter	ftp	hash	iconv
json	mcrypt	SPL	odbc	pcre
Reflection	session	standard	mysqlnd	tokenizer
zip	zlib	libxml	dom	PDO
Phar	SimpleXML	wddx	xml	xmlreader
xmlwriter	apache2handler	gd	mbstring	mysql
mysqli	pdo_mysql	pdo_sqlite	mhash	xdebug

MySQL Version : 5.5.12

Tools

phpinfo()

phpmyadmin

Your Projects

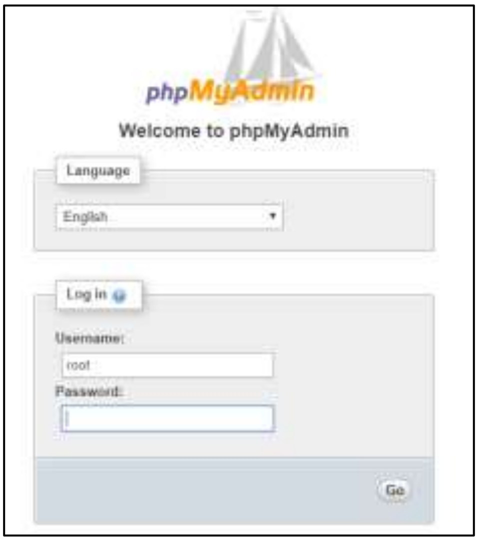
AGRA

example

Your Aliases

phpmyadmin

Figure A.6 Index page of WampServer



Welcome to phpMyAdmin

Language

English

Log in

Username: root

Password:

Figure A.7 Welcome page

Step 7: After login click Import tab, in that tab click browse and select the database file agra.sql file from CD and click go button.

Step 8: You now successfully upload database

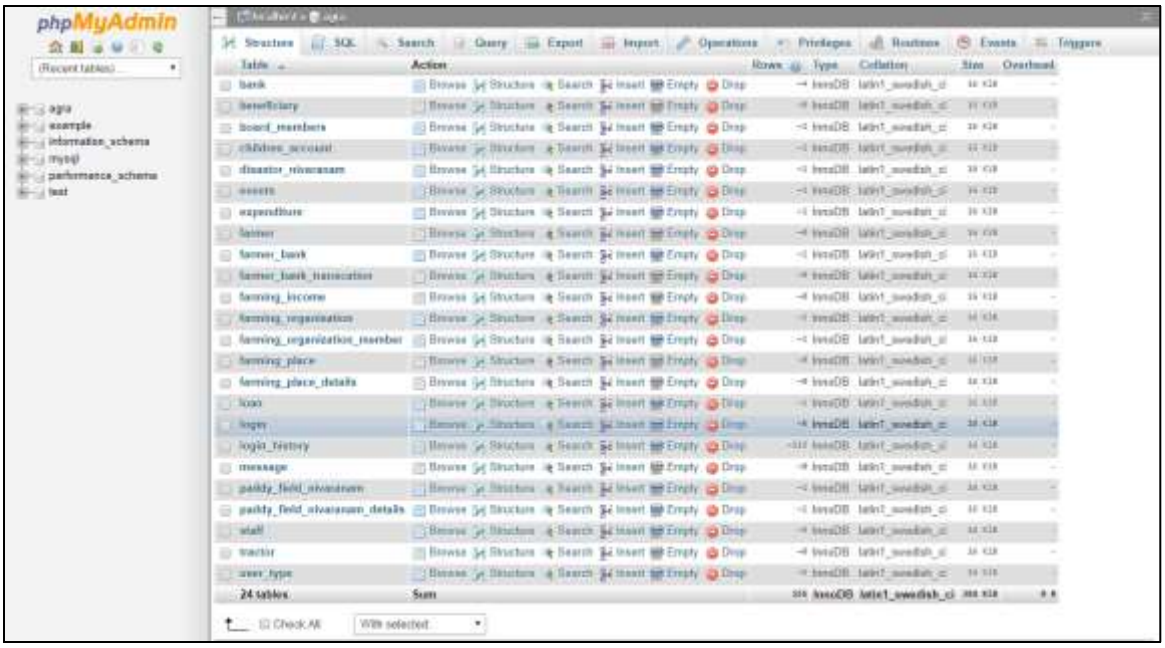


Table	Action	Rows	Type	Collation	Size	Overhead
bank	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
beneficiary	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
board_members	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
children_account	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
disaster_nivaranam	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
events	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
expenditure	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farmer	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farmer_bank	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farmer_bank_transcation	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farming_income	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farming_organization	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farming_organization_member	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farming_place	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
farming_place_details	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
loan	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
login	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
login_history	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
message	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
paddy_field_nivaranam	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
paddy_field_nivaranam_details	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
staff	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
tractor	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
user_type	Browse Structure Search Insert Empty Drop	[illegible]	InnoDB	latin1_swedish_ci	16 KiB	-
24 tables	Sum	[illegible]	InnoDB	latin1_swedish_ci	[illegible] KiB	0 B

Figure A.8 Database

Step 9: The “www” directory will be automatically created. According the Step 3, our www folder is under F:\wamp. In www folder we create a subdirectory and put our PHP files.

APPENDIX B – DESIGN DOCUMENTATION

The additional diagram such as activity diagram and sequence diagram of this system include in this section.

Activity diagram for Loan Apply process as shown in Figure B.1 Activity diagram

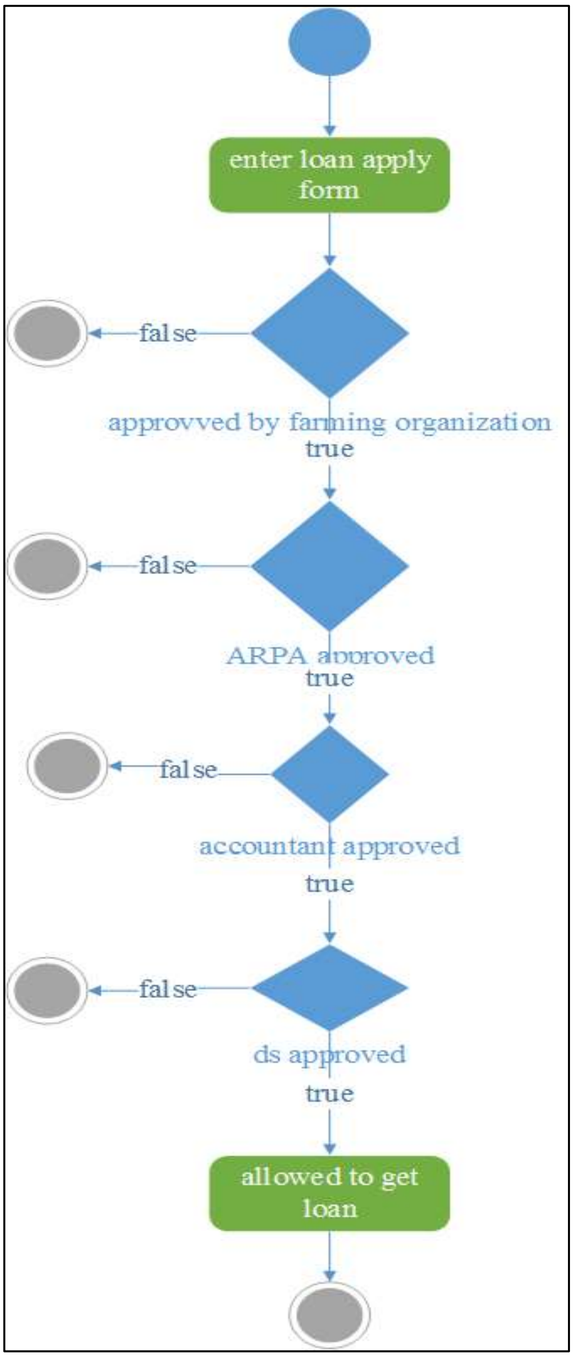


Figure B.1 Activity Diagram for Loan Apply

Activity diagram for Generate report process as shown in Figure B.2 Activity diagram

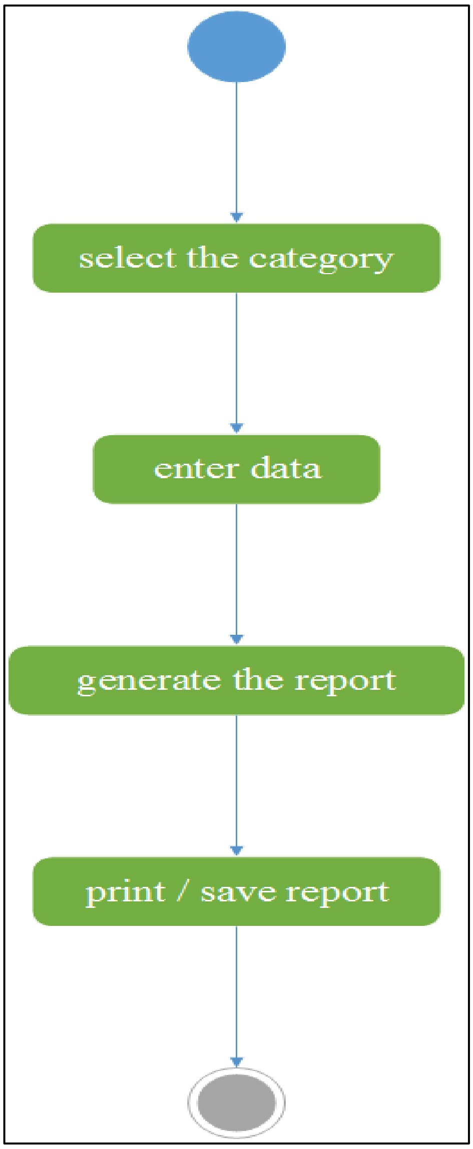


Figure B.2 Activity Diagram for Generate Report

Sequence diagram for loan apply process as shown in Figure B.3Sequence diagram

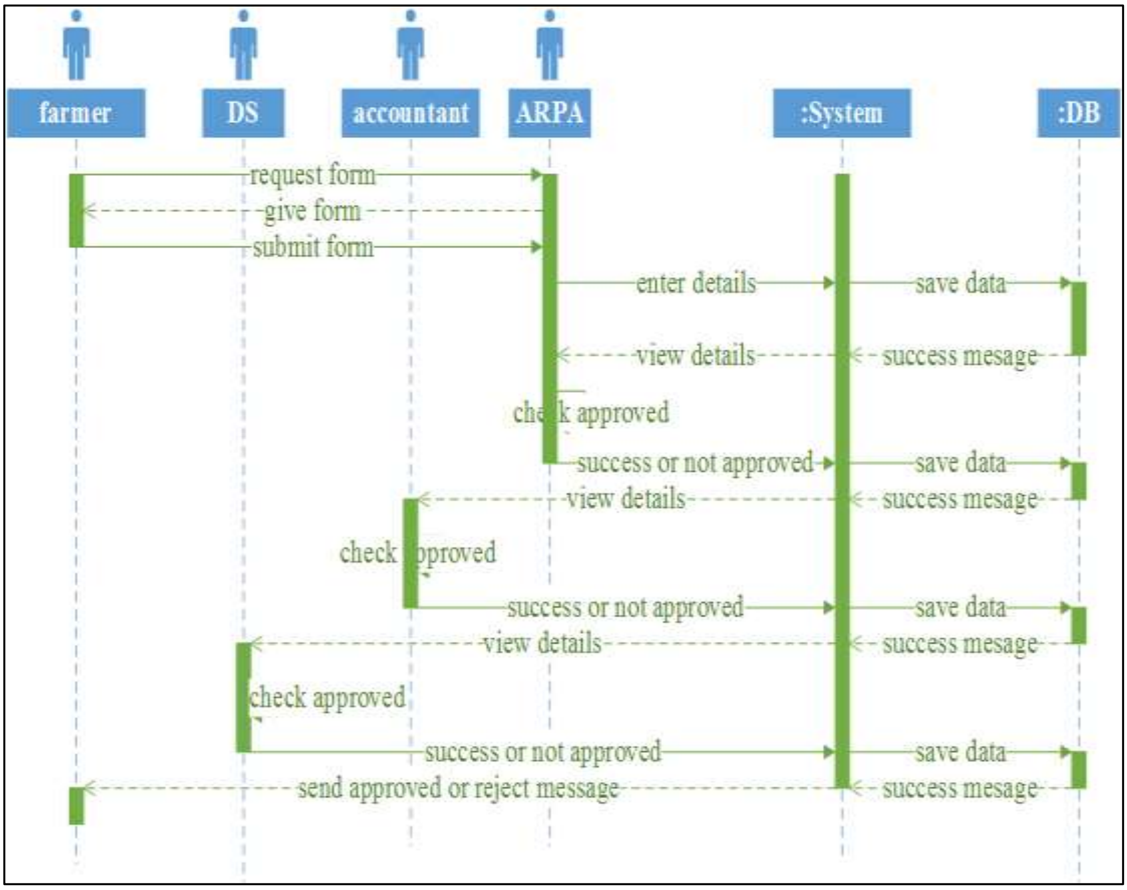


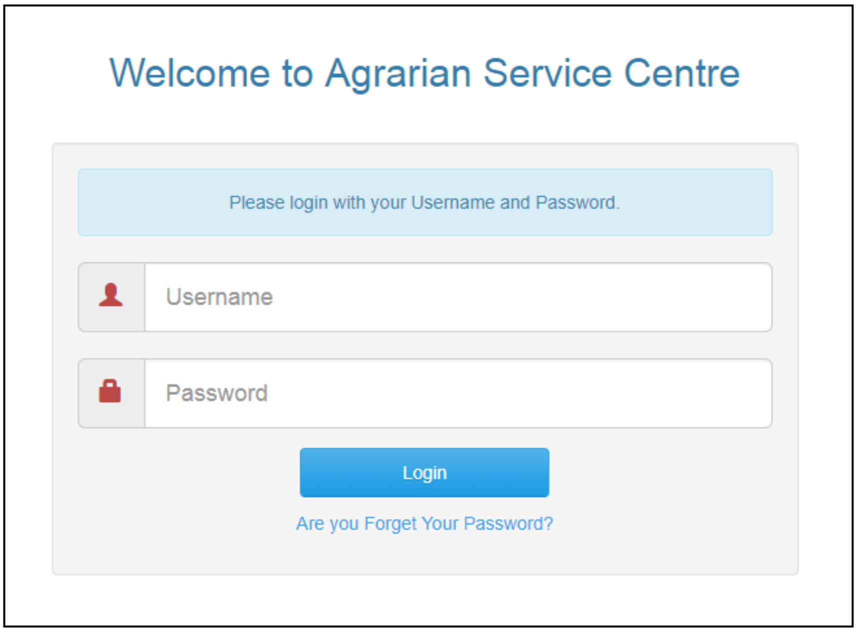
Figure B.3 Sequence Diagram for Loan apply

APPENDIX C – USER DOCUMENTATION

This documentation gives a total explanation of our system web based management system, how to handle this system.

LOGIN

Registered user allowed to login into the system. User should provide the username and password and click the submit button to gain access into the system. Figure C.1 login shown login window.

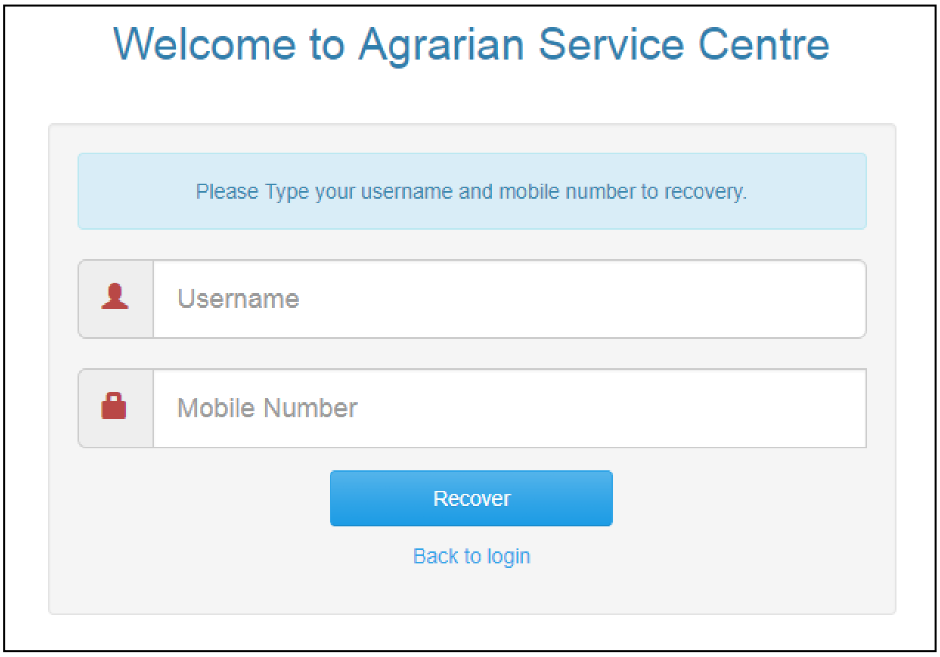


Welcome to Agrarian Service Centre

Please login with your Username and Password.

Figure C.1 Login window

User can request for the password if he /she forget it clicking on the forgot password button will direct the user to the forgot password page.Figure C.2 Forget password illustrate Forget password window.



Welcome to Agrarian Service Centre

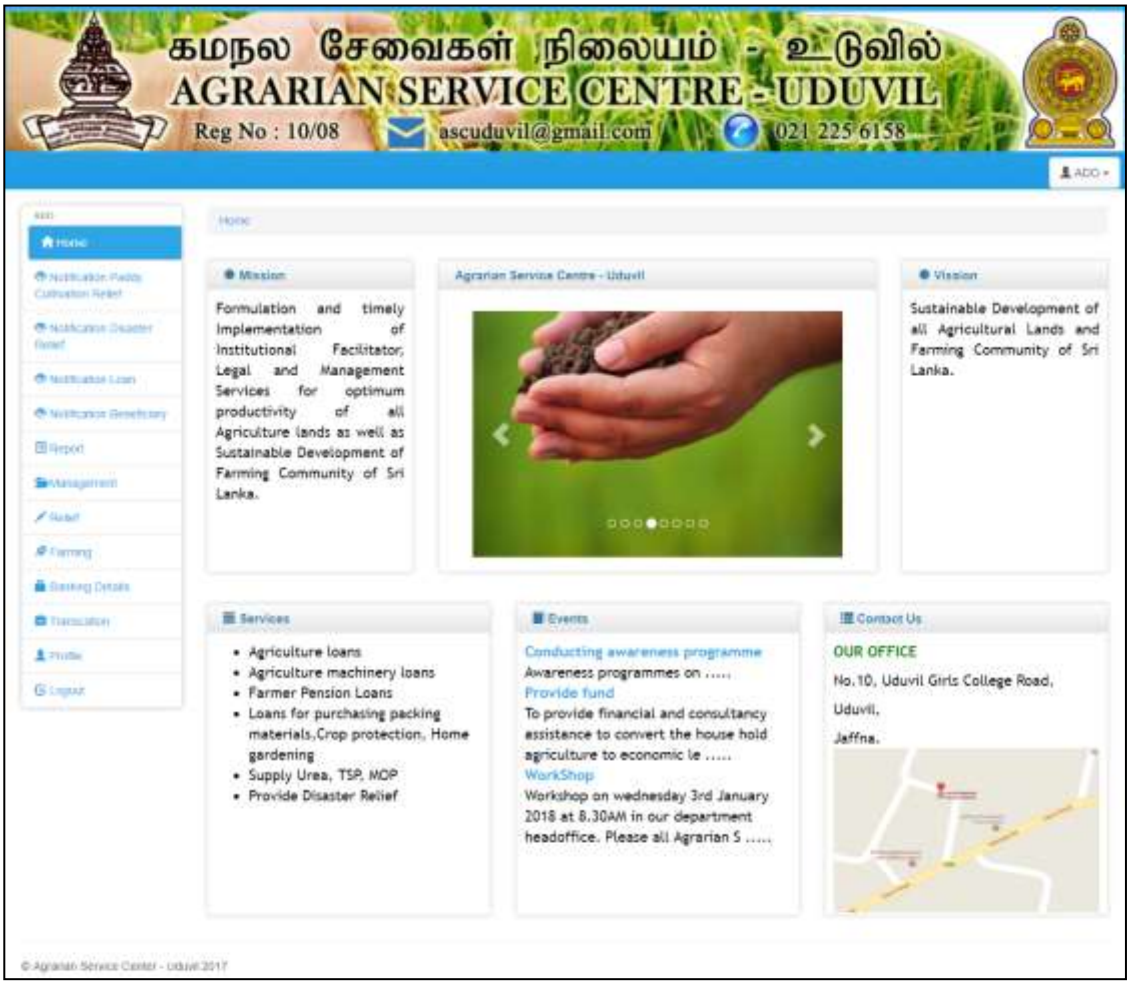
Please Type your username and mobile number to recovery.

Figure C.2 Forget Password

ADMINISTRATOR INTERFACE

After correct administration login, the administrator interface will appear. In administration menu page have some link such as Report, Management, Relief, Farming, Banking details, Transaction, Profile and Logout. The home link redirects to go to home page, in every user interface have this link with same function.

In Management there is some option menu they are Staff, Bank, Board Members, Tractors, Events and User type. Click staff from this menu there will be visible staff information. Administrator can make changes from action by clicking appropriate buttons such as view edit and delete. Like this branchBank, Board Members, Tractors, Events and User type. After Relief menu there is Disaster Relief, Paddy Cultivation relief. Next menu is Farming it has some sub menus such as Farmer, Farming organization and Farming place details. Transaction menu has Loan, Beneficiary and Expenditure. There is in report menu for generating reports. Figure C.3 Administrator shown Administrator Interface.



Formulation and timely Implementation of Institutional Facilitator, Legal and Management Services for optimum productivity of all Agriculture lands as well as Sustainable Development of Farming Community of Sri Lanka.

Sustainable Development of all Agricultural Lands and Farming Community of Sri Lanka.

- Agriculture loans
- Agriculture machinery loans
- Farmer Pension Loans
- Loans for purchasing packing materials,Crop protection, Home gardening
- Supply Urea, TSP, MOP
- Provide Disaster Relief

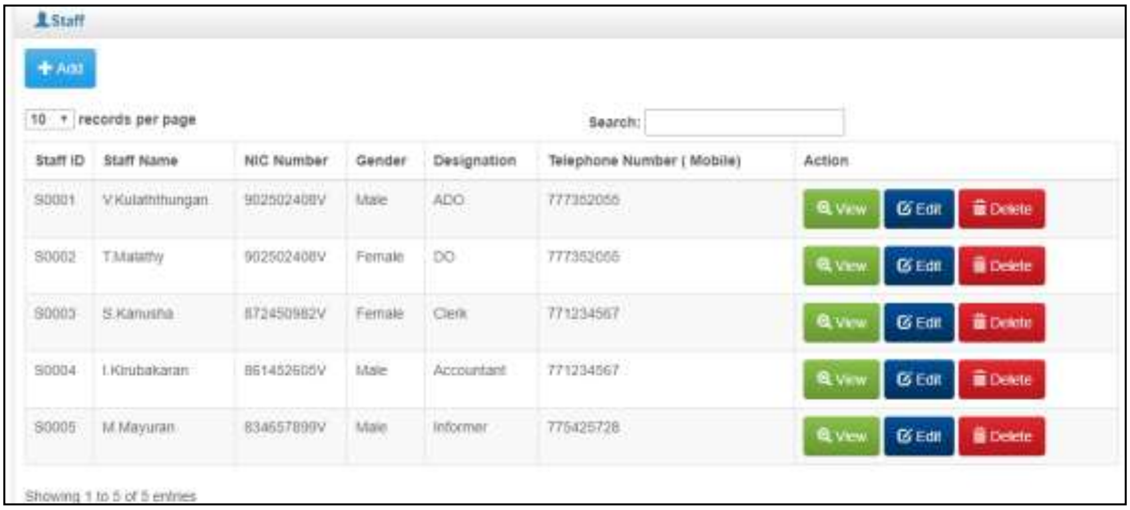
Conducting awareness programme
Awareness programmes on .....
Provide fund
To provide financial and consultancy assistance to convert the house hold agriculture to economic le .....
WorkShop
Workshop on wednesday 3rd January 2018 at 8.30AM in our department headoffice. Please all Agrarian S .....

OUR OFFICE
No.10, Uduvil Girls College Road,
Uduvil,
Jaffna.

Figure C.3 Admin Page

STAFF INFORMATION

Figure C.4 Staff information illustrates when administrator click staff menu there will be staff information. This page contains important details of staff, action buttons for mange staff, add new staff button and print staff button. Click add new staff button there will be form appear. Figure C.7 Staff Form illustrate add new staff form.



Staff

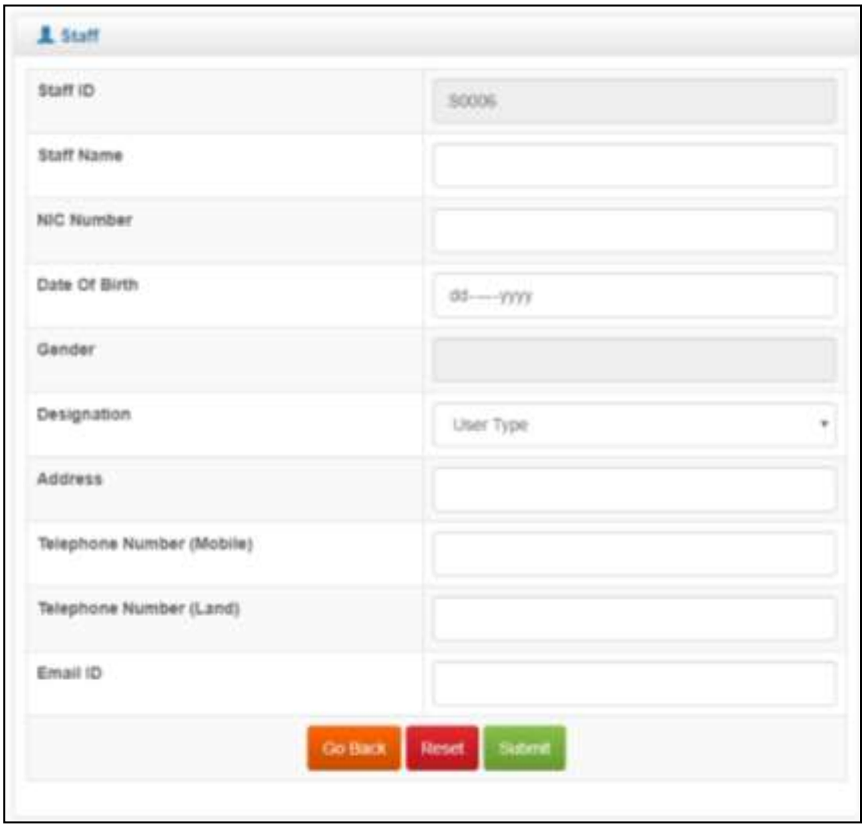
10 records per page

Search:

Staff ID	Staff Name	NIC Number	Gender	Designation	Telephone Number (Mobile)	Action
S0001	V.Kulaththungan	902502408V	Male	ADO	777352066	View Edit Delete
S0002	T.Malathy	902502408V	Female	DO	777352066	View Edit Delete
S0003	S.Kanusha	872450982V	Female	Clerk	771234567	View Edit Delete
S0004	I.Kirubakaran	861452605V	Male	Accountant	771234567	View Edit Delete
S0005	M.Mayuran	834657899V	Male	Informer	775425728	View Edit Delete

Showing 1 to 5 of 5 entries

Figure C.4 Staff Details



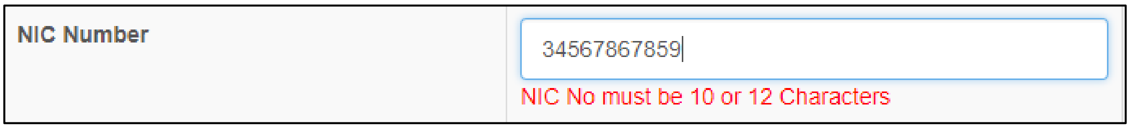
Staff

Staff ID	S0006
Staff Name	
NIC Number	
Date Of Birth	dd----yyyy
Gender	
Designation	User Type
Address	
Telephone Number (Mobile)	
Telephone Number (Land)	
Email ID	

Figure C.5 Add Staff Detail

In a staff form Staff Id automatically increased. Designation is contains select drop down menu so user can select easily. User can enter NIC No , DOB, Gender are automatically entered. NIC field user entered wrong information there will be a message. In a date field contains date picker.

If the user enter the NIC no wrong format there will be a message which mistake they do. Figure C.6 NIC Validate illustrate uncompleted the NIC field.



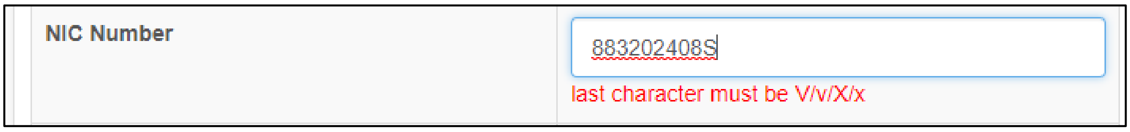
NIC Number

34567867859

NIC No must be 10 or 12 Characters

Figure C.6 NIC Validate for incomplete

Figure C.7 NIC character shown if the user enter 9numbers and the last V or X or 12 numbers otherwise there will be an error message.



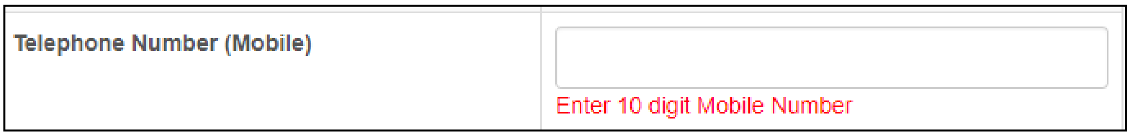
NIC Number

883202408S

last character must be V/v/X/x

Figure C.7 NIC Validate for format

Figure C.8 TP No shown if the user enters wrong format telephone number there will be an error message.

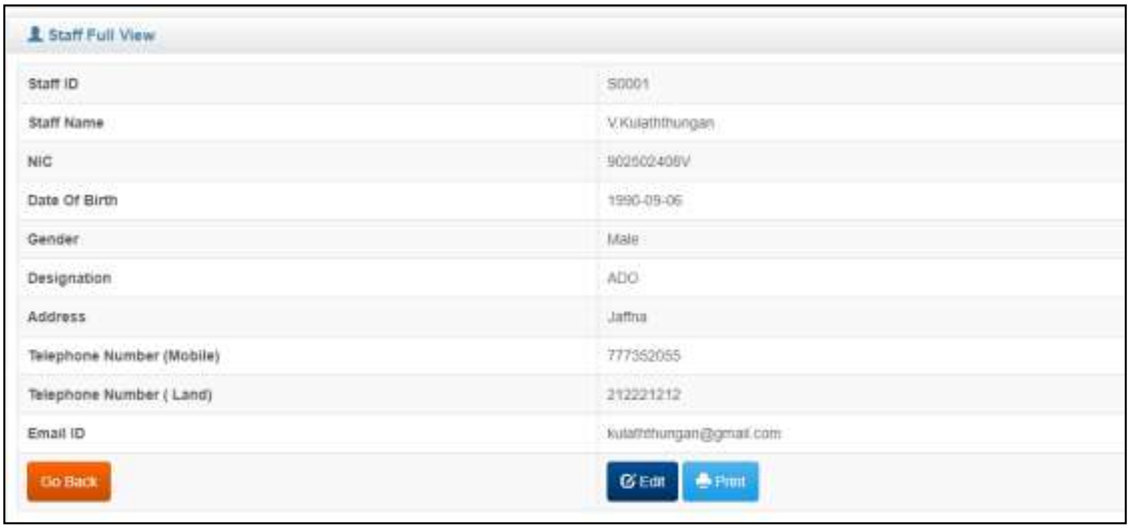


Telephone Number (Mobile)

Enter 10 digit Mobile Number

Figure C.8 TP No

In staff information table contain action field it used to view, edit and delete appropriate actions by click the button. If a user click search button there will be staff information for a particular staff. Figure C.9 Staff View shown staffs view.

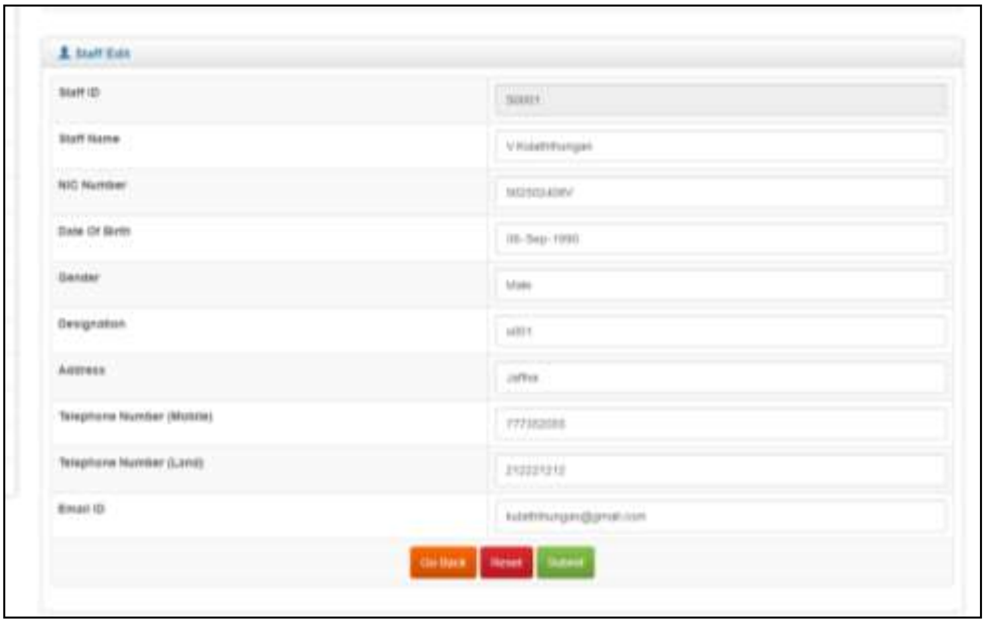


Staff Full View

Staff ID	S0001
Staff Name	V.Kulaththungan
NIC	902502408V
Date Of Birth	1990-09-06
Gender	Male
Designation	ADO
Address	Jaffna
Telephone Number (Mobile)	777352055
Telephone Number (Land)	212221212
Email ID	kulaththungan@gmail.com

Figure C.9 Staff View

If user clicks Go Back button the page go to the staff information and click the edit button go to staff edit page. If the user click edit from action go to staff edit page and user can edit wanted field and click the save button it will be change data in to the database. Edit page shown in Figure C.10 Staff Edit

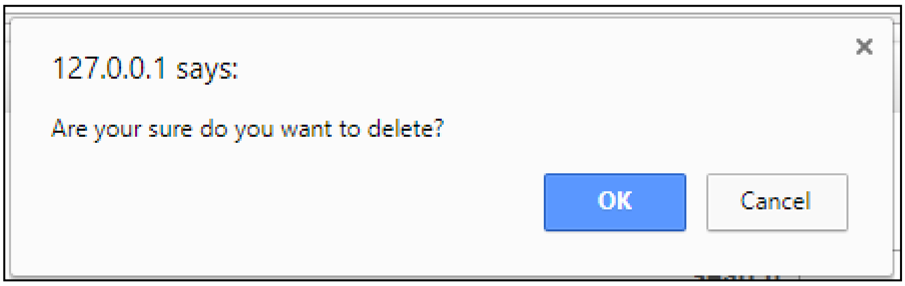


Staff Edit

Staff ID	S0001
Staff Name	V.Kulathihungan
NIC Number	902503438V
Date Of Birth	06-Sep-1990
Gender	Male
Designation	st01
Address	Jaffna
Telephone Number (Mobile)	777352055
Telephone Number (Land)	212221212
Email ID	kulathihungan@gmail.com

Figure C.10 Staff Edit

User want to delete a record click the delete button. If click delete button there will be a conformation message it has ok and cancel button. Click ok the record will be deleted. Click cancel record not delete. Figure C.11 Delete Confirm Message.



127.0.0.1 says:

Are your sure do you want to delete?

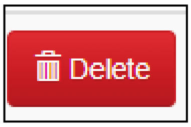
Figure C.11 Delete Confirm Message

GENERAL GUIDE LINE IN USING MANAGE

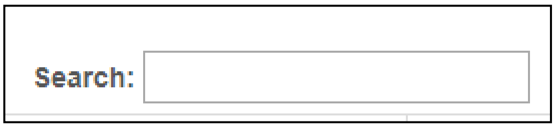
View – Click button directs you to view form to display more details of the selected data.

Edit – Click button will direct you to update form with the values of the selected record.

Delete – Click button will delete the entire record of the selected row.



Search – Entering the filtering words within the search textbox then automatically filter records in table view.



Records per page – Selecting the no of records you want to view within this page. User can select 10/25/50/100 Records per page.

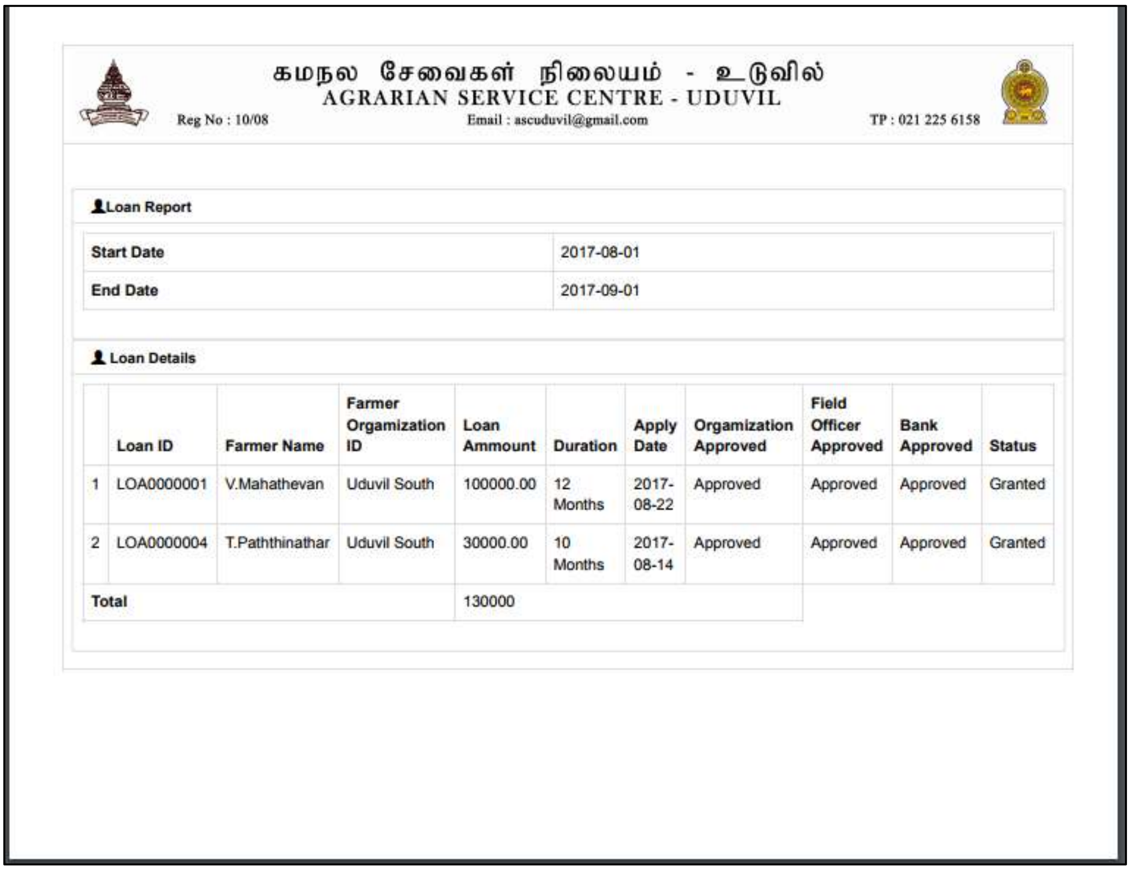
APPENDIX D – MANAGEMENT REPORTS

The system allows the ADO and DO to generate reports with their appropriate power. They can generate these reports staff details, farmer details, loan details and etc.

In the loan report, there is start date and end date, when we change the date and generate the report with based on different area.

LOAN REPORT

If user click print staff button it will show the report and print button. If click print button go to print document.Figure D.1 shown print staff detail



கமநல சேவைகள் நிலையம் - உடுவில்
AGRARIAN SERVICE CENTRE - UDUVIL
Reg No : 10/08
Email : ascuduvil@gmail.com
TP : 021 225 6158

Loan Report

Start Date	2017-08-01
End Date	2017-09-01

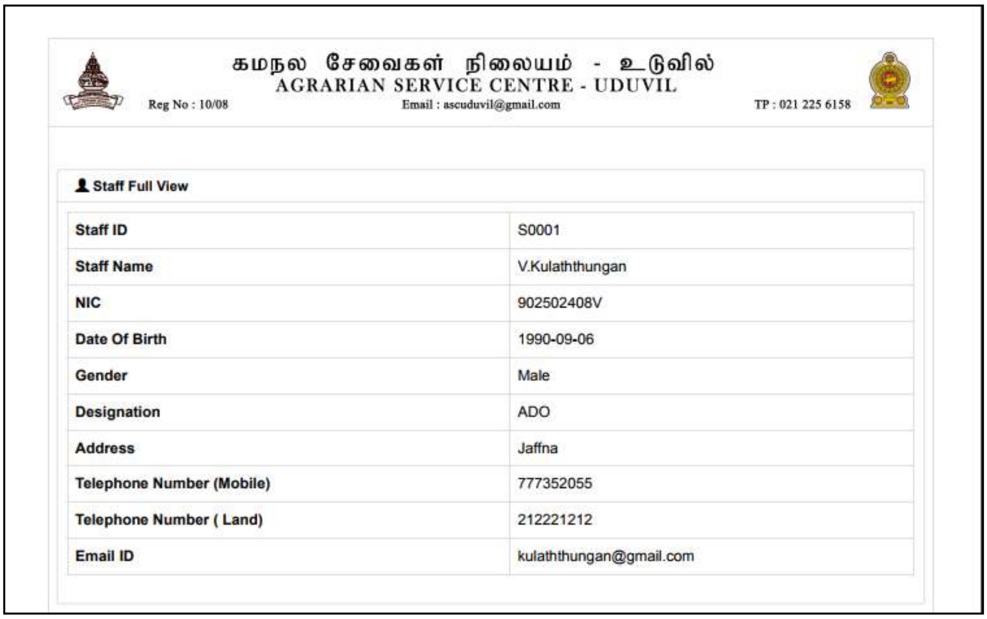
Loan Details

	Loan ID	Farmer Name	Farmer Orgamization ID	Loan Ammount	Duration	Apply Date	Orgamization Approved	Field Officer Approved	Bank Approved	Status
1	LOA0000001	V.Mahathevan	Uduvil South	100000.00	12 Months	2017-08-22	Approved	Approved	Approved	Granted
2	LOA0000004	T.Paththinathar	Uduvil South	30000.00	10 Months	2017-08-14	Approved	Approved	Approved	Granted
Total				130000						

Figure D.1 Loan Report

STAFF INFORMATION REPORT

If user click print staff button it will show the report and print button. If click print button go to print document. Figure D.2 shown print staff detail



கமநல சேவைகள் நிலையம் - உடுவில்
AGRARIAN SERVICE CENTRE - UDUVIL
Reg No : 10/08 Email : ascuduvil@gmail.com TP : 021 225 6158

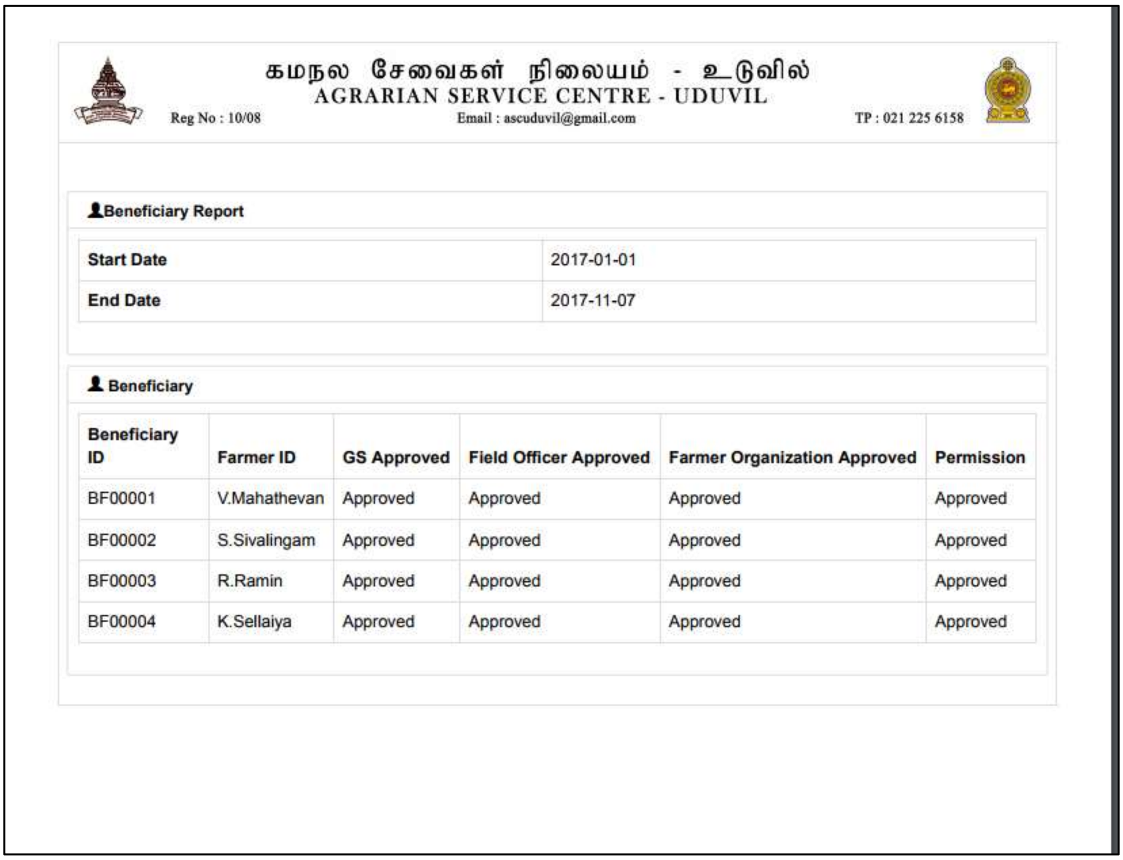
Staff Full View

Staff ID	S0001
Staff Name	V.Kulaththungan
NIC	902502408V
Date Of Birth	1990-09-06
Gender	Male
Designation	ADO
Address	Jaffna
Telephone Number (Mobile)	777352055
Telephone Number (Land)	212221212
Email ID	kulaththungan@gmail.com

Figure D.2 Staff Detail Report

BENEFICIARY REPORT

If user click print staff button it will show the report and print button. If click print button go to print document. Figure D.3 shown print beneficiary detail



கமநல சேவைகள் நிலையம் - உடுவில்
AGRARIAN SERVICE CENTRE - UDUVIL
Reg No : 10/08 Email : ascuduvil@gmail.com TP : 021 225 6158

Beneficiary Report

Start Date	2017-01-01
End Date	2017-11-07

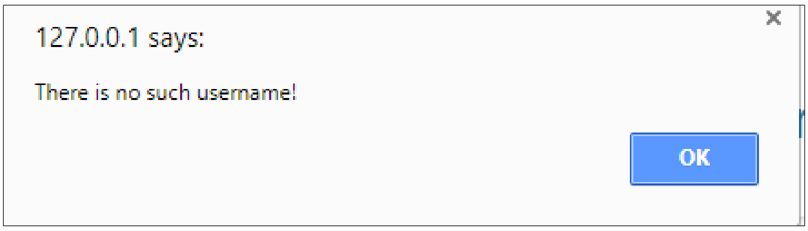
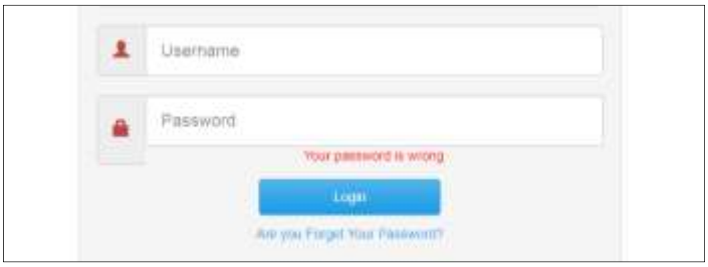
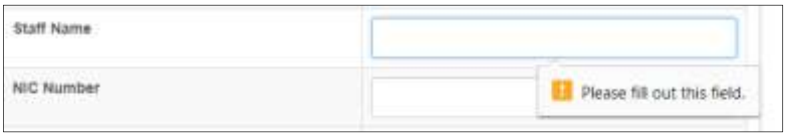
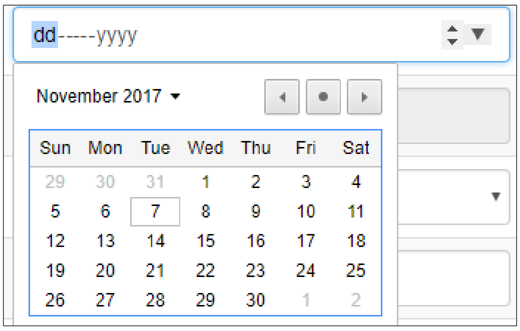
Beneficiary

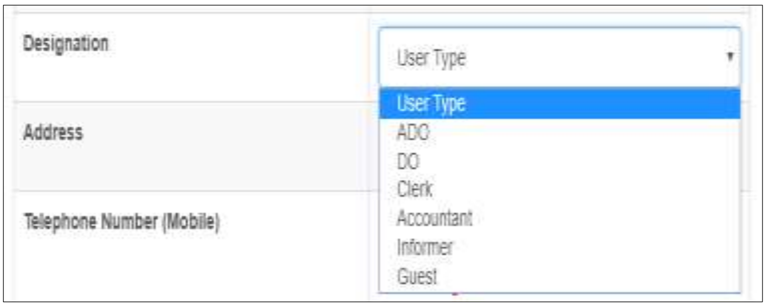
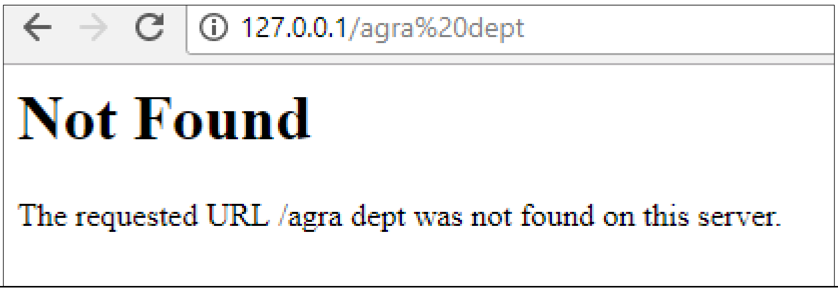
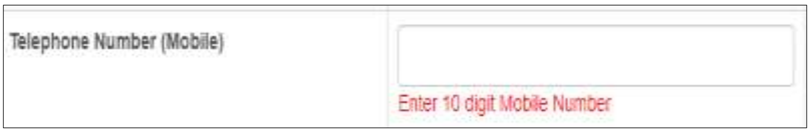
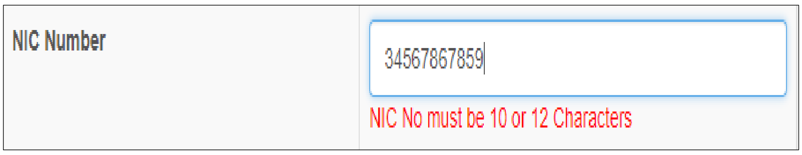
Beneficiary ID	Farmer ID	GS Approved	Field Officer Approved	Farmer Organization Approved	Permission
BF00001	V.Mahathevan	Approved	Approved	Approved	Approved
BF00002	S.Sivalingam	Approved	Approved	Approved	Approved
BF00003	R.Ramin	Approved	Approved	Approved	Approved
BF00004	K.Sellaiya	Approved	Approved	Approved	Approved

Figure D.3 Beneficiary Detail Report

APPENDIX E – TEST RESULTS

Since the “Web Based management system for Agrarian Service Centre” is mainly developed for staff usage along with management. Appropriate error messages and success messages and information were displayed to notify the user about the activities. Detected errors were carried out to provide the enhanced working system to the client. The following test cases and resulting screenshots given below to enlighten the user about the testing phases carried out.

Test Case	Screen shot	Status
When we enter wrong username	 127.0.0.1 says: There is no such username! 	Pass
When we enter wrong password	 Your password is wrong 	Pass
Submit without enter value in mandatory field	 Staff Name NIC Number Please fill out this field.	Pass
Date Picker	 dd-----yyyy November 2017 Sun Mon Tue Wed Thu Fri Sat 29 30 31 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 1 2	Pass

Click dropdown to check does the values loaded from database	 Designation User Type User Type ADO DO Clerk Accountant Informer Guest Address Telephone Number (Mobile)	Pass
Enter wrong URL	 127.0.0.1/agra%20dept Not Found The requested URL /agra dept was not found on this server.	Pass
Enter wrong format of telephone number	 Telephone Number (Mobile) Enter 10 digit Mobile Number	Pass
Enter the wrong format of NIC Number	 NIC Number 34567867859 NIC No must be 10 or 12 Characters	Pass

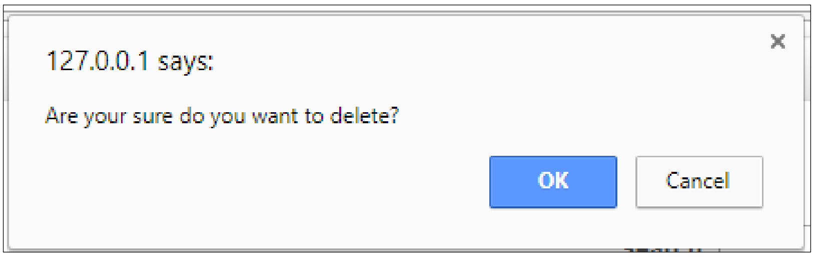
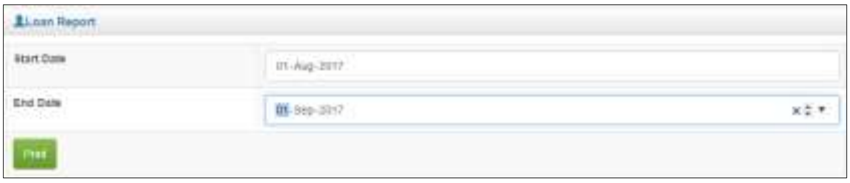
Delete Record confirm Message	 127.0.0.1 says: Are your sure do you want to delete? 	Pass
Display print button in report	 Loan Report Start Date: 01-Aug-2017 End Date: 01-Sep-2017 	Pass

Table E.1 Test Result

APPENDIX F – CODE LISTING

This section provides more detailed view of the coding to the users. Since the code section is very lengthy, only important section of the coding are provided below with appropriate comments to make the user more comfortable with the coding.

LOGIN

Login page gets the username and password using the post method and check with the database whether it is matches the username and password in the database.

```
<?php

if(!isset($_SESSION))

{

session_start();

}

date_default_timezone_set("Asia/Colombo");

include("connection.php");

if(isset($_POST["btnsubmit"]))

{

$entereruname=$_POST["txtusername"];

$enterpwd=$_POST["txtpassword"];

$sqlusername="SELECT * FROM login WHERE user_name='$entereruname'";

$resultusername=mysql_query($sqlusername)ordie("sql error in sqlusername ".mysql_error());

if(mysql_num_rows($resultusername)>0)

{

//username is correct

$rowusername=mysql_fetch_assoc($resultusername);

$sqlpassword="SELECT * FROM login WHERE user_name='$entereruname' AND password='$enterpwd'";

$resultpassword=mysql_query($sqlpassword)ordie("sql error in sqlpassword ".mysql_error());

if(mysql_num_rows($resultpassword)>0)

{

//username and password is correct
```

```
$sqlusertype="SELECT * FROM user_type WHERE user_type_id='$rowusername[user_type_id]'";
$resultusertype=mysql_query($sqlusertype)ordie("sql error in sqlusertype ".mysql_error());
$rowusertype=mysql_fetch_assoc($resultusertype);
$_SESSION["username"]=$entererуname;
$_SESSION["usertype"]=$rowusertype["user_type"];
//set attempt to zero
$sqlupdate="UPDATE login SET attempt='0' WHERE user_name='$entereruname'";
$resultupdate=mysql_query($sqlupdate)ordie("sql error in sqlupdate ".mysql_error());
//generate login_history_id
$sqllogin_history_id="SELECT login_history_id FROM login_history ORDER BY login_history_id DESC LIMIT 1";
$resultlogin_history_id=mysql_query($sqllogin_history_id)ordie("sql error in sqllogin_history_id ".mysql_error());
if(mysql_num_rows($resultlogin_history_id)>0)
{
//second or more
$rowlogin_history_id=mysql_fetch_assoc($resultlogin_history_id);
$login_history_id=++$rowlogin_history_id["login_history_id"];
}
else
{
//first time
$login_history_id="LH00000001";
}
$todaydate=date("Y-m-d");
$thistime=date("H:i:s");
//insert into login_history
$sqlloginhistory="INSERT INTO login_history(login_history_id,user_name,date,log_in_time)
VALUES('".$login_history_id."','".$entereruname."','".$todaydate."','".$thistime."')";
$resultloginhistory=mysql_query($sqlloginhistory)ordie("sql error in sqlloginhistory ".mysql_error());
```

```
header("location:index.php");

}

elseif($rowusername["attempt"]<3)

{

//attempt less than three

$sqlupdate="UPDATE login SET attempt=attempt+1 WHERE user_name='$entererunname'";

$resultupdate=mysql_query($sqlupdate)ordie("sql error in sqlupdate ".mysql_error());

echo'<script>alert("Your password is wrong");</script>';

}

else

{

//forget password redirect

$_SESSION["forgetusername"]=$entererunname;

echo'<script>alert("You attempt more than three time; please recover your password!");

window.location.href="forget.php";</script>';

}}

else

{

//username is wrong

echo'<script>alert("There is no such username!");</script>';

}

}

?>
```

LOG OUT

When the user logging out the system all session variables will be destroyed and the user will be directed to index page.

```
<?php

if(!isset($_SESSION))

{

session_start();

}

date_default_timezone_set("Asia/Colombo");

include("connection.php");

$thistime=date("H:i:s");

$username=$_SESSION["username"];

//get last login_history_id for login user

$sqllasthistoryid="SELECT login_history_id FROM login_history WHERE
user_name='$username' ORDER BY login_history_id DESC LIMIT 1";

$resultlasthistoryid=mysql_query($sqllasthistoryid)ordie("sql error in sqllasthistoryid
".mysql_error());

$rowlasthistoryid=mysql_fetch_assoc($resultlasthistoryid);

$lasthistoryid=$rowlasthistoryid["login_history_id"];

//set the logout time

$sqlupdateloginhistory="UPDATE login_history SET log_out_time='".$thistime."'

                        WHERE login_history_id='".$lasthistoryid."'";

$resultupdateloginhistory=mysql_query($sqlupdateloginhistory)ordie("sql error in
sqlupdateloginhistory ".mysql_error());

//session destroy

session_destroy();

if(isset($_GET["cp"]))

{

header("location:login.php");

}

else

{
```

```
header("location:index.php");

}

?>
```

ADD NEW RECORDS

```
if(isset($_POST["btnsubmitnew"]))

{

$sqlinsert=" INSERT INTO
staff(staff_id,staff_name,nic,dob,gender,designation,address,mobile_number,land_number,em
ail)

VALUES('".mysql_real_escape_string($_POST["txtstaffid"])."',

            '".mysql_real_escape_string($_POST["txtstaffname"])."',

            '".mysql_real_escape_string($_POST["txtnic"])."',

            '".mysql_real_escape_string($_POST["txtdob"])."',

            '".mysql_real_escape_string($_POST["txtgender"])."',

            '".mysql_real_escape_string($_POST["txtdesignation"])."',

            '".mysql_real_escape_string($_POST["txtaddress"])."',

            '".mysql_real_escape_string($_POST["txtmobile"])."',

            '".mysql_real_escape_string($_POST["txtland"])."',

            '".mysql_real_escape_string($_POST["txtemail"])."')";

$resultinsert=mysql_query($sqlinsert)ORdie("sql error in sqlinsert".mysql_error());


$sqlinsert=" INSERT INTO login(user_name,password,user_type_id,attempt,code,status)

VALUES('".mysql_real_escape_string($_POST["txtstaffid"])."',

            '".mysql_real_escape_string($_POST["txtnic"])."',

            '".mysql_real_escape_string($_POST["txtdesignation"])."',

            '".mysql_real_escape_string(0)."',

            '".mysql_real_escape_string(0)."',

            '".mysql_real_escape_string("Active")."')";

$resultinsert=mysql_query($sqlinsert)ORdie("sql error in sqlinsert".mysql_error());

if($resultinsert)

{
```

```
echo'<script>alert("successfully insert");</script>';

}

}
```

AUTO NUMBER IDS WHEN ADDING RECORDS

When adding record into the database some auto generated values also added with the user entered values. The given below code describe how the auto number or ids get from the database and increase sequentially.

```
<?php

$sqlfarmerid="SELECT farmer_id FROM farmer ORDER BY farmer_id DESC LIMIT 1";

$resultfarid=mysql_query($sqlfarmerid)ordie("sql error in sqlfarmerid ".mysql_error());

if(mysql_num_rows($resultfarid)>0)

{

//second or more

$rowfarmerid=mysql_fetch_assoc($resultfarid);

$farmerid=++$rowfarmerid["farmer_id"];

}

else

{

//first time

$farmerid="FA000001";

}

?>
```

The field that contains auto generated value should be set as read only to prevent user from changing it.

```
<inputtype                                                            =
"text"name="txtfarmerid"value="<?phpecho$farmerid;?>"id="txtfarmerid"readonlyclass
="form-control"></td>
```

EDIT/UPDATE RECORDS

```
if(isset($_POST["btnsubmitedit"]))

{

$sqlupdate="UPDATE farmer SET
```

```
                farmer_name='".mysql_real_escape_string($_POST["txtfarmername"])."',
nic='".mysql_real_escape_string($_POST["txtnic"])."' ,
dob='".mysql_real_escape_string($_POST["txtdob"])."' ,
gender='".mysql_real_escape_string($_POST["txtgender"])."' ,
address='".mysql_real_escape_string($_POST["txtaddress"])."' ,
                mobile_number='".mysql_real_escape_string($_POST["txtmobile"])."' ,
land_number='".mysql_real_escape_string($_POST["txtland"])."' ,
                farmer_type='".mysql_real_escape_string($_POST["txtfarmertype"])."' ,
                gs_division='".mysql_real_escape_string($_POST["txtgsdivision"])."'
                WHERE farmer_id= '".mysql_real_escape_string($_POST["txtfarmerid"])."'";
$resultupdate=mysql_query($sqlupdate)ORdie("sql error in sqlupdate".mysql_error());
if($resultupdate)
{
echo'<script>alert("successfully update"); window.location.href="index.php?pg=farmer.php&option=fullview&farmerid='.$_POST["txtfarmerid"].'";</script>';
}
}
```

DELETE RECORDS

Deleting record from the database such as ‘Delete Item’, ‘Delete Route’, ‘Delete News’ and ‘Delete Users’. But deleting records from database is only allowed to administrator to ensure security.

```
if($_GET["option"]=="delete")
{
if($usertype=="ADO"||$usertype=="DO"||$usertype=="Clerk")// for who can goto the delete option
{
$farmerid=$_GET["farmerid"];
$sqlfarmerdelete="DELETE FROM farmer WHERE farmer_id='$farmerid'";
$resultfarmerdelete=mysql_query($sqlfarmerdelete)ordie("sql error in sqlfarmerdelete ".mysql_error());
if($resultfarmerdelete)
{
echo'<script>alert("Successfully Deleted");
window.location.href="index.php?pg=farmer.php&option=view";</script>';
}
}
else
{
header("location:index.php?pg=farmer.php&option=view");
}
}
```

TELEPHONE NUMBER VALIDATION

```
functionphonenumber()// Mobile No
{
varphoneno=/^\d{10}$/;
if(document.getElementById("txtmobile").value=="")
{
}
else
{
if(document.getElementById("txtmobile").value.match(phoneno))
{
//return true;
hand();
}
else
{
//alert("Enter 10 digit Mobile Number");
document.getElementById("mobile_error").innerHTML="Enter 10 digit Mobile Number";
document.getElementById("txtmobile").value="";
document.getElementById("txtmobile").focus()=true;
returnfalse;

}

}

}

function hand()

{

varstr=document.getElementById("txtmobile").value;

var res =str.substring(0,2);

if(res=="07")

{


document.getElementById("mobile_error").innerHTML="";

returntrue;

}

else

{

//alert("enter 10 digit of Mobile Number; start with 07");

document.getElementById("mobile_error").innerHTML="enter 10 digit of Mobile Number; start with 07";

document.getElementById("txtmobile").value="";

document.getElementById("txtmobile").focus()=true;

returnfalse;

}


}
```

BEFORE DELETE ENTRY CONFIRM MESSASE

```
functiondeletedata()

{

var x=confirm("Are your sure do you want to delete?");

if(x)
{
returntrue;
}
else
{
returnfalse;
}
}
```

NIC NUMBER VALIDATION CODE

```
functionnicnumber()//nic validation start
{
varnic=document.getElementById("txtnic").value;
if(nic.length==10)//for nic length 10
{
var gender=nic.substring(2,5);
var nicformat1=/^[0-9]{9}[a-zA-Z0-9]{1}$/;
if(nic.match(nicformat1))// validate first 9 character are number
{
var nicformat2=/^[0-9]{9}[vVxX]{1}$/;
if(nic.match(nicformat2))//validate first 9 character are number and last v/x
{
if(gender>500)
{
document.getElementById("txtgender").value="Female";
}
else
{
document.getElementById("txtgender").value="Male";
}
calculatedob(nic);//if correct goto this function
}
else
{
alert("last character must be V/v/X/x");
document.getElementById("txtnic").value="";
document.getElementById("txtdob").value="";
}
}
else
{
alert("First 9 characters must be numbers");
document.getElementById("txtnic").value="";
document.getElementById("txtnic").focus();
document.getElementById("txtdob").value="";
}
}
elseif(nic.length==12)//for nic length 12
{
var gender=nic.substring(4,7);
var nicformat3=/^[0-9]{12}$/;
if(nic.match(nicformat3))
{
if(gender>500)
{
document.getElementById("txtgender").value="Female";
}
else
{
document.getElementById("txtgender").value="Male";
}
calculatedob(nic);//if correct goto this function
```

```
}
else
{
alert("All 12 characters must be number");
document.getElementById("txtnic").value="";
document.getElementById("txtnic").focus();
document.getElementById("txtdob").value="";
}
}
elseif(nic.length==0)//nic field blank
{

}
else//nic is not 10 or 12
{
alert("NIC No must be 10 or 12 Characters");
document.getElementById("txtnic").value="";
document.getElementById("txtnic").focus();
document.getElementById("txtdob").value="";
}
}
```

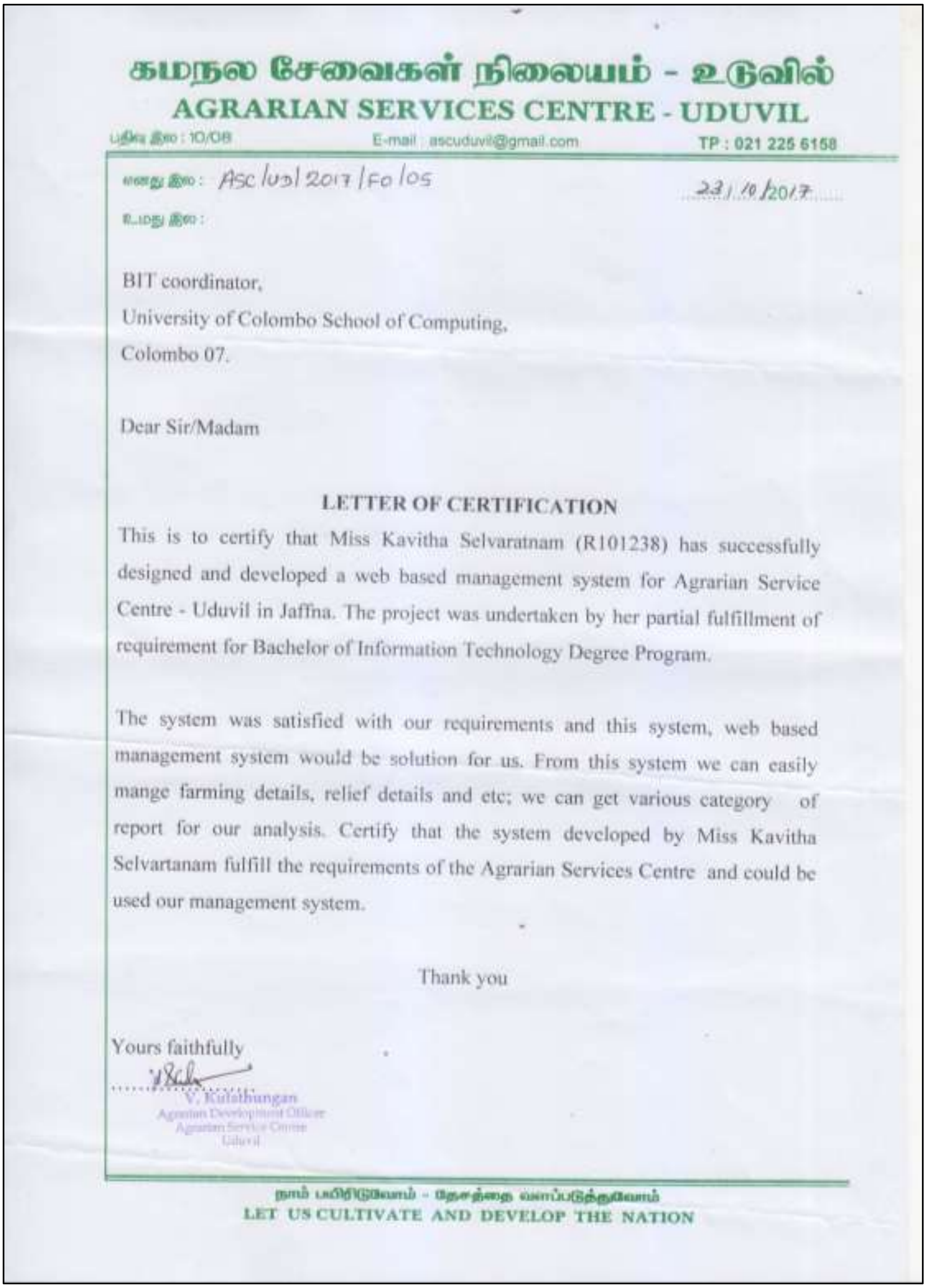
GENERATE REPORT CODE

```
functionload_loan_report()
{
varstartdate=document.getElementById("txtstartdate").value;
varenddate=document.getElementById("txtenddate").value;
varxmlhttp=newXMLHttpRequest();
xmlhttp.onreadystatechange=function()
{
if(xmlhttp.readyState==4&&xmlhttp.status==200)
{
```

```
document.getElementById("load_ajax_loan").innerHTML=xmlhttp.responseText;
document.getElementById("btnprint_loan_report").disabled=false;
}
};
xmlhttp.open("GET","report_ajax.php?option=loan_report&startdate="+startdate+"&enddate="+enddate,true);
xmlhttp.send();
}
if($_GET["option"]=="loan_report")//for in print page
{
echo'<body onload="load_loan_report()">';
}
if($_GET["option"]=="loan_report")
{
echo'<div class="row">
<div class="box col-md-12">
<div class="box-inner">
<div class="box-header well" data-original-title="">
<h2><i class="glyphiconglyphicon-user"></i>Loan Report</h2>
</div>
<div class="box-content">
<table class="table table-striped table-bordered bootstrap-datatabledatatable responsive">';
if(!isset($_GET["pr"]))
{
echo'<tr><th>Start Date</th><td><input type="date" name="txtstartdate" id="txtstartdate" class="form-control" onchange="load_loan_report()" value="'.date("Y-m-d").'"></td></tr>';
}
else
{
echo'<tr><th>Start Date</th><td><input type="hidden" name="txtstartdate" id="txtstartdate" class="form-control" value="'.$_GET["startdate"].'">'.$_GET["startdate"].'</td></tr>';
}
```

```
if(!isset($_GET["pr"]))

{

echo'<tr><th>End Date</th><td><input type="date" name="txtenddate" id="txtenddate"
class="form-control" onchange="load_loan_report()" value="'.date("Y-m-d").'"></td></tr>';

}

else

{

echo'<tr><th>End Date</th><td><input type="hidden" name="txtenddate" id="txtenddate"
class="form-control" value="'.$_GET["enddate"].'">'.$_GET["enddate"].'</td></tr>';

}

if(!isset($_GET["pr"]))

{

echo'<tr><td colspan="2"><input type="button" onclick="print_load_loan_report()" disabled
class="btnbtn-success" value="Print" name="btnprint_loan_report"
id="btnprint_loan_report"></td></tr>';

}

echo'</table></div></div>';

echo'<div id="load_ajax_loan"></div>';

echo'</div></div>';

}
```

APPENDIX G - CLIENT CERTIFICATE



கமநல சேவைகள் நிலையம் - உடுவில்

AGRARIAN SERVICES CENTRE - UDUVIL

பதிவு இல : 10/08 E-mail : ascuduvil@gmail.com TP : 021 225 6158

எமது இல : ASC/UDl/2017/FO/05 23/10/2017

உமது இல :

BIT coordinator,
University of Colombo School of Computing,
Colombo 07.

Dear Sir/Madam

LETTER OF CERTIFICATION

This is to certify that Miss Kavitha Selvaratnam (R101238) has successfully designed and developed a web based management system for Agrarian Service Centre - Uduvil in Jaffna. The project was undertaken by her partial fulfillment of requirement for Bachelor of Information Technology Degree Program.

The system was satisfied with our requirements and this system, web based management system would be solution for us. From this system we can easily mange farming details, relief details and etc; we can get various category of report for our analysis. Certify that the system developed by Miss Kavitha Selvartanam fulfill the requirements of the Agrarian Services Centre and could be used our management system.

Thank you

Yours faithfully

V. Kulathungan
Agrarian Development Officer
Agrarian Service Centre
Uduvil

நாம் பயிரிடுவோம் - தேசத்தை வளப்படுத்துவோம்

LET US CULTIVATE AND DEVELOP THE NATION

GLOSSARY

Apache – Open source web server.

PHP – Hypertext Pre-Processer, it is one of the famous server side scripting languages.

CSS – Cascading Style Sheet described how the structured element must be rendered on screen or on media.

JavaScript – it is one of the Client-side scripting languages.

Database – The backend storage of system.

SDLC – Software Development Life Cycle is a structure imposed on the development of a software product.

OO – Object Oriented, is an approach to designing modular, reusable software systems.

UML – Unified Modeling Language, is a graphical language for visualizing, specifying, constructing and documenting the artifacts of a software-intensive system.

WWW – World Wide Web.

Web browser – is software to access the web.

GUI – Graphical User Interface, utilized to support to user to interaction with system