



“LifeLink” – Life Insurance System For ALMAO

R.M.N.P.B. Ratnayake

BIT Registration Number: R004702

Index Number: 0047023

Supervisor’s Name: Prasanna Weerakoon

Submission Month: November Submission Year: 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

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 summary to be made available to outside organization.



Date : 2017.11.07

R.M.N.P.B. Ratnayake

(Name of Candidate)

Countersigned by :



Date : 2017.11.07

Prasanna Weerakoon

(Name of Supervisor)

Abstract

ALMAO (All Lanka Mutual Organization) is an Insurance Company. It has a system but there isn't code. In addition to that this software has a lot of errors and lack of functionalities. It uses Ms Access database as its database. Further ; it isn't a web base system. This software can run only on Windows 98, Windows 2000 machines. These are the problems with the current system. So ALMAO has a requirement for a new system.

Proposed system called LifeLink system had to overcome above problems. Because of LifeLink system is a new system it has a code. Then we can correct errors of existing system and can add more functionalities to new system."LifeLink" new system is used MsSQL Server as its database. Ms SQL Server is better than Ms Access database. New system is a remote logging system. Then this LifeLink system can be accessed from anywhere .This new system can be run on any modern Operating System.

LifeLink Software has been developed based on the Rational Unified Process method. For develop this LifeLink software ; i used Visual Studio .Net and its VB.Net programming language. In addition to that I used MsSQL Server as its database and Crystal Reports software for reporting purposes. I used Helawadana Piwisuma software for display Sinhala fonts in the LifeLink software..

This LifeLink software has been developed on time and very well. It has achieved all the prospects of a new system for ALMAO.

Acknowledgement

First I would like to thanks to University of Colombo School of Computing(UCSC) ; because this Individual Project module is very important to us to learn how to develop a Software Project by own. It is a hurdle for us who are learning Information Technology(IT) field.

In addition to that I would like to thanks Project Co-ordinator of UCSC. She did very good job for us. BIT Individual Project VLE web site also a very good one. There were very good information about “How to develop a Project Report”.

My supervisor ; Mr.Prasanna Weerakoon gave me very good advices about LifeLink Software project. So I would like to thanks him also.

IT Manager and General Manager of Sanasa Insurance Company LTD ; they are my bosses. They kept confidence me about” I will be able to do this serious task very well”. They gave me advices and nessesory facilities to do this task good. So I like to thanks them also.

My special thanks to ALMAO manager and it’s staff. They were very friendly and they gave me nessesory information and facilities and very good help. I like to thanks them again.

Sanasa Insurance IT department’s staff who are System Administrator and Network Adminstrator done the very good job for me. They are my friends in IT department. I like to thanks them also.

Table of Contents

Declaration.....	ii
Abstract.....	iii
Acknowledgement.....	iv
Table of Content,,,.....	v
List of Figures	x
List of Tables	xiii
List of Acronyms	xiv
Chapter 01 - Introduction	1
1.1 Introduction	1
1.2 Motivation of the Project	1
1.3 Objective and the Scope of the Project	2
Chapter 02 – Analysis.....	4
2.1 Introduction	4
2.2 Fact Gathering Techniques	4
2.2.1 Observations	4
2.2.2 Interviews	4
2.2.3 Sampling and Documentation	4
2.2.4 Prototyping	5

2.3 Current System and the Process	5
2.4 Functional Requirements	6
2.4.1 Collecting Premiums.....	6
2.4.2 Administration.....	6
2.4.3 Claim Payments.....	6
2.4.4 Customer Inquiry And Maintenance.....	6
2.4.5 Agent Inquiry And Maintenance.....	6
2.4.6 Society Inquiry And Maintenance.....	6
2.4.7 Display Ledger Information.....	6
2.4.8 Policy Inquiry And Maintenance.....	6
2.4.9 Display Pre-Maturity Policy.....	6
2.4.10 Claim Inquiry And Maintenance.....	6
2.4.11 Generate Reports.....	6
2.5 Non Functional Requirements	6
2.5.1 Accuracy.....	6
2.5.2 Security.....	6
2.5.3 Usability	7
2.5.4 Interoperability	7
2.5.5 Performance	7
2.5.6 Availability	7
2.5.7 Reliability	7
2.5.8 Capacity	7

2.6 Existing Similar Systems	7
2.6.1 SIS.Net 10	8
2.7 Methodology for the Proposed System	9
Chapter 03 – Design Of Solution.....	11
3.1 Introduction	11
3.2 Alternative Solutions to the System	11
3.3 Object Oriented Designing.....	11
3.4 Database Design	12
3.5 User Interface Design	13
Chapter 04 – Implementation.....	24
4.1 Introduction	24
4.2 Hardware and Software Requirements	24
4.2.1 Hardware Requirements For Server Computer.....	24.
4.2.2 Hardware Requirements For Client Computer.....	24
4.2.3 Software Requirements For Server Computer	24
4.2.4 Software Requirements For Client Computer	25
4.3 Development Tools.....	25
4.3.1 .NET Framework	25
4.3.2 MsSQL Server 2008	25
4.3.3 Ms Visual Studio.Net 2008	26
4.3.4 VB.Net Language	26

4.4 Major Code Segments	27
4.4.1 Database Connection	27
4.4.2 Login Handling And Data Validation	27
4.4.3 Search, Insert, Update. Data	29
4.4.4 Reporting	33
4.4.5 Re-usable Components	31
Chapter 05 – Evaluation	34
5.1 Introduction	34
5.2 Software Testing	34
5.3 Types of Testing	34
5.3.1 Unit Testing	35
5.3.2 Integration Testing	35
5.3.3 System Testing	35
5.3.4 Acceptance Testing	35
5.3.5 Regression Testing	35
5.4 System Test Plan	36
5.5 System Test Cases	36
5.5.1 Test Cases for User Login Module	36
5.5.2 Test Cases for Collecting Premium Module	37
5.6 User Evaluation	37
Chapter 06 – Conclusion	40
6.1 Introduction	40

6.2 Future Enhancement	40
6.3 Lesson Learnt	40
References.....	42
Appendix A - System Documentation.....	47
Appendix B - Design Documentation	51
Appendix C - User Documentation	56
Appendix D - Management Reports.....	75
Appendix E - Test Results.....	82
Appendix F - Code Listing.....	86
Appendix G - My Service Letter.....	91
Appendix H - Client Certificate.....	92
Glossary	93
Index	94

List of Figures

Figure 2.1	High Level UseCase Diagram	5
Figure 2.2	SIS.Net 10	8
Figure 3.1	High Level UseCase Diagram	12
Figure 3.2	Database Diagram	13
Figure 3.3	Login Form	14
Figure 3.4	Customer Search Form	14
Figure 3.5	Collecting Premium Form	15
Figure 3.6	Policy Inquiry Form	16
Figure 3.7	Customer Maintenance Form	17
Figure 3.8	Claim Maintenance Form	18
Figure 3.9	Claim Paid Form	19
Figure 3.10	Policy Pre-Maturity Form	20
Figure 3.11	Ledger Form	21
Figure 3.12	LifeLink Reports Form	22
Figure 3.13	Change Password Form	22
Figure 3.14	To Do Policy Active Form	23
Figure 3.15	To Do Policy Close Form	23
Figure 4.1	User Evaluation Form 1	38
Figure 4.2	User Evaluation Form 2	38
Figure 4.3	User Evaluation Form 3	39
Figure A.1	System Architecture	47
Figure A.2	SQL Server Import and Export Wizard Form	49
Figure A.3	LifeLink Software's DeskTop Icon	50

Figure B.1	UseCase Diagram for Login Module	51
Figure B.2	UseCase Diagram For Search Module	52
Figure B.3	UseCase Diagram For Policy Maintenance Module	53
Figure B.4	UseCase Diagram For Claim Maintenance Module	54
Figure B.5	UseCase Diagram For Report Generation Module	55
Figure C.1	Login Form	56
Figure C.2	Home Form.....	57
Figure C.3	Popup Menues	58
Figure C.4	Customer Search Form	59
Figure C.5	Society Search Form	60
Figure C.6	Agent Search Form	61
Figure C.7	Policy Search Form	62
Figure C.8	Claim Search Form	63
Figure C.9	Popup Menu Form Collecting Premium Form	64
Figure C.10	Collecting Premium Form	65
Figure C.11	Cash Clear Form	66
Figure C.12	Policy Inquiry Form	67
Figure C.13	Popup Menu For Claim Payments Form	68
Figure C.14	Policy Claim Maintenance Form	69
Figure C.15	Policy Claim Maintenance Form	70
Figure C.16	Policy Claim Maintenance Form	71
Figure C.17	Claim Payment Form	72
Figure C.18	Popup Menu For LifeLink Reports Form	73
Figure C.19	LifeLink Reports Form	74

Figure D.1 Collecting Premium Report	75
Figure D.2 Cash Clear Report.....	76
Figure D.3 Society Wise Daily Report	77
Figure D.4 Society Wise Policy Status Report	78
Figure D.5 Claim Add Report	79
Figure D.6 Claim Approve Report	80
Figure D.7 Claim Payment Report	81
Figure H.1 My Service Letter	91
Figure H.2 Client Certificate	92

List of Tables

Table 5.1	Test Cases For User Login Module	36
Table 5.2	Test Cases For Collecting Premium Module	37
Table A.1	Hardware Requirements For Server Computer	47
Table A.2	Hardware Requirements For Client Computer	48
Table A.3	Software Requirements For Server Computer	48
Table A.4	Software Requirements For Client Computer	48
Table B.1	UseCase Description For Login To The System UseCase	51
Table B.2	UseCase Description For Customer Search UseCase	52
Table B.3	UseCase Description For Collecting Premiums UseCase	53
Table B.4	UseCase Description For Policy Claim Payments UseCase	54
Table B.5	UseCase Description For Generate Report UseCase	55
Table E.1	Test Cases For User Login Module.....	82
Table E.2	Test Cases For Collecting Premium Module.....	82
Table E.3	Test Cases For Policy Claim Maintenance Module.....	83
Table E.4	Test Cases For Ledger Module.....	83
Table E.5	Test Cases For Policy Inquiry Module.....	84
Table E.6	Test Cases For Change Password Module.....	84
Table E.7	Test Cases For End Of Day Process Module.....	84
Table E.8	Test Cases For To Do Policy Active Module.....	85
Table E.9	Test Cases For To Do Policy Close Module.....	85

List of Acronyms

ALMAO - All Lanka Mutual Organization

SQL – Structured Query Language

MS – Microsoft

RUP - Rational Unified Process

UI - User interface

Chapter 1 : Introduction

1.1 Introduction

ALMAO(All Lanka Mutual Organization) is an Insurance Company which is situated in Kegalle. It is a very old Insurance Company which is the mother company of Sanasa Insurance Company Ltd today. ALMAO hasn't a IT Department. I am working Sanasa Insurance Company's IT Department since 2007 (8 years) as a Software Engineer.

1.2 Motivation of the Project

ALMAO has a system for doing day-to-day business but it has a lot of problems. As this reason my boss in IT Department and GM(General Manager) of Sanasa Insurance told me to develop a system to overcome existing system's problems. These are the current system's problems.

* *There wasn't code* –

ALMAO has a system but there wasn't code. As this reason this current system can't be modified. In addition to that we don't know how some functions happen. This is very serious problem ; because company's business processors not fixed ones. Business processors can be changed time to time. So ALMAO's software also has to be modified time to time .

* *Current system uses Ms Access Database as it's Database* –

We think Ms SQL Server Database much better than to Ms Access Database for a backend Database. Ms SQL Server has more functionalities and more popular type of Database for Visual Studio .Net programming language based softwares.

* *Lack of functionalities* –

ALMAO uses current system ; but there aren't sufficient functionalities for every aspect of ALMAO's business. There aren't enough inquiry options , types of reporting, security options , etc...

* *It is not a web based system -*

Current system isn't a web based system . This situation is very difficult to handle this software anywhere from Sri lanka. ALMAO is an insurance company . So it wants to enter data from anywhere . In addition to that Sanasa Insurance's management decided to store data in Sanasa insurance's IT department's servers . Then management can check ALMAO's data from Sanasa Insurance's Head Office.

* *Current system can run only on Windows 98 , Windows 2000 machines -*

This situation is also a very serious problem . Type of Operating System changes time to time . As this reason particular software can be run any Operating System. Machine types are also changing . Now a days Laptops are used very where .Windows 98 , Windows 2000 can't be installed into Laptops .

* *Existing system has some errors –*

ALMAO uses current system ; but it has some errors. So this errors should be corrected with the new proposed system

1.3 Objective and the Scope of the Project

With my proposed system I tried to overcome problems above.

- * After I developed new software ; we have software code also. Then we can modify new system's codes in future ; according to the new business processes of ALMAO.
- * After proposed system has been attached to MsSQL Server ; we can obtain SQL Server's functionalities. MsSQL Server is the database of new system. StoredProcedures which are compiled code that we can call from within SQL statements or from client applications ; have been added to new system . Current system hasn't this feature also.
- * New system has been added new functionalities . Current system hasn't enough functionalities; so new system has been added new reports , new inquiry options , more security features , new administration options and many more.
- * Propose system is going to be a remote login system which means this system can be accessed anywhere from Sri lanka through the private network.

* New system is going to be run on any new Operating System . As this reason no need to fear about technology changing in IT field .

* New system has to be corrected errors which are in the current system .

* Because of new system is an online system all the activities are saving in the Sanasa Insurance's IT department's servers . This situation is very important for the top management to check how is ALMAO's business happening. This situation is also good for auditing purpose.

This new system development task is very difficult task ; because there isn't code of existing system . But I have to develop a software which runs exactly(100% accurately) to current system. If not there will be a serious problem .In addition to that I have to transfer current system's Ms Access Database's data into Ms SQL Server Database . This also a very serious task.

Chapter 2 : Analysis

2.1 Introduction

Analysis should come early in any project. The Analysis phase defines the requirements of the system, independent of how these requirements will be accomplished. This phase defines the problem that the customer is trying to solve.

2.2 Fact Gathering Techniques

The most important part of that analysis is the gathering of business requirements. For this purpose can be used methods below .

2.2.1 Observations

In this technique, system analyst participates in the organization, studies the flow of documents, applies the existing system, and interacts with the users. Observation can be a useful technique when the system analyst have user point of view. Sampling technique called work sampling is useful for observation. By using this technique, system analyst can know how employees spend their days. [9]

2.2.2 Interviews

Interview is the most commonly used technique to collect information from the face-to-face interviews. The purpose of interview is to find, verify, clarify facts, motivate end-users involved, identify requirements and gather ideas and opinions. The role of interview includes interviewer who is system analyst and interviewee who are system owner or user. Interviewing technique needs good communication skills for interaction between system analyst and user. [10]

2.2.3 Sampling and Documentation

The best way to analyse the existing system is to collect facts from existing documentation rather than from human sources.

There are various kinds of documents to collect facts from existing documents. These include:

e-mails, customer complaints, suggestion box notes and reports that document the problem area

problem performance reviews, samples of completed manual forms and reports and samples of completed computerized forms and reports

various types of flowcharts and diagrams, program documentation and user training manuals

System analyst uses sampling techniques in order to organize the above documentation. Sampling technique is the process of combing a representative sample of documents, form and records (Bentley, Whitten, 2007). According to these authors there are two commonly used sampling techniques namely randomization and stratification. Randomization is the process of selecting sample data randomly. Stratification is the systematic process to deduct the variance of sampling data. We can have better understanding of the system due to the analysis of existing documents, forms, files related to the current system. [11]

2.2.4 Prototyping

Another fact-finding technique is known as prototyping which collects the requirement facts of the system. Prototyping is sampling a small working model and it is more related to pre-design of the information system. The implementation of prototyping can be developed in earlier stage of system development life cycle when analyzing the facts. The process of prototyping facts in order to specify the user's requirements is also known as discovery prototyping. [12]

2.3 Current System and the Process

Figure 2.1 shows the UseCase Diagram of the current system.

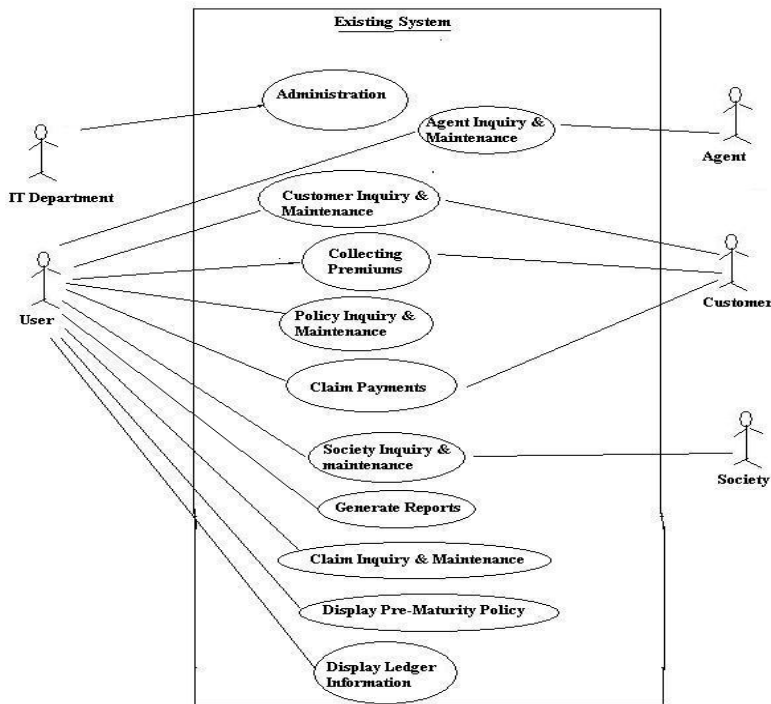


Figure 2.1 High level Use Case diagram

2.4 Functional Requirements

A **Functional Requirement** defines a function of a system or its component. A function is described as a set of inputs, the behavior, and outputs. [1]

Following are the Functional Requirements that are identified during the Analysis stage.

2.4.1 Collecting Premiums

2.4.2 Administration

2.4.3 Claim Payments

2.4.4 Customer Inquiry And Maintenance

2.4.5 Agent Inquiry And Maintenance

2.4.6 Society Inquiry And Maintenance

2.4.7 Display Ledger Information

2.4.8 Policy Inquiry And Maintenance

2.4.9 Display Pre-Maturity Policy

2.4.10 Claim Inquiry And Maintenance

2.4.11 Generate Reports

2.5 Non Functional Requirements

A **Non-Functional Requirement** (NFR) is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. [2]

These are the Non-functional requirements of the Proposed System :

2.5.1 Accuracy

Proposed system should be work 100% correct. If not this software no longer important.

2.5.2 Security

Security is the degree of resistance to, or protection from, harm. It applies to any vulnerable or valuable asset, such as a person, dwelling, community, item, nation, or organization. [13]

2.5.3 Usability

Usability is the ease of use and learnability of a human-made object such as a tool or device.^[1] In software engineering, usability is the degree to which a software can be used by specified consumers to achieve quantified objectives with effectiveness, efficiency, and satisfaction in a quantified context of use. [14]

2.5.4 Interoperability

Interoperability is a characteristic of a product or system, whose interfaces are completely understood, to work with other products or systems, at present or future, in either implementation or access, without any restrictions. [15]

2.5.5 Performance

Computer performance is the amount of work accomplished by a computer system. Depending on the context, high computer performance may involve one or more of the following: [16]

2.5.6 Availability

Availability of a system is typically measured as a factor of its reliability – as reliability increases, so does availability. [17]

2.5.7 Reliability

Reliability is theoretically defined as the probability of success as the frequency of failures; or in terms of availability, as a probability derived from reliability, testability and maintainability. [18]

2.5.8 Capacity

Proposed system shouldn't allocate a lot of disk storage.

2.6 Existing Similar Systems

When I was finding Insurance type softwares in the internet ; I could find several softwares from the internet. SIS.Net 10 is one of them.

2.6.1 SIS.Net 10

“SIS.Net is the result of over four decades of experience in the insurance and technology fields. It is a **full web platform** that allows you to support all the insurance company operations, allowing you to replace your legacy systems and transform your business.

SIS.Net **automatically generates your accounting operations entries** at coverage and dual currency levels, generating their **financial statements in real time** and with maximum detail. This lets you easily meet regulatory obligations and provides the necessary financial visibility for taking real-time strategic decisions.

SIS.Net lets you **design, control and optimize company workflows** providing operational flexibility and unprecedented visibility.

SIS.Net manages all life and non-life business lines, both individually and collectively with a workshop of products for the **design and management of product’s life cycle**. It allows the insurance company to adapt quickly to regulatory and strategic changes and launch a product to the market, without the need of an IT department.

SIS.Net offers facilities for the **mass management** of certificates and micro insurance, increasing the volume of its portfolio by reducing its operating expenses.

SIS.Net is programmed on a service layer, exposing all the functionality through API web services, allowing you to easily use and orchestrate them from other systems by implementing a **SOA architecture**”[37]

Figure 2.2 shows screenshot of SIS.Net 10 software.

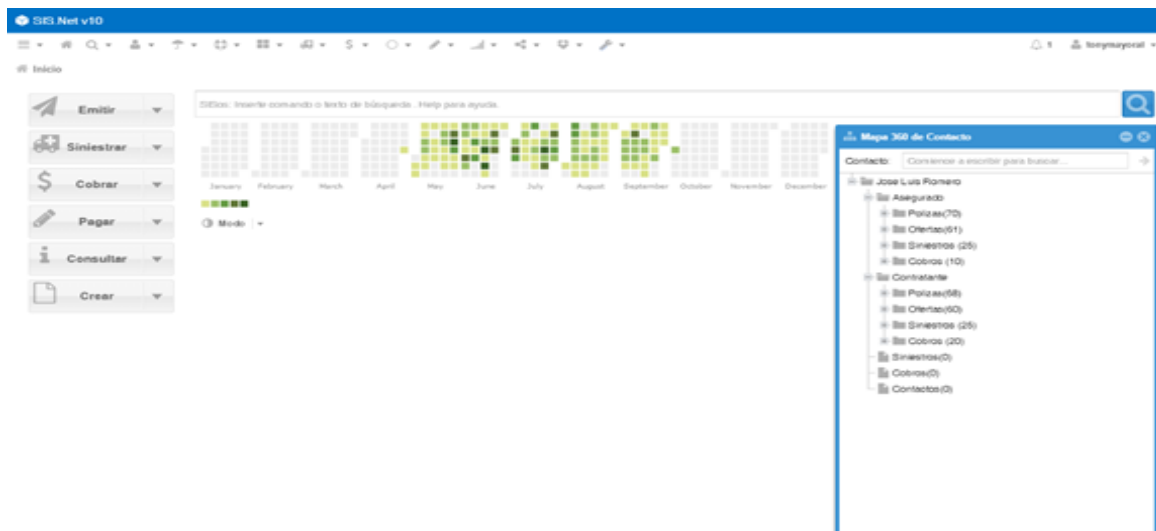


Figure 2.2 SIS.Net 10

But very difficult to find these Insurance type softwares from the internet as a free version. In addition to that ;ALMAO's business processes and databases aren't common to other Insurance Company's business processes and databases. So that the best option is to make bespoke software to match with the ALMAO's existing system.

2.7 Methodology for the Proposed System

The Rational Unified Process(RUP) has been selected as the process model.

The Rational Unified Process (RUP) is an iterative software development process framework created by the Rational Software Corporation, a division of IBM since 2003.^[1] RUP is not a single concrete prescriptive process, but rather an adaptable process framework, intended to be tailored by the development organizations and software project teams that will select the elements of the process that are appropriate for their needs. RUP is a specific implementation of the Unified Process. [6]

The main phases of the RUP are Inception , Elaboration , Construction and Transition .

* Inception –

This phase is oriented to achieve agreement between customer and team on requirements, the expectation for the functionality, and the procedural approach that is to be taken to implement the project.

Basic activities for this phase are:

- **Express clearly project scope:** to capture context, as well requirements, constraints and key features for acceptance criterias.
- **Plan and prepare business case:** to assess alternatives for risk management, team organization and project plan.
- **Possible draft architecture:** draft architecture through some PoC development.
- **Prepare environment:** assess project and organizations, select tools and which parts should be improved. [19]

* Elaboration –

Aim for this phase is to construct an architectural baseline that implements a working application with limited functionality, and to formulate an agreement with customer to further pursue the project.

Basic activities for elaboration are:

- **Create baseline architecture:** create an *executable* architecture
- **Refine vision**
- **Create detailed iteration plans and baselines for construction**
- **Refine use case and prepare construction phase:** at the end of the phase 80% of use case descriptions should be complete. [20]

* Construction –

This phase is oriented to supporting documents, software product, test suite and user manual. Objective here is to understand all requirements and finish development based on baseline architecture.

Basic activities for construction are:

- Manage resources, control and process optimization.
- Component development and acceptance criteria test development.
- Product release assessment based on acceptance criteria. [21]

* Transition -

Transition starts after initial product testing has been performed and the system has been finally deployed. Its objective is to assure software is available to end user.

Basic activities for transition are:

- Execute implanting plans.
- Finish support material.
- Test released product in development environment.
- Create product release.
- Get user feedback.
- Adjust product based on user feedback.
- Make software available to end user. [22]

Chapter 3: Design of Solution

3.1 Introduction

System design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. Systems design could be seen as the application of systems theory to product development. [3]

3.2 Alternative Solutions to the System

When mentioning an alternative solution to this system it can divide into few different sub topics.

***Using mobile phones and laptops altogether with desktop computers :-**

Existing system run only on desktop computers. But this can be expanded upto mobile phones and laptops. Desktop LifeLink software's icon can be arranged easily; but arranging that icon on mobile phone's screen much difficult.

***Purchase commercial software for ALMAO :-**

This can be done by the top management of Sanasa Insurance; but it will want huge amount of money.

*** Using desktop computers only:-**

In this stage ; developing new software for computers is easier than developing new software for mobile phones also. In addition to that purchasing a commercial software is very costly option. As this reasons this solution is selected.

3.3 Object Oriented Designing

Object-oriented design is the process of planning a system of interacting objects for the purpose of solving a software problem. It is one approach to software design. [5] . Figure 3.1 shows the Usecase diagram of Object-oriented design of current system.

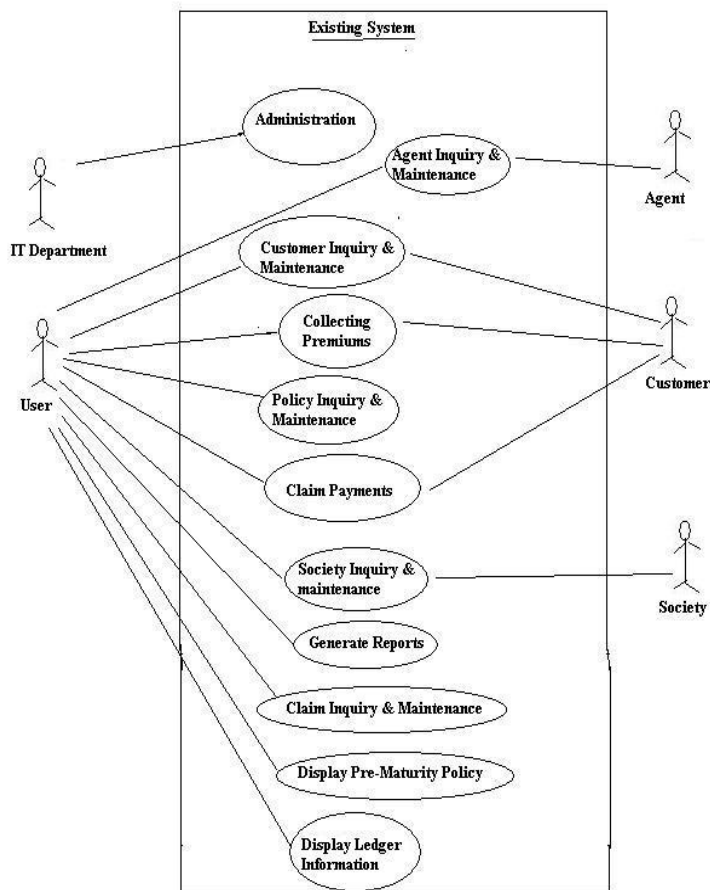


Figure 3.1 High level Use Case diagram

3.4 Database Design

Database design is the process of producing a detailed data model of database. This data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database. A fully attributed data model contains detailed attributes for each entity. [4]

Database normalization, or simply normalization, is the process of organizing the columns (attributes) and tables (relations) of a relational database to reduce data redundancy and improve data integrity. Normalization is also the process of simplifying the design of a database so that it achieves the optimal structure [7]. Figure 3.2 represent the table structure of the database for proposed system is given below.

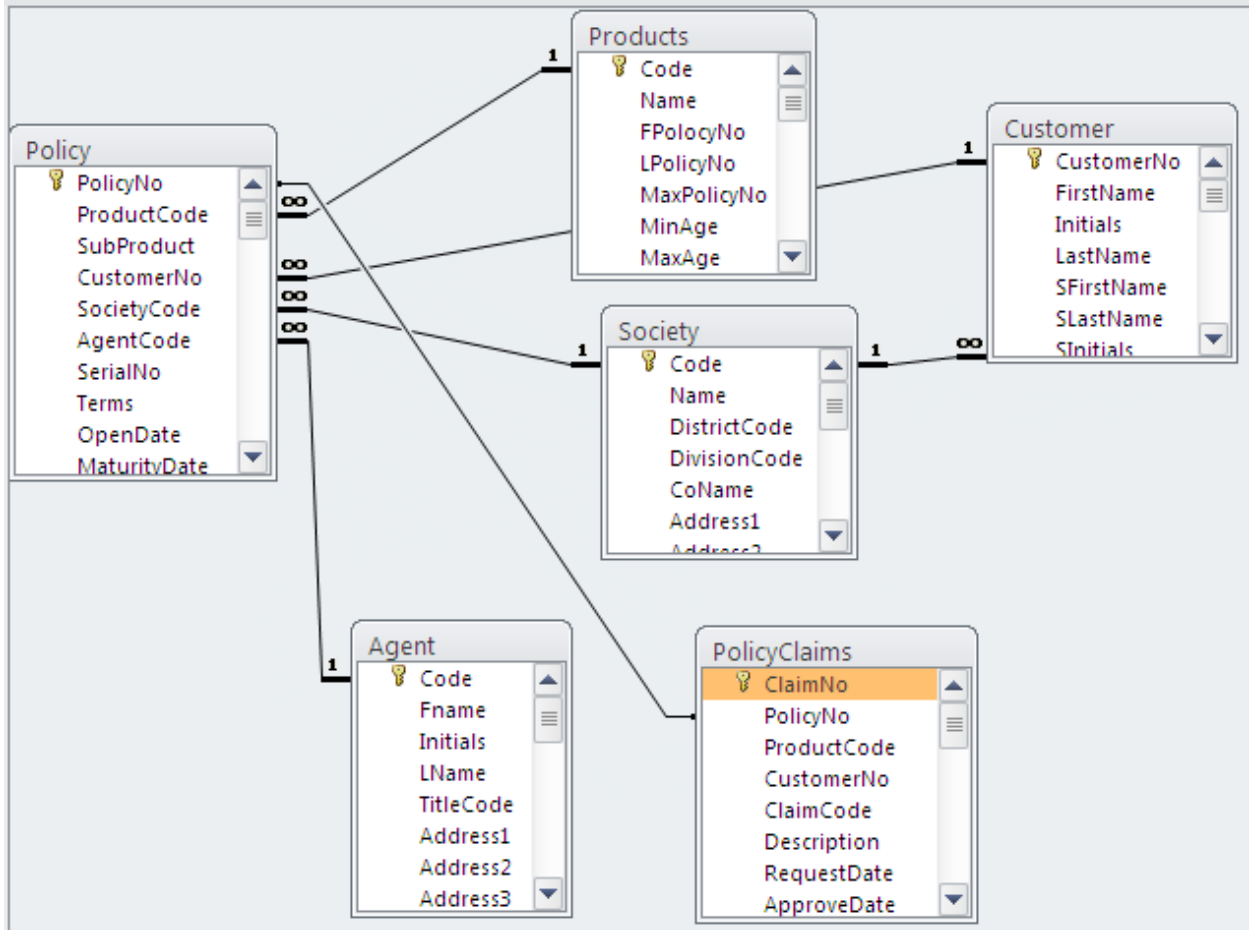


Figure 3.2 Database Diagram

3.5 User Interface Design

User interface design (UI) or user interface engineering is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing usability and the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user-centered design). [8]

Figure 3.3 shows the Login screen of LifeLink software.

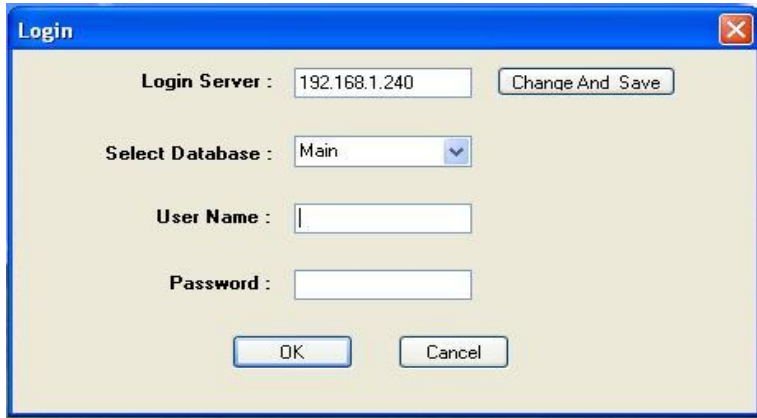


Figure 3.3 Login Form

Figure 3.4 shows the Customer Search window. This can be used to find customer details.

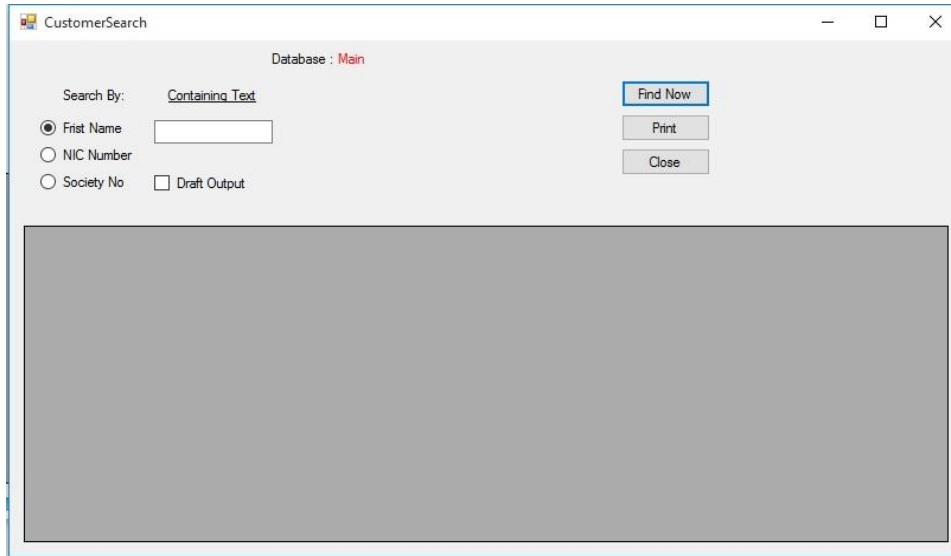


Figure 3.4 Customer Search Form

Figure 3.5 shows the Collecting Premium window. This is the first step of the ALMAO business process. This can be used to collect premiums. There is some calculations also.

Front Office Transactions - CollectingPremium Database : Main

Policy Number Customer Number

Client's Name and Address

Frequency Policy Type

Payment Mode Details/Cheque #

Nos Premiums 0 Amount Received 0 Label18

Appropriated As Follows

Unappropriated Balance		Premium Details	
B/F	<input type="text"/> 0.00	(MM/DD/YY)	
C/F	<input type="text"/> 0	FROM	<input type="text"/> <input type="text"/>
		TO	<input type="text"/> <input type="text"/> 0

Done Re-Print Cancele

Type the Policy Number and Press <Enter> Key

Close

Figure 3.5 Collecting Premium Form

Figure 3.6 shows the Policy Inquiry window. This window can be used to find policy information in detail.

The screenshot shows a software window titled "Policy Inquiry" with a standard Windows interface (minimize, maximize, close buttons). The window displays a search form and a detailed view of policy information. At the top, it indicates the database is "Main".

Search Fields:

- Policy Number:
- Customer Number:
- Three additional empty text input fields arranged in two rows.

Policy Information Tab:

Policy Information	Beneficiary & Payment Information
Date Of Birth <small>(MM/DD/YY)</small>	Society Name <input type="text"/>
Monthly Premium <input type="text"/>	Agency Code <input type="text"/>
Premium Frequency <input type="text"/>	Open Date <input type="text"/> <small>(MM/DD/YY)</small>
Premium <input type="text"/>	Maturity Date <input type="text"/> <small>(MM/DD/YY)</small>
Terms in years <input type="text"/>	Basic Cover <input type="text"/>
Exp. Claim Rate <input type="text"/>	Maturity Value <input type="text"/>
Exp. Claim Days <input type="text"/>	Minor Dissable <input type="text"/>
Accident Cover <input type="text"/>	Partial Dissable <input type="text"/>
Double Acc. Cover <input type="text"/>	Full Dissable <input type="text"/>

Buttons and Instructions:

- Search
- If Error
- Type the Policy Number and Press <Search> Button
- Close

Figure 3.6 Policy Inquiry Form

Figure 3.7 represent the Customer Maintenance which can be used to find customer information in detail. In addition to that; this window can be used to add a customer or modify a customer or delete a customer .

The screenshot shows a software window titled "Customer Maintenance" with a blue title bar and standard window controls. The main area has a light beige background. At the top, it says "Database: Main". The form contains the following fields:

- Customer Number: text input
- Title: dropdown menu
- First Name: text input
- Surname: text input
- Initials: text input
- Address: three stacked text input fields
- Post Code: text input
- NIC No: text input
- Sex: dropdown menu
- Date Of Birth: text input
- Profession: text input
- Civil Status: dropdown menu
- Telephone: text input
- Fax / Office: text input
- Sector: dropdown menu
- District: dropdown menu
- Division: dropdown menu
- Society: dropdown menu
- Spouse / Partner: text input
- Deformaties: large text area

At the bottom, there are three buttons: "Search", "Save", and "Close".

Figure 3.7 Customer Maintenance Form

Figure 3.8 shows the Claim Maintenance window. This window can be used to managing claims like that add a claim, approve a claim and get reports.

The screenshot shows the 'Policy Claim Maintenances' window with the following fields and controls:

- Database:** Main
- Policy Number:** [Text Box]
- Customer Number:** [Text Box]
- Claim Information Entry** (Active Tab):
 - Open Date: [Text Box]
 - Maturity Date: [Text Box]
 - Acc. Balance: [Text Box]
 - Exp. Claim Rate: [Text Box] (Value: 1)
 - Basic Cover: [Text Box]
 - Maturity Value: [Text Box]
 - Exp. Claim Days: [Text Box] (Value: 1)
 - Minor Dissable: [Text Box]
 - Partial Dissable: [Text Box]
 - Full Dissable: [Text Box]
 - Normal Death: [Text Box]
 - Accident Death: [Text Box]
 - PC Death Cover: [Text Box]
 - Beneficiary's Name (1, 2, 3): [Text Boxes]
 - Relationship: [Text Boxes]
 - Claim Classification: [Dropdown] (Value: 001-Road Accidents)
 - Claim Request On: [Text Box] (Value: Tuesday, Marc)
 - Claim Number: [Text Box]
 - Claim Amount: [Text Box] (Value: 0)
- Claim History Of The Policy** (Inactive Tab)
- Claim History Of The Customer** (Inactive Tab)
- Buttons:** Search, Add, Delete, Approve, Re-print, Cancel, Close

Figure 3.8 Claim Maintenance Form

Figure 3.9 represent the Claim Payment Window. This window is used to manage claim payments and get a report.

The screenshot shows a software window titled "Claim Payment" with a blue title bar and standard Windows window controls (minimize, maximize, close). The window content is organized into several sections:

- Database:** Main
- Claim Number:** [Text Input Field]
- Policy No:** [Text Input Field]
- Insured's Information:** A section containing a "Customer No" label followed by four text input fields arranged in two rows of two.
- Claim Information:** A section containing "Claim Type" and "Sub Type" (each with a text input field), and "Requested Date" and "Amount Approved" (each with a text input field).
- Payment Information:** A section containing "Payment Mode" (a dropdown menu currently showing "Cheque"), "Cheque Bank" (a text input field), "Amount Paid" (a text input field), and "Cheque No" (a text input field).

At the bottom of the window, there are three buttons: "Search", "Done", and "Close".

Figure 3.9 Claim Paid Form

Figure 3.10 shows the Policy Pre-Maturity window .This window can be used to managing Pre-maturity policies.

The screenshot displays a software window titled "Policy Pre-Maturity". At the top, it shows "Database : Main" and "Date : 05/26/2012". The window is divided into several sections:

- Policy Number:** A single text input field.
- Insured's Information:** A section containing two rows of four text input fields each.
- Policy Information:** A section with two columns of fields. The left column includes "Account Balance", "Terms (in years)", "Nos Months", and "Nos Months Due". The right column includes "Opened Date", "Maturity Date", "Basic Cover", and "Maturity Value".
- Payment Information:** A section with "Payment Mode" (a dropdown menu), "Cheque Bank" (a text field), "Amount Paid" (a text field with "0" entered), and "Cheque No." (a text field).

At the bottom of the window, there are four buttons: "Search New Policy", "Search Closed Policy", "Save", and "Close".

Figure 3.10 Policy Pre-Maturity Form

Figure 3.11 represent the Ledger Window. This window can be used to find various policy's payment history by year wise.

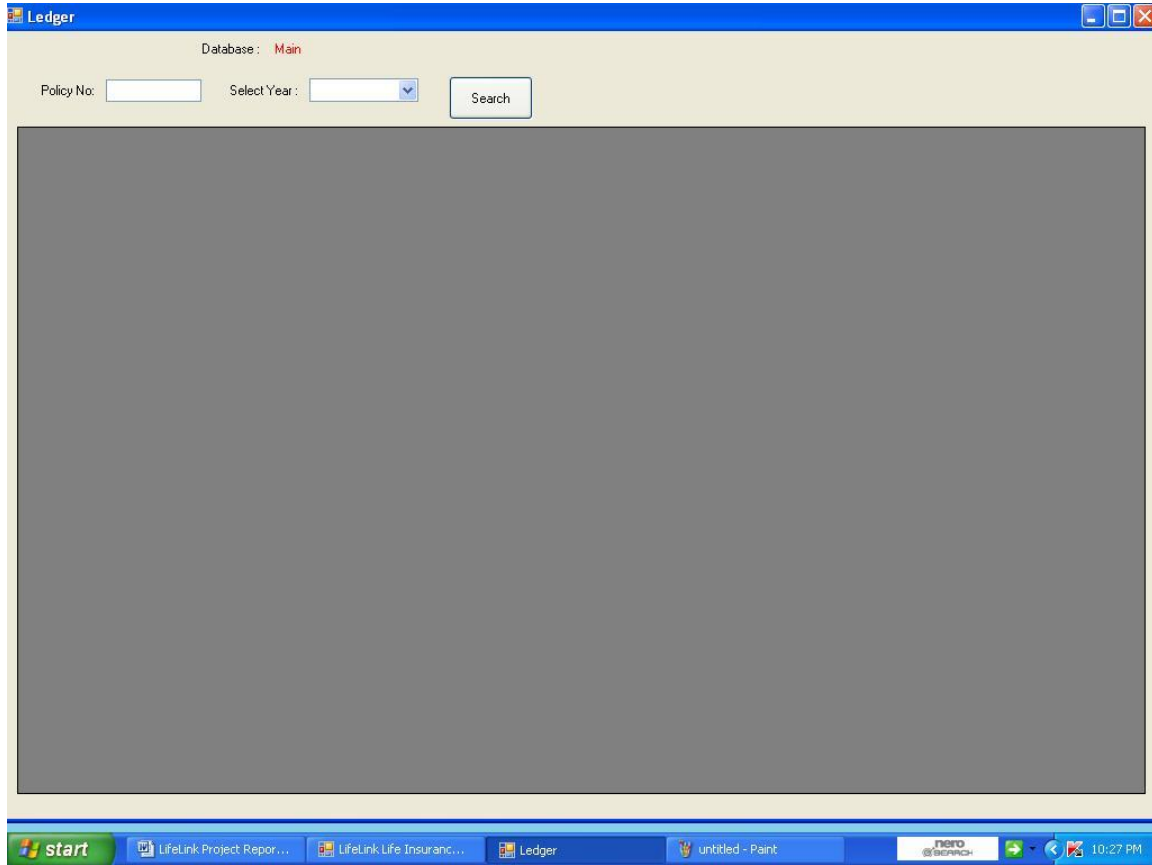


Figure 3.11 Ledger Form

Figure 3.12 shows the Life Link Reports window and this window can be used to get various management reports and user reports.

The screenshot shows a window titled "Life Link Reports". It features a dropdown menu for selecting a report name, a large empty box for the report description, and a "Report Category" section with radio buttons for "Daily Reports", "Monthly Reports", "Ad hoc Reports", "GL Reports", and "All Reports". The "Daily Reports" option is selected. At the bottom, there are "View" and "Close" buttons.

Figure 3.12 LifeLink Reports Form

Figure 3.13 shows the Change Password window and this can be used to change user's password.

The screenshot shows a window titled "Change Password". It has a "Database : Main" label. Below it are three input fields for "User ID", "New Password", and "Verify Password". At the bottom, there are "Save" and "Close" buttons.

Figure 3.13 Change Password Form

Figure 3.14 represent the “To Do Policy Active” window and Figure 3.15 represent the “To Do Policy Close” window. They can be used to change policy status from Closed policy into Active policy or from Active policy into Closed policy.

To Do Policy Active

Database : **Main** **UPDATE status**

Policy No: Customer No:

Name

Address

No.s Premiums Made No.s Premium Due Premium

Open Date Next Due Date

Current Status Type Of Statuses: **0 - Non Approved Policy** ▼

Figure 3.14 To Do Policy Active Form

To Do Policy Close

Database : **Main** **CHANGE POLISY STATUS FROM DATABASE**

Policy No: Customer No:

Name

Address

No.s Premiums Made No.s Premium Due Premium

Open Date Next Due Date Status

Account Balance

Real Status To Change

Figure 3.15 To Do Policy Close Form

Chapter 4: Implementation

4.1 Introduction

The Implementation phase is when the end user of your software is foremost in your mind. During this phase you create the documentation and tools the customer uses to make informed decisions about how to deploy your software securely. To this end, the Implementation phase is when you establish development best practices to detect and remove security and privacy issues early in the development cycle. [23]

4.2 Hardware and Software Requirements

4.2.1 Hardware Requirements For Server Computer

- * Maximum powerful processor
- * 4GB RAM or above memory
- * 250GB Hard Disk or above.

4.2.2 Hardware Requirements For Client Computer

- * Intel Pentium 4 or above processor
- * 512 MB RAM or above memory
- * 10GB Hard Disk or above.
- * LQ 300 Dot Matrix Epson Printer

4.2.3 Software Requirements For Server Computer

- * Windows Workstation OS
- * Ms SQL Server 2008

4.2.4 Software Requirements For Client Computer

- * Windows XP or above OS
- * Microsoft .Net Framework 3.5 or above.
- * Ms Visual Studio.Net 2008
- * Helawadana Pivisuma software

4.3 Development Tools

4.3.1 .NET Framework

The .NET framework is a software development framework from Microsoft. It provides a controlled programming environment where software can be developed, installed and executed on Windows-based operating systems.

The principal design features are:

- Interoperability: This allows for .NET-developed programs to access functionalities in programs developed outside .NET.
- Common Runtime Engine: Also known as the common language runtime, this allows programs developed in .NET to exhibit common behaviors in memory usage, exception handling and security.
- Language Independence: Common language infrastructure specifications (CLI) allow for the exchange of data types between two programs developed in different languages.
- Base Class Library: A library of code for most common functions--used by programmers to avoid repetitive rewriting of code.
- Ease of Deployment: There are tools to ensure the ease of installing programs without interfering with previously installed applications.
- Security: Programs developed in .NET are based on a common security model.
- [24]
-

4.3.2 MsSQL Server 2008

Microsoft SQL Server is a relational database management system developed by Microsoft. As a database server, it is a software product with the primary function of storing and retrieving data as requested by other software applications—which may run either on the same computer or on another computer across a network (including the Internet).

Microsoft markets at least a dozen different editions of Microsoft SQL Server, aimed at different audiences and for workloads ranging from small single-machine applications to large Internet-facing applications with many concurrent users. [25]

4.3.3 Ms Visual Studio.Net 2008

Visual Studio .NET is a Microsoft-integrated development environment (IDE) that can be used for developing consoles, graphical user interfaces (GUIs), Windows Forms, Web services and Web applications.

Visual Studio is used to write native code and managed code supported by Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, .NET Compact Framework and Microsoft Silverlight. Visual Studio .NET's code editor supports IntelliSense and code refactoring, while the Visual Studio .NET integrated debugger supports both source and machine-level debugging. Visual Studio .NET includes other built-in tools, like a form designer, which is useful when building GUI applications; a Web designer that creates dynamic Web pages; a class designer that is used to create custom libraries, and a schema designer for database support. [26]

4.3.4 VB .Net Language

Visual Basic .NET (VB.NET) is a Microsoft object-oriented programming (OOP) language. It evolved from Visual Basic 6 (VB6) to meet an increasing need for easy web-services and web development.

VB.Net was designed to take advantage of the .NET framework-based classes and run-time environment. It was re-engineered by Microsoft as part of its .NET product group. VB.NET supports abstraction, inheritance, and polymorphism.

The most substantial VB6 to VB.NET modification is OOP, which allows for class and object creation and increased code reusability. Many new controls were added to streamline program development. VB.NET also supports multithreading and Web development services, such as Web forms and services. VB.NET's data handling is represented and exchanged via XML-based ADO.NET, which allows for efficient and easy handling of large amounts of data via the Web.

[27]

4.4 Major Code Segments

4.4.1 Database Connection

```
Dim MyConn As SqlConnection
Dim MyCommand As SqlDataAdapter
Dim ds As DataSet
```

```
MyConn = New SqlConnection("Data Source=" & LoginForm1.TextBox1.Text & ";Initial Catalog=" &
Label4.Text & "; User ID=sa; Password=sic1almao;")
```

```
MyCommand = New SqlDataAdapter("Select
CustomerNo,FirstName,Initials,Address1,Address2,Address3,Address4,NIC,SocCode from Customer WHERE
FirstName Like" & TextBox1.Text & "%", MyConn)
```

```
ds = New DataSet
MyCommand.Fill(ds, "Customer")
DataGridView1.DataSource = ds
DataGridView1.DataMember = "Customer"
```

4.4.2 Login Handling And Data Validation

```
Imports System.Data.SqlClient
Imports System.Data
Imports System.Text
```

```
Imports System.Security.Cryptography
Imports System
Imports System.IO
Imports System.Security
Imports System.Runtime.InteropServices
```

```
'*****
```

```
Public Class LoginForm1
```

```
Dim MyConn As SqlConnection
Dim MyCommand1 As SqlCommand
Dim uu As SqlDataReader
```

```
Dim Form1 As New LifeLink.Form1
```

```
'*****
```

```

Private Sub OK_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles OK.Click
    PasswordTextBox.Text = EncryptPassword(PasswordTextBox.Text)

    If (CheckCredentials(UsernameTextBox.Text, PasswordTextBox.Text)) Then

        MyConn = New SqlConnection("Data Source=" & TextBox1.Text & ";Initial Catalog=" & ComboBox1.Text & ";
        User ID=sa; Password=sic1almao; ")

        MyCommand1 = New SqlCommand("UPDATE [User] SET LLogDate=@LLogDate WHERE UserID=" &
        UsernameTextBox.Text & "'", MyConn)

        MyConn.Open()

        MyCommand1.CommandType = CommandType.Text

        Dim param1 As New SqlParameter("@LLogDate", SqlDbType.DateTime)
        param1.Direction = ParameterDirection.Input
        param1.Value = Label11.Text

        MyCommand1.Parameters.Add(param1)

        MyCommand1.ExecuteNonQuery()
        MyConn.Close()

        Me.Hide()
        Form1.Show()

    Else

        MsgBox("      Incorrect UserID or Password      ")

    End If

    MyConn.Close()
End Sub

```

```

'*****

```

```

Function CheckCredentials(ByVal UserID As String, ByVal Pass As String) As Boolean

```

```

    MyConn = New SqlConnection("Data Source=" & TextBox1.Text & ";Initial Catalog=" & ComboBox1.Text & ";
    User ID=sa; Password=sic1almao; ")

```

```

    MyConn.Open()

```

```

    Dim MyCommand As SqlCommand

```

```

    MyCommand = New SqlCommand("SELECT UserID,PWard FROM [User]WHERE
    UserID=@givenUsername AND PWard=@givenPassword", MyConn)

```

```

MyCommand.Parameters.Add(New SqlParameter("@givenUsername", SqlDbType.VarChar))
MyCommand.Parameters("@givenUsername").Value = UserIDD
MyCommand.Parameters.Add(New SqlParameter("@givenPassword", SqlDbType.NVarChar, 10))
MyCommand.Parameters("@givenPassword").Value = Pass

uu = MyCommand.ExecuteReader()
If Not uu.Read() Then
    Return False
Else
    Return True
End If

End Function
'*****

Public Function EncryptPassword(ByVal Password As String) As String
    Dim PassNew As String
    Dim Cntr As Integer
    Cntr = 1
    Do While Cntr <= Len(Password)
        PassNew = CStr(PassNew) & Chr((Asc(Mid(Trim(Password), Cntr, 1)) + 20))
        Cntr = Cntr + 1
    Loop
    EncryptPassword = PassNew

End Function

'*****

End Class

```

4.4.3 Search,Insert,Update data

(a)Search Data Code

```

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

Dim MyConn As SqlConnection
Dim MyCommand As SqlCommand

MyConn = New SqlConnection("Data Source=" & LoginForm1.TextBox1.Text & ";Initial Catalog=" &
Label15.Text & "; User ID=sa; Password=sielalmao;")

MyCommand = New SqlCommand("SELECT PolicyClaims.ClaimNo, PolicyClaims.PolicyNo,
PolicyClaims.CustomerNo, Customer.FirstName, Customer.LastName, Customer.Initials, Customer.Address1,
Customer.Address2, Customer.Address3,

```

```
Customer.Address4,PolicyClaims.ClaimCategory,PolicyClaims.Description,
PolicyClaims.RequestDate,PolicyClaims.AmountClaimed FROM PolicyClaims INNER JOIN Customer ON
PolicyClaims.CustomerNo = Customer.CustomerNo WHERE (PolicyClaims.ClaimNo)=" & TextBox1.Text & "",
MyConn)
```

```
Dim MyReader As SqlDataReader
MyConn.Open()
MyCommand.CommandType = CommandType.Text
MyReader = MyCommand.ExecuteReader
```

```
Do While MyReader.Read
TextBox1.Text = MyReader("ClaimNo")
Loop
```

```
End Sub
```

(b)Insert Data Code

```
Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
```

```
Dim MyConn As SqlConnection
Dim MyCommand As SqlCommand
```

```
MyConn = New SqlConnection("Data Source=" & LoginForm1.TextBox1.Text & ";Initial Catalog=" &
Label15.Text & "; User ID=sa; Password=sic1almao;")
```

```
MyCommand = New SqlCommand("INSERT INTO
TellerGLTran(TranNo,TranDate,TranTime,Terminal,UserID,Status,TranCode,Cash,Cheque,Voucher,MOrder,Sum1
)VALUES(@TranNo,@TranDate,@TranTime,@Terminal,@UserID,@Status,@TranCode,@Cash,@Cheque,@Vou
cher,@MOrder,@Sum1)", MyConn)
```

```
MyConn.Open()
```

```
*****
```

```
MyCommand.CommandType = CommandType.Text
```

```
*****
```

```
Dim param1 As New SqlParameter("@TranNo", SqlDbType.Decimal)
param1.Direction = ParameterDirection.Input
param1.Value = Val(TextBox20.Text)
MyCommand.Parameters.Add(param1)
```

```
Dim param2 As New SqlParameter("@TranDate", SqlDbType.DateTime)
param2.Direction = ParameterDirection.Input
param2.Value = TextBox3.Text
MyCommand.Parameters.Add(param2)
```

```
Dim param3 As New SqlParameter("@TranTime", SqlDbType.DateTime)
param3.Direction = ParameterDirection.Input
Label13.Text = String.Format("{0:T}", DateTime.Now)
param3.Value = Label13.Text
MyCommand.Parameters.Add(param3)
```

```
Dim param4 As New SqlParameter("@Terminal", SqlDbType.Decimal)
param4.Direction = ParameterDirection.Input
param4.Value = "3"
MyCommand.Parameters.Add(param4)
```

```
Dim param5 As New SqlParameter("@UserID", SqlDbType.VarChar)
param5.Direction = ParameterDirection.Input
param5.Value = LoginForm1.Label5.Text
MyCommand.Parameters.Add(param5)
```

```
Dim param6 As New SqlParameter("@Status", SqlDbType.Decimal)
param6.Direction = ParameterDirection.Input
param6.Value = "0"
MyCommand.Parameters.Add(param6)
```

```
Dim param7 As New SqlParameter("@TranCode", SqlDbType.VarChar)
param7.Direction = ParameterDirection.Input
param7.Value = "TCC"
MyCommand.Parameters.Add(param7)
```

```
Dim param8 As New SqlParameter("@Cash", SqlDbType.Money)
param8.Direction = ParameterDirection.Input
param8.Value = TextBox7.Text
MyCommand.Parameters.Add(param8)
```

```
Dim param9 As New SqlParameter("@Cheque", SqlDbType.Money)
param9.Direction = ParameterDirection.Input
param9.Value = TextBox11.Text
MyCommand.Parameters.Add(param9)
```

```
Dim param10 As New SqlParameter("@Voucher", SqlDbType.Money)
param10.Direction = ParameterDirection.Input
param10.Value = TextBox15.Text
MyCommand.Parameters.Add(param10)
```

```
Dim param11 As New SqlParameter("@MOrder", SqlDbType.Money)
param11.Direction = ParameterDirection.Input
param11.Value = TextBox19.Text
MyCommand.Parameters.Add(param11)
```

```
Dim param12 As New SqlParameter("@Sum1", SqlDbType.Money)
param12.Direction = ParameterDirection.Input
param12.Value = TextBox21.Text
MyCommand.Parameters.Add(param12)
```

```
MyCommand.ExecuteNonQuery()
```

(c)Update Data Code

```
Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button2.Click
```

```
Dim MyConn As SqlConnection
```

```
Dim MyCommand As SqlCommand
```

```
MyConn = New SqlConnection("Data Source=" & LoginForm1.TextBox1.Text & ";Initial Catalog=" & Label15.Text & "; User ID=sa; Password=siclalmao;")
```

```
MyCommand = New SqlCommand("UPDATE PolicyClaims SET  
ChequeBank=@ChequeBank,AmountPaid=@AmountPaid,ChequeNo=@ChequeNo,PaidUser=@PaidUser  
WHERE ClaimNo=" & TextBox1.Text & "", MyConn)
```

```
MyConn.Open()
```

```
MyCommand.CommandType = CommandType.Text
```

```
Dim param1 As New SqlParameter("@ChequeBank", SqlDbType.VarChar)  
param1.Direction = ParameterDirection.Input  
param1.Value = TextBox13.Text  
MyCommand.Parameters.Add(param1)
```

```
Dim param2 As New SqlParameter("@AmountPaid", SqlDbType.Money)  
param2.Direction = ParameterDirection.Input  
param2.Value = TextBox14.Text  
MyCommand.Parameters.Add(param2)
```

```
Dim param3 As New SqlParameter("@ChequeNo", SqlDbType.VarChar)  
param3.Direction = ParameterDirection.Input  
param3.Value = TextBox15.Text  
MyCommand.Parameters.Add(param3)
```

```
Dim param4 As New SqlParameter("@PaidDate", SqlDbType.DateTime)  
param4.Direction = ParameterDirection.Input  
Dim myDate As Date = DateTime.Now  
param4.Value = LoginForm1.Label7.Text  
MyCommand.Parameters.Add(param4)
```

```
Dim param5 As New SqlParameter("@PaidUser", SqlDbType.VarChar)  
param5.Direction = ParameterDirection.Input  
param5.Value = LoginForm1.Label5.Text  
MyCommand.Parameters.Add(param5)
```

```
MyCommand.ExecuteNonQuery()
```

```
MyConn.Close()
```

```
End Sub
```

4.4.4 Reporting

`Imports System.Data.SqlClient`

`Imports System.Data`

`Imports CrystalDecisions.CrystalReports.Engine`

`Imports CrystalDecisions.Shared`

`Imports System.Configuration`

`Imports System.Text`

`Public Class Receipt`

`Private Sub Receipt_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load`

`Dim myConn As SqlConnection`

`Dim strSql As New StringBuilder`

`Dim oTable As New DataTable()`

`Dim crReportDocument As New ReportDocument`

`Label2.Text = LoginForm1.ComboBox1.Text`

`myConn = New SqlConnection("Data Source=" & LoginForm1.TextBox1.Text & ";Initial Catalog=" & Label2.Text & "; User ID=sa; Password=siclalmao;")`

`Dim strReportPath As String = Application.StartupPath & "\" & "CrystalReport5" & ".rpt"`

`crReportDocument.Load(strReportPath)`

`myConn.Open()`

`strSql.Append("SELECT Policy.PolicyNo,Policy.Premium,Policy.NextDueDate,Temp.UserID, Temp.Terminal, Temp.SocietyCode, Temp.TranAmount, Customer.SFirstName, Customer.SLastName, Customer.SInitials, Customer.SAddress1, Customer.SAddress2, Customer.SAddress3, Customer.SAddress4, Temp.FromDate, Temp.No1, Temp.No2, Temp.CF, Temp.Total, Temp.Deduct, Temp.BF, Temp.NoOfAreasPremium, Temp.NoOfPremiums, Temp.TranNo, Temp.TranDate, Temp.TranTime, Temp.AgencyCode FROM Customer INNER JOIN (Policy INNER JOIN Temp ON Policy.LastTranNo = Temp.TranNo) ON Customer.CustomerNo = Policy.CustomerNo WHERE Policy.PolicyNo=" & TextBox1.Text & """)`

`Dim oDA As New SqlDataAdapter(strSql.ToString, myConn)`

`oDA.Fill(oTable)`

`crReportDocument.SetDataSource(oTable)`

`CrystalReportViewer1.ReportSource = crReportDocument`

`End Sub`

4.4.5 Re-usable Components

I have written various softwares as a Software Engineer from long time. So I have collected a lot of code storage . I used these codes for this software development project also . In addition to that I got some open source codes from the internet and I acknowledge for the internet .

Chapter 5: Evaluation

5.1 Introduction

Evaluation is a process that critically examines a program. It involves collecting and analyzing information about a program's activities, characteristics, and outcomes. Its purpose is to make judgments about a program, to improve its effectiveness, and/or to inform programming decisions (Patton, 1987). [28]

5.2 Software Testing

Software testing is a process of executing a program or application with the intent of finding the software bugs. [29]

5.3 Types of Testing

5.3.1 Unit Testing

Unit 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 include the constructors and destructors.^[27]

These types of tests are usually written by developers as they work on code (white-box style), to ensure that the specific function is working as expected. One function might have multiple tests, to catch corner cases or other branches in the code. Unit testing alone cannot verify the functionality of a piece of software, but rather is used to ensure that the building blocks of the software work independently from each other.

Unit testing is a software development process that involves synchronized application of a broad spectrum of defect prevention and detection strategies in order to reduce software development risks, time, and costs. It is performed by the software developer or engineer during the construction phase of the software development lifecycle. Rather than replace traditional QA focuses, it augments it. Unit testing aims to eliminate construction errors before code is promoted to QA; this strategy is intended to increase the quality of the resulting software as well as the efficiency of the overall development and QA process.

Depending on the organization's expectations for software development, unit testing might include static code analysis, data-flow analysis, metrics analysis, peer code reviews, code coverage analysis and other software verification practices. [32]

5.3.2 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.

Integration testing works to expose defects in the interfaces and interaction between integrated components (modules). Progressively larger groups of tested software components corresponding to elements of the architectural design are integrated and tested until the software works as a system. [33]

5.3.3 System Testing

System testing tests a completely integrated system to verify that the system meets its requirements. [31] 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. [34]

5.3.4 Acceptance Testing

Operational acceptance is used to conduct operational readiness (pre-release) of a product, service or system as part of a quality management system. OAT is a common type of non-functional software testing, used mainly in software development and software maintenance projects. This type of testing focuses on the operational readiness of the system to be supported, or to become part of the production environment. Hence, it is also known as operational readiness testing (ORT) or Operations readiness and assurance (OR&A) testing. Functional testing within OAT is limited to those tests which are required to verify the *non-functional* aspects of the system [35]

5.3.5 Regression Testing

Regression testing is a type of software testing which verifies that software which was previously developed and tested still performs the same way after it was changed or interfaced with other software. Changes may include software enhancements, patches, configuration changes, etc. During regression testing, new software bugs or *regressions* may be uncovered. Sometimes a software change impact analysis is performed to determine what areas could be affected by the proposed changes. These areas may include functional and non-functional areas of the system.

The purpose of regression testing is to ensure that changes such as those mentioned above have not introduced new faults.^[1] One of the main reasons for regression testing is to determine whether a change in one part of the software affects other parts of the software. [36]

5.4 System Test Plan

A Software Test Plan is a document describing the testing scope and activities. It is the basis for formally testing any software/product in a project. [31]

5.5 System Test Cases And Test Results

A test case is a document which consists of a set of conditions or actions which are performed on the software application in order to verify the expected functionality of the feature. Here we describe the end to end logical flow of a specific requirement with test data, prerequisites and expected results. [30]

5.5.1 Test Cases For User Login Module

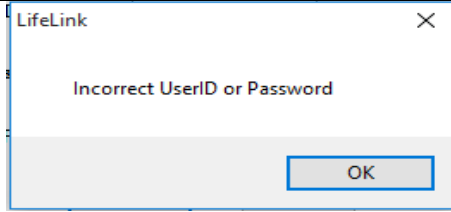
Test No	Test Case	Expected Output	Actual Output	Status
1	When incorrect UserName or Password is entered	Display a message called "Incorrect User ID or Password"	 A screenshot of a dialog box titled "LifeLink" with a close button (X) in the top right corner. The dialog box contains the text "Incorrect UserID or Password" and an "OK" button at the bottom right.	Pass

Table 5.1 Test Cases For User Login Module

5.5.2 Test Cases For Collecting Premium Module

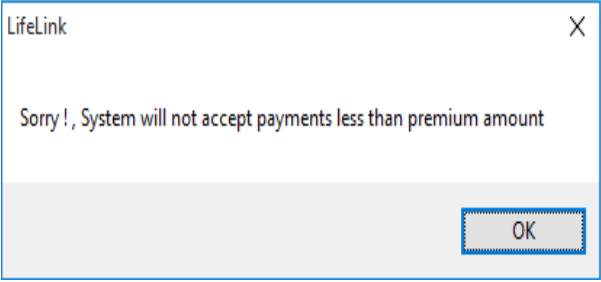
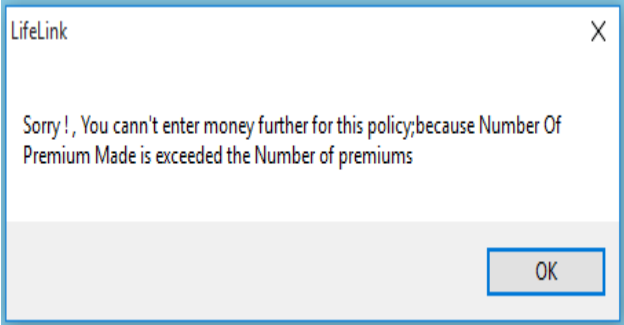
Test No	Test Case	Expected Output	Actual Output	Status
1	When entering a payment which is less than or equal to premium amount.	Display a message called "Sorry ! , System will not accept payments less than premium amount"		Pass
2		Display a message called "Sorry ! , You can't enter money further for this policy;because Number Of Premium Made is exceeded the Number of premiums"		Pass

Table 5.2 Test Cases For Collecting Premium Module

Remaining test cases and test results are given in Appendix E.

5.6 User Evaluation

User evaluation is done by selecting different users of the system. Miss Deepa ,Mrs Chandralatha and Miss Chandrika are three clerks and they work different sections of ALMAO office. Figure 4.1 and Figure 4.2 and Figure 4.3 show User Evaluation Questionires of each clerks.

User Evaluation Questionire

Name of user : <i>Deepa chandani</i>		Role of user : <i>clerk</i>			
Evaluating Item	Very Good	Good	Average	Poor	Very Poor
Overall reaction		✓			
Character readability	✓				
Color scheme		✓			
System navigation	✓				
Ease of usage	✓				
Functionalities		✓			
Interfaces	✓				
Ease of learning	✓				
Response time		✓			
Comments:	—				

Figure 4.1 User Evaluation Form 1

User Evaluation Questionire

Name of user : <i>chondralatha</i>		Role of user : <i>clerk</i>			
Evaluating Item	Very Good	Good	Average	Poor	Very Poor
Overall reaction		✓			
Character readability	✓				
Color scheme	✓				
System navigation	✓				
Ease of usage	✓				
Functionalities		✓			
Interfaces		✓			
Ease of learning	✓				
Response time		✓			
Comments:	—				

Figure 4.2 User Evaluation Form 2

User Evaluation Questionire

Name of user : Kumari gurusingh		Role of user : clerk			
Evaluating Item	Very Good	Good	Average	Poor	Very Poor
Overall reaction		✓			
Character readability		✓			
Color scheme		✓			
System navigation	✓				
Ease of usage	✓				
Functionalities		✓			
Interfaces	✓				
Ease of learning	✓				
Response time		✓			
Comments:	—				

Figure 4.3 User Evaluation Form 3

Chapter 6: Conclusion

6.1 Introduction

This LifeLink software is developed for particular weaknesses of existing system. They are ;

- * *There wasn't code*
- * *Current system uses Ms Access Database as it's Database*
- * *Lack of functionalities*
- * *It is not a web based system*
- * *Current system can run only on Windows 98 , Windows 2000 machines*
- * *Existing system has some errors*

This LifeLink Software project is completed on time with necessary requirements of ALMAO and top management of Sanasa Insurance Company LTD. They are happy with my task.

6.2 Future Enhancement

This new LifeLink software is developed as an urgent project; because of faults of existing system. But this new software can be modified time to time according to new technology.

Following are the some of the future enhancement of the system.

- This software can be converted into web based software.
- Can be added more reports.
- Can be added more analytical tools to find customer's information.
- Can be added advanced administration options to this software.
- For this software ; recommended printer is called Epson LQ 300 Dot matrix printer. But it should be changed to any printer for this software.

6.3 Lesson Learnt

This project development task is gave me a lot of experience about “How to develop a software by own” . These are the some of lesson learnt ;

- How to develop some codes.
- How to interact LifeLink software with network systems of the company.
- How to handle peoples for my purpose.
- Increased my patient capability.
- How to do a work with time frame factors.
- I improved my database handling capabilities.
- I studied how to deal with managers.
- Main thing is ; I learnt how to get a responsibilities for a serious task.

References

- [1]"Functional requirement", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Functional_requirement. [Accessed: 21- Dec- 2017].
- [2]"Non-functional requirement", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Non-functional_requirement. [Accessed: 21- Dec- 2017].
- [3]"Systems design", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Systems_design. [Accessed: 21- Dec- 2017].
- [4]"Database design", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Database_design. [Accessed: 21- Dec- 2017].
- [5]"Object-oriented design", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Object-oriented_design. [Accessed: 23- Dec- 2017].
- [6]"Rational Unified Process", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Rational_Unified_Process. [Accessed: 23- Dec- 2017].
- [7]"Database normalization", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Database_normalization. [Accessed: 23- Dec- 2017].
- [8]"User interface design", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/User_interface_design. [Accessed: 23- Dec- 2017].
- [9]"Definition Of Fact Finding Techniques Information Technology Essay", *UKEssays*, 2017. [Online]. Available: <https://www.ukessays.com/essays/information-technology/definition-of-fact-finding-techniques-information-technology-essay.php>. [Accessed: 23- Dec- 2017].
- [10]"Definition Of Fact Finding Techniques Information Technology Essay", *UKEssays*, 2017. [Online]. Available: <https://www.ukessays.com/essays/information-technology/definition-of-fact-finding-techniques-information-technology-essay.php>. [Accessed: 23- Dec- 2017].)

- [11]"Definition Of Fact Finding Techniques Information Technology Essay", *UKEssays*, 2017. [Online]. Available: <https://www.ukessays.com/essays/information-technology/definition-of-fact-finding-techniques-information-technology-essay.php>. [Accessed: 23- Dec- 2017].
- [12]"Definition Of Fact Finding Techniques Information Technology Essay", *UKEssays*, 2017. [Online]. Available: <https://www.ukessays.com/essays/information-technology/definition-of-fact-finding-techniques-information-technology-essay.php>. [Accessed: 23- Dec- 2017].
- [13]"Security", *En.wikipedia.org*, 2017. [Online]. Available: <https://en.wikipedia.org/wiki/Security>. [Accessed: 23- Dec- 2017].
- [14]"Usability", *En.wikipedia.org*, 2017. [Online]. Available: <https://en.wikipedia.org/wiki/Usability>. [Accessed: 23- Dec- 2017].
- [15]"Interoperability", *En.wikipedia.org*, 2017. [Online]. Available: <https://en.wikipedia.org/wiki/Interoperability>. [Accessed: 23- Dec- 2017].
- [16]"Computer performance", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Computer_performance. [Accessed: 23- Dec- 2017].
- [17]"Availability", *En.wikipedia.org*, 2017. [Online]. Available: <https://en.wikipedia.org/wiki/Availability>. [Accessed: 23- Dec- 2017]
- [18]"Reliability engineering", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Reliability_engineering. [Accessed: 23- Dec- 2017].

[19]"RUP - IBM Rational Unified Process/Phases - Wikibooks, open books for an open world", *En.wikibooks.org*, 2017. [Online]. Available: https://en.wikibooks.org/wiki/RUP_-_IBM_Rational_Unified_Process/Phases. [Accessed: 23- Dec- 2017].

[20]"RUP - IBM Rational Unified Process/Phases - Wikibooks, open books for an open world", *En.wikibooks.org*, 2017. [Online]. Available: https://en.wikibooks.org/wiki/RUP_-_IBM_Rational_Unified_Process/Phases. [Accessed: 23- Dec- 2017].

[21]"RUP - IBM Rational Unified Process/Phases - Wikibooks, open books for an open world", *En.wikibooks.org*, 2017. [Online]. Available: https://en.wikibooks.org/wiki/RUP_-_IBM_Rational_Unified_Process/Phases. [Accessed: 23- Dec- 2017].

[22]"RUP - IBM Rational Unified Process/Phases - Wikibooks, open books for an open world", *En.wikibooks.org*, 2017. [Online]. Available: https://en.wikibooks.org/wiki/RUP_-_IBM_Rational_Unified_Process/Phases. [Accessed: 23- Dec- 2017].

[23]"Phase 3: Implementation", *Msdn.microsoft.com*, 2017. [Online]. Available: <https://msdn.microsoft.com/en-us/library/windows/desktop/cc307416.aspx>. [Accessed: 23- Dec- 2017].

[24]"What is the .NET Framework (.NET)? - Definition from Techopedia", *Techopedia.com*, 2017. [Online]. Available: <https://www.techopedia.com/definition/3734/net-framework-net>. [Accessed: 23- Dec- 2017].

[25]"Microsoft SQL Server", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Microsoft_SQL_Server. [Accessed: 23- Dec- 2017].

[26]"What is Visual Studio .NET? - Definition from Techopedia", *Techopedia.com*, 2017. [Online]. Available: <https://www.techopedia.com/definition/15740/visual-studio-net>. [Accessed: 23- Dec- 2017]

[27]"What is Visual Basic .NET (VB.NET)? - Definition from Techopedia", *Techopedia.com*, 2017. [Online]. Available: <https://www.techopedia.com/definition/16458/visual-basic-net-vb.net>. [Accessed: 23- Dec- 2017].

[28]"Evaluation: What is it and why do it? | Meera", *Meera.snre.umich.edu*, 2017. [Online]. Available: <http://meera.snre.umich.edu/evaluation-what-it-and-why-do-it>. [Accessed: 23- Dec- 2017].

[29]I. Level, A. Tutorial, 2. Dates, I. Tests, C. Us, P. Policy, T. Use, A. Us, W. us, j. maurya, v. r, H. talla, M. DASARADHA, I. Guide, I. Guide and S. donald, "What is Software Testing?", *Istqbexamcertification.com*, 2017. [Online]. Available: <http://istqbexamcertification.com/what-is-software-testing>. [Accessed: 23- Dec- 2017].

[30]I. Level, A. Tutorial, 2. Dates, I. Tests, C. Us, P. Policy, T. Use, A. Us and W. us, "ISTQB Exam Certification – Study material for Foundation level, Premium & Free for ISTQB and ASTQB Exam, Certification questions, answers, software testing tutorials and more", *Istqbexamcertification.com*, 2017. [Online]. Available: <http://istqbexamcertification.com>. [Accessed: 23- Dec- 2017].

[31]"Test Plan - Software Testing Fundamentals", *Software Testing Fundamentals*, 2017. [Online]. Available: <http://softwaretestingfundamentals.com/test-plan>. [Accessed: 23- Dec- 2017].

[32]"Software testing", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Software_testing. [Accessed: 23- Dec- 2017].

[33]"Software testing", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Software_testing. [Accessed: 23- Dec- 2017].

[34]"Software testing", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Software_testing. [Accessed: 23- Dec- 2017].

[35]"Software testing", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Software_testing. [Accessed: 23- Dec- 2017].

[36]"Regression testing", *En.wikipedia.org*, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Regression_testing. [Accessed: 23- Dec- 2017].

[37]"Insurance Carriers – Axxis Systems", *En.axxis-systems.com*, 2018. [Online]. Available: <http://en.axxis-systems.com/insurers/>. [Accessed: 03- Jan- 2018].

Appendix A – System Documentation

Figure A.1 shows the network diagram of proposed system.

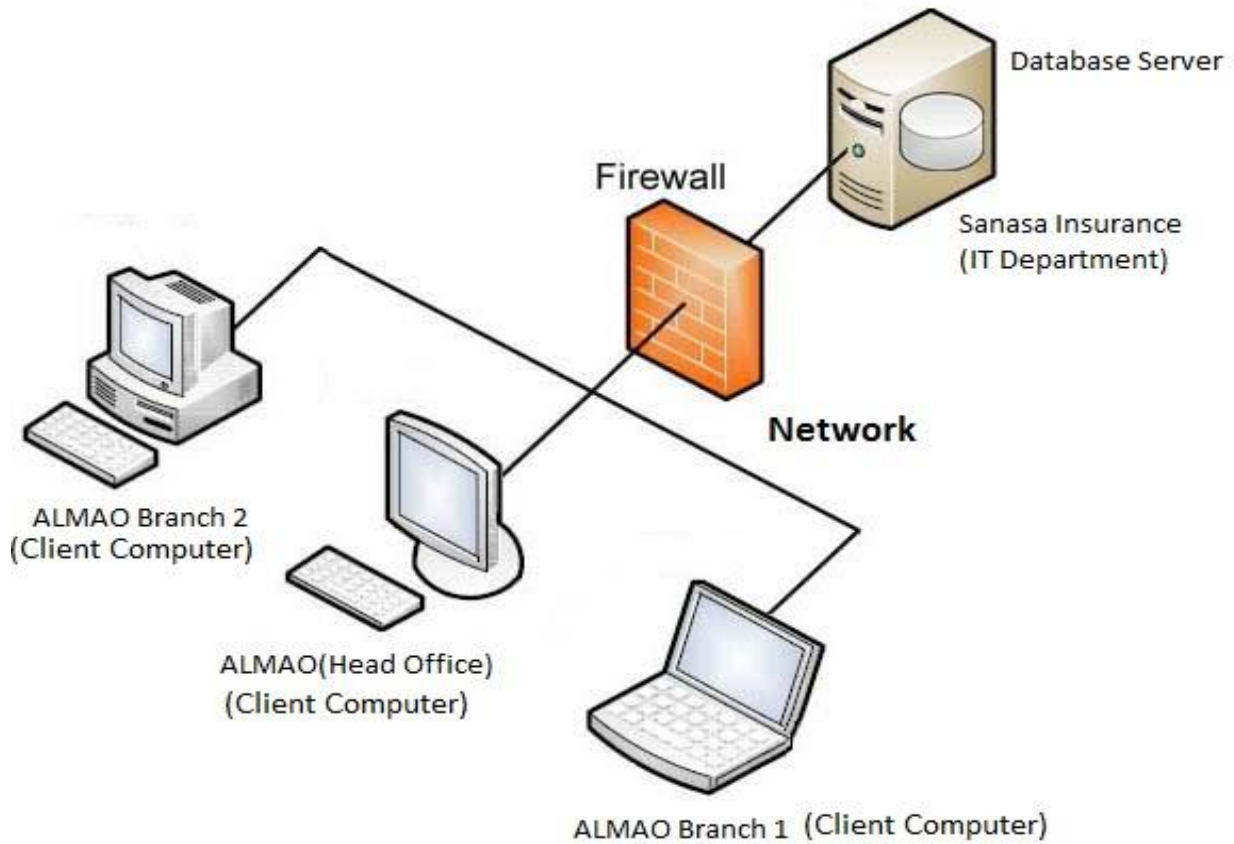


Figure A.1 System Architecture

Step 01

Hardware Requirements For Server Computer

Hardware	Minimum Requirements
Processor	3GHz or above
Memory	4GB or above
Hard Disk	250GB or above
Display	Any Monitor
Network	Wired or wireless network

Table A.1 Hardware Requirements For Server Computer

Hardware Requirements For Client Computer

Hardware	Minimum Requirements
Processor	Pentium 4 or above
Memory	512MB or more
Hard Disk	10 GB or more
Display	Any Monitor
Printer	Epson LQ 300 Dot Matrix Printer
Network	Wired or wireless network

Table A.2 Hardware Requirements For Client Computer

Software Requirements For Server Computer

Software	Minimum Requirements
Operating System	Windows Server latest version
MsSQL Server	Microsoft SQL Server 2008

Table A.3 Software Requirements For Server Computer

Software Requirements For Client Computer

Software	Minimum Requirements
Operating System	Windows XP or above
Ms Visual Studio.Net	Microsoft Visual Studio.Net 2008
Helawadana	Any version

Table A.4 Software Requirements For Client Computer

Step 02

Installing Ms SQL Server 2008 to Server

Install Ms SQL Server 2008 to Server computer which is located in Sanasa Insurance IT department by using MsSQL Server setup DVD.

Step 03

Transfer MsAccess's Databases into MsSQL Server's Databases

Create Colombo, Galle, Gampaha, Kalutara, Kegalle, Kurunegala, Main, Ratnapura databases in the server SQL Server and transfer data from MsAccess databases data into SQL Server databases by using SQL Server tool below which is represented by Figure A.2 .

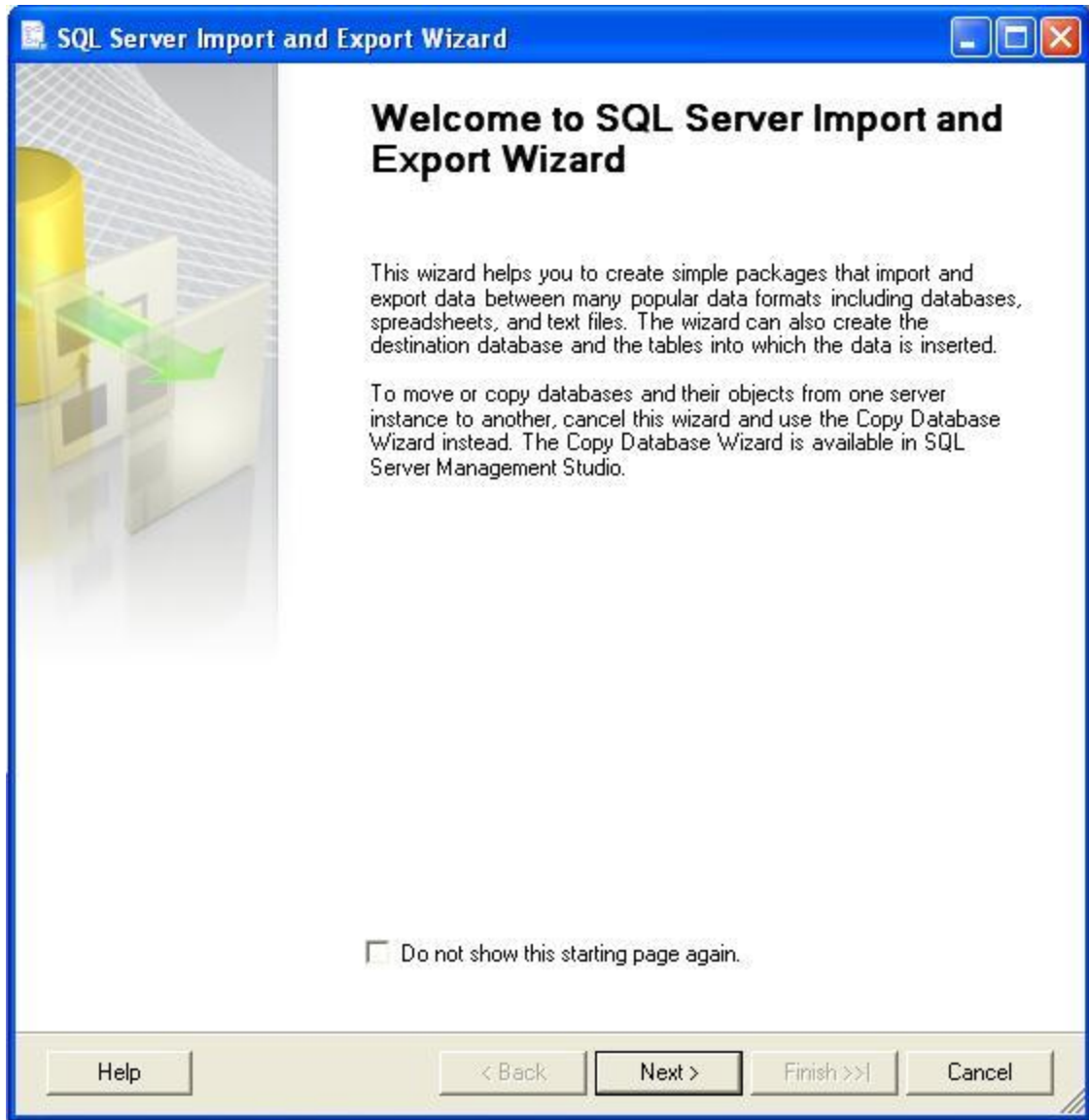


Figure A.2 SQL Server Import and Export Wizard Form

Step 04

Installing MsVisual Studio.Net 2008 to Client Computer

Install MsVisual Studio,Net 2008 to Client computer.

Step 05

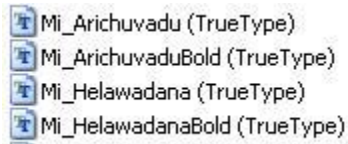
Installing Helawadana to Client Computer

Install Helawadana software to Client computer also.

Step 06

Copy some Sinhala fonts into C:\Windows\Fonts directory.

Copy fonts below into the C:\Windows\Fonts directory; because LifeLink software is a Sinhala fonts based software if not them properly configured it will be a problem.



Step 07

Create LifeLink Software's shortcut on the each Client Computer's Desktop.

Create a shortcut LifeLink\bin\debug\LifeLink.exe file on the each Client Computer's Desktops below which is represented by Figure A.3.



Figure A.3 LifeLink Software's DeskTop Icon

Appendix B – Design Documentation

Figure B.1 shows the UseCase Diagram for Login Module.

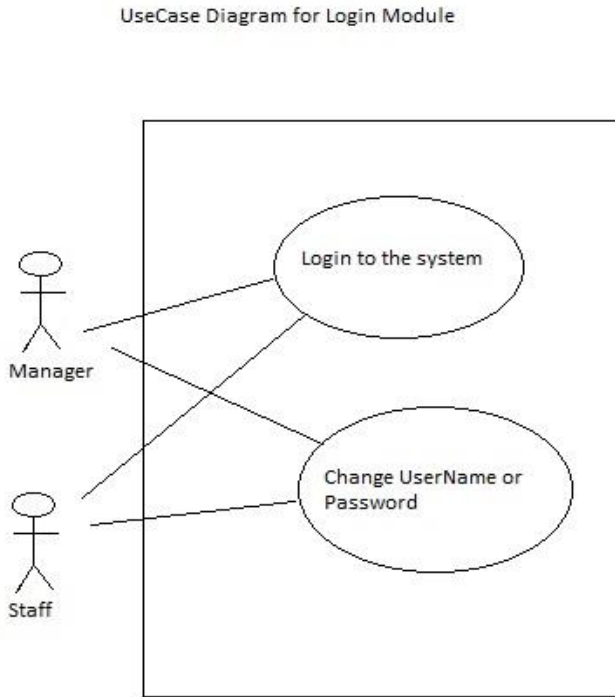


Figure B.1 UseCase Diagram for Login Module

UseCase	Login to the system	
Actors	Manager,Staff	
Description	User can login to the system	
Pre-Conditions	User should have a valid UserName & Password	
Flow of Events	Actor Action	System Response
	1.Type Database Server Name or IP address	
	2.Select the Database	
	3.Type the UserName & Password	
		4.Check this UserName or Password are valid .
	5.Logging to the system.	
Post-Conditions	User can login to the system.	

Table B.1 UseCase Description for Login to the system UseCase

Figure B.2 shows the UseCase Diagram for Search Module.

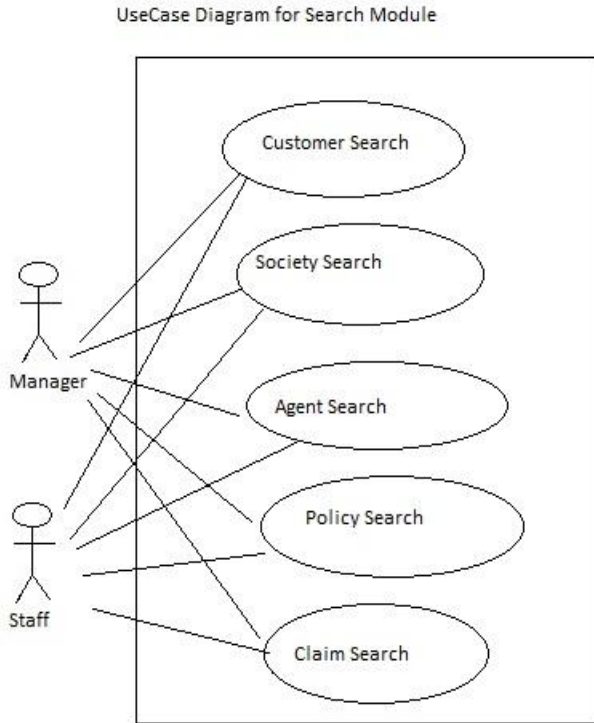


Figure B.2 UseCase Diagram for Search Module

UseCase	Customer Search	
Actors	Manager,Staff	
Description	User can search the customer details	
Pre-Conditions	Users should be logged into the system	
Flow of Events	Actor Action	System Response
	1.Type Policy No or First Name or NIC Number or Society Number and press acceptance button like “Find Now”	
		2.Search the database and find the wanted informations.
		3.Display result on the screen.
Post-Conditions	User can know everything of a customer.	

Table B.2 UseCase Description for Customer Search UseCase

Figure B.3 shows the UseCase Diagram for Policy Maintenance Module.

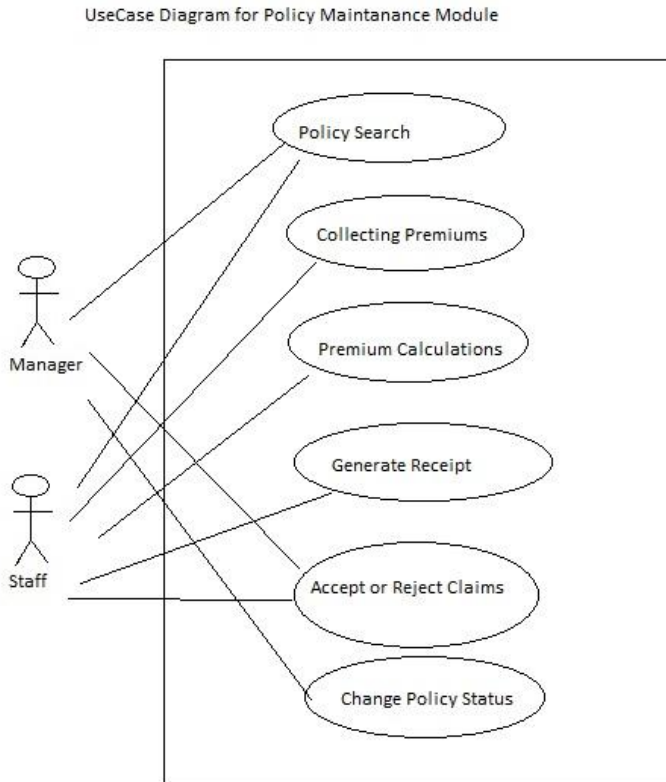


Figure B.3 UseCase Diagram for Policy Maintenance Module

UseCase	Collecting Premiums	
Actors	Staff	
Description	Staff can collect premiums of customers	
Pre-Conditions	Users should be logged into the system	
Flow of Events	Actor Action	System Response
	1.Enter policy Number and click “Done” button.	
		2.Search the details of that policy number and display.
	3.Select Payment Mode and enter Received Amount and press Enter button.	
		4.Do the nessesary calculations.
		5.Display Receipt Report.
Post-Conditions	Customer policy premium details can be updated.	

Table B.3 UseCase Description for Collecting Premiums UseCase

Figure B.4 shows the UseCase Diagram for Claim Maintenance Module.

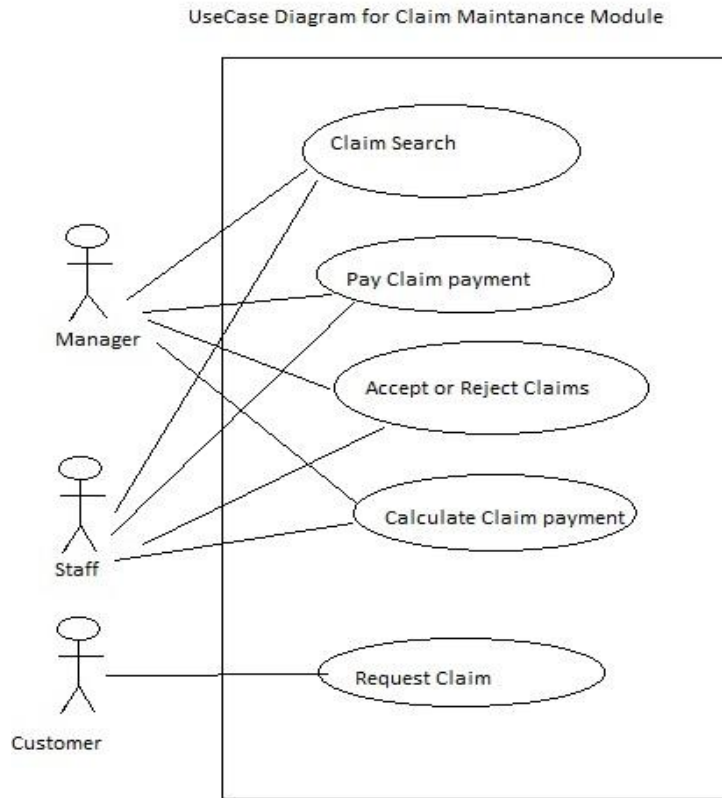


Figure B.4 UseCase Diagram for Claim Maintenance Module

UseCase	Pay Claim Payment	
Actors	Manager,Staff	
Description	Manager and staff can try to pay payments of claims.	
Pre-Conditions	Users should be logged into the system	
Flow of Events	Actor Action	System Response
	1.Enter policy number	
	2.Enter claim amount	
		3.check whether this claim amount is acceptable amount.
	4.If this amount is acceptable one arrange this amount to pay for the customer.	
Post-Conditions	User can pay payments of claims if they are valid.	

Table B.4 UseCase Description for Pay Claim Payment UseCase

(01)Use Case Diagram for Report Generation Module

Figure B.5 shows the UseCase Diagram for Report Generation Module.

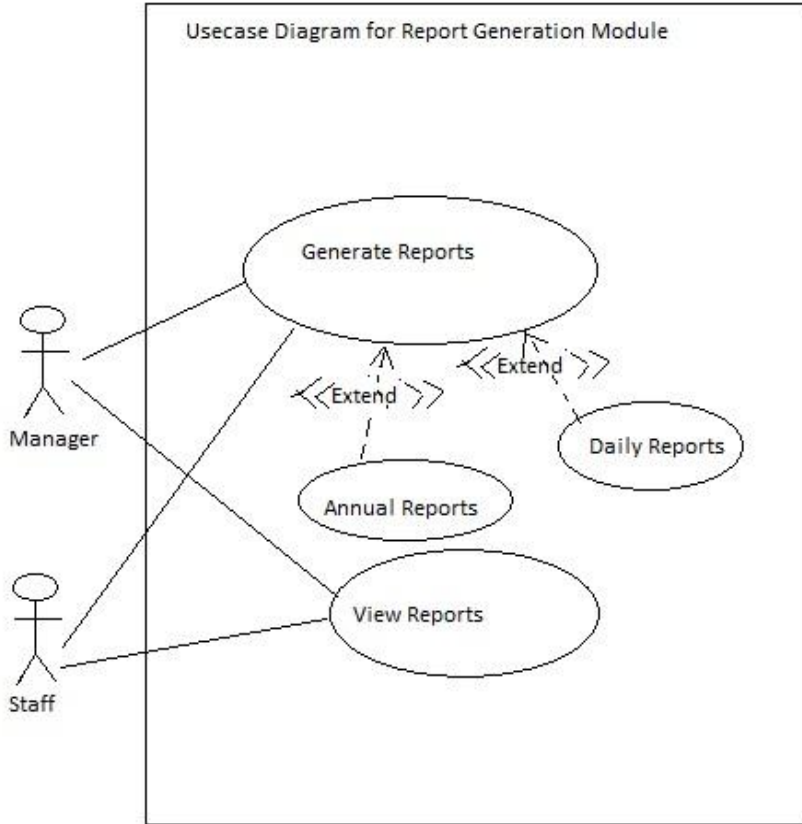


Figure B.5 UseCase Diagram for Report Generation Module

UseCase	Generate Reports	
Actors	Manager,Staff	
Description	Generate the reports of the system	
Pre-Conditions	Users should be logged into the system	
Flow of Events	Actor Action	System Response
	1.Select report type	
	2.Select report category	
	3.Click "View" button	
		4.Generate Report
Post-Conditions	View reports	

Table B.5 UseCase Description for Generate Report UseCase

Appendix C – User Documentation

This is Login window which is represented by Figure C.1



The screenshot shows a 'Login' dialog box with the following fields and controls:

- Login Server :** A text box containing '192.168.1.240' and a 'Change And Save' button.
- Select Database :** A dropdown menu with 'Main' selected.
- User Name :** An empty text box.
- Password :** An empty text box.
- Buttons:** 'OK' and 'Cancel' buttons at the bottom.

Figure C.1 Login Form

- (1) Select the database you want
- (2) Enter user name at the “User Name” area
- (3) Enter password at the “Password” area
- (4) Then press “OK” button
- (5) You don't want to log into the system ;press “CANCEL” button

Figure C.2 shows the Home Form .

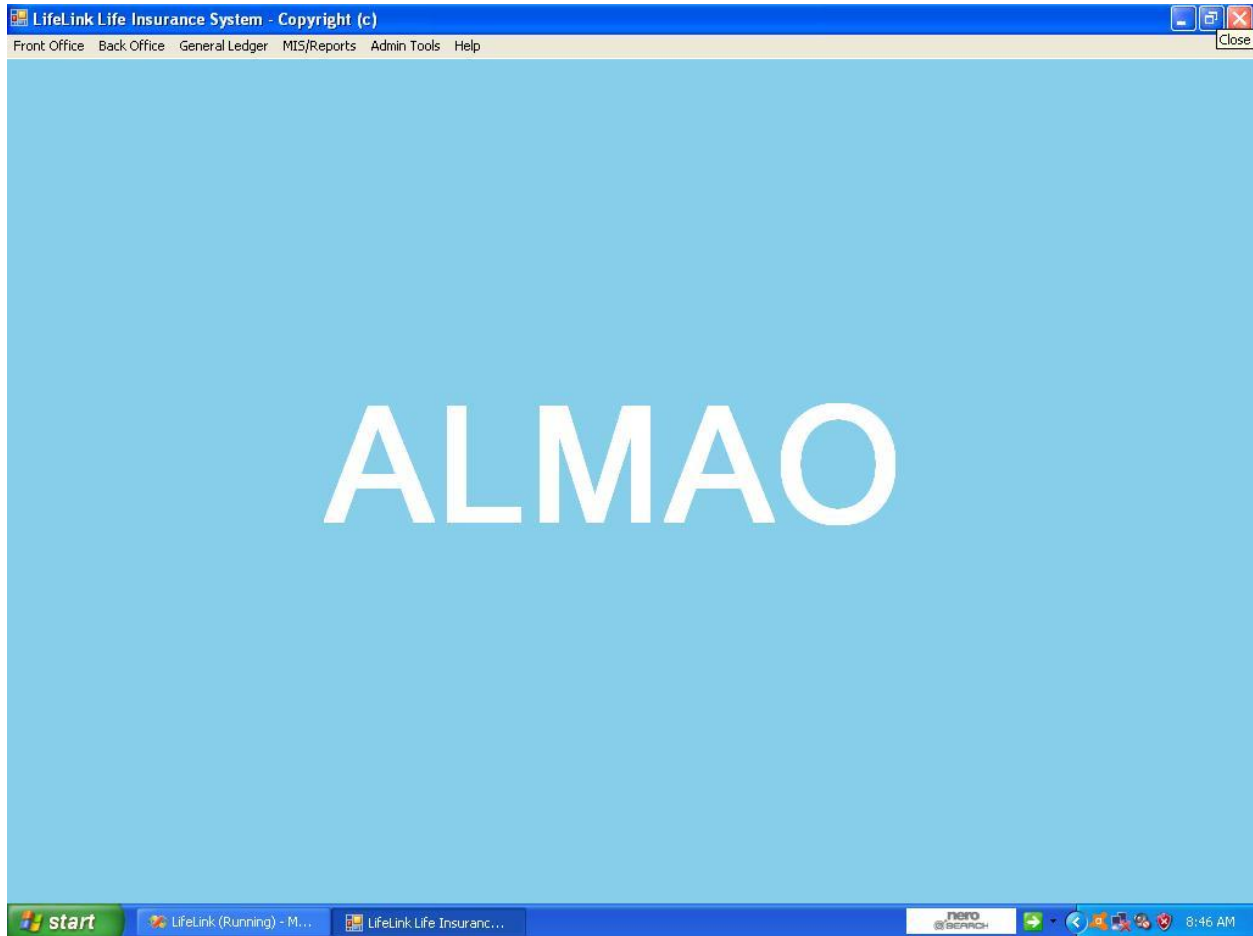


Figure C.2 Home Form

This is search options of front office menu which is represented by Figure C.3

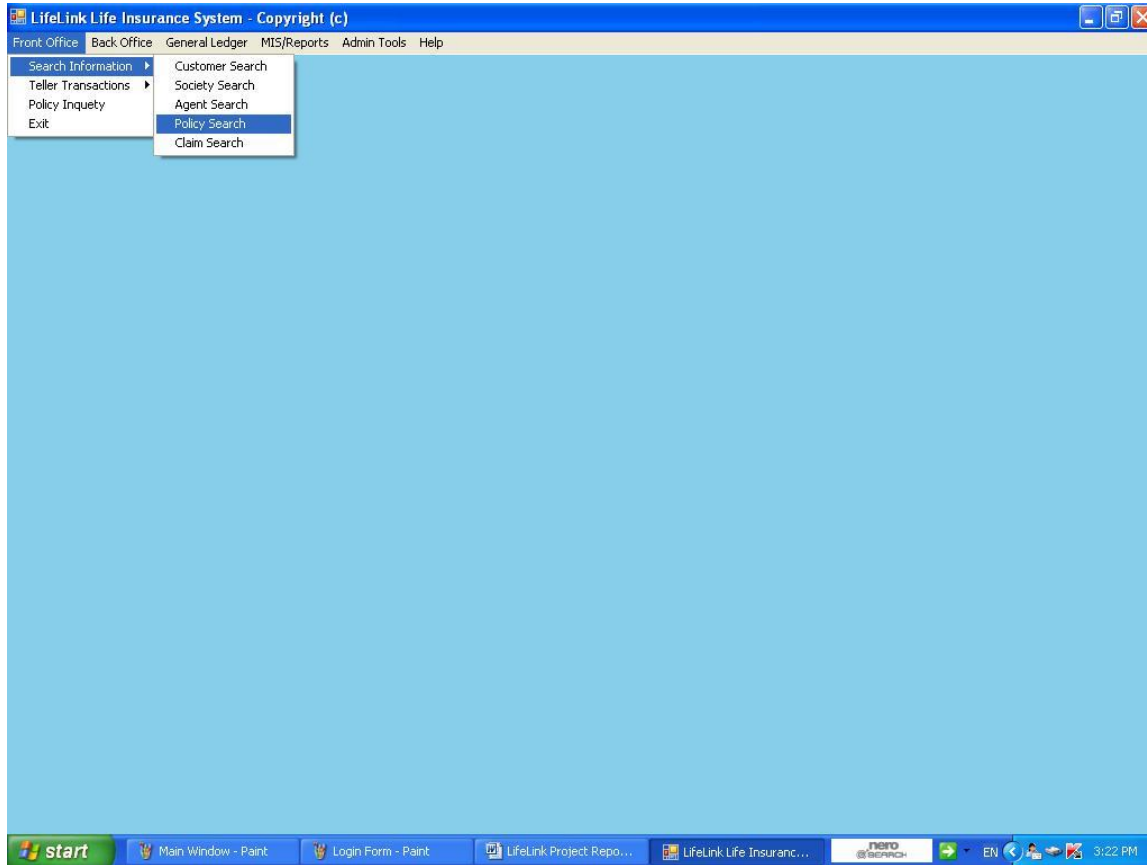


Figure C.3 Popup Menues

This window can use to search customer informations which is shown by Figure C.4.

The screenshot shows a window titled "CustomerSearch" with a standard Windows-style title bar (minimize, maximize, close). The main area of the window has a light gray background. At the top center, it says "Database : Main". Below this, on the left, is a "Search By:" label followed by a dropdown menu showing "Containing Text". Underneath are three radio button options: "Frist Name" (which is selected), "NIC Number", and "Society No". To the right of the "Frist Name" option is a text input field. To the right of the "Society No" option is a checkbox labeled "Draft Output". On the right side of the window, there are three buttons stacked vertically: "Find Now", "Print", and "Close". The "Find Now" button is highlighted with a blue border. The bottom half of the window is a large, empty gray rectangular area, likely intended for displaying search results.

Figure C.4 Customer Search Form

(1)There are three options to search a customer;If you want to search a customer by using first name ; select the “First Name” option; after type you want first name and then press “Find New” button.

(2) If you want to search a customer by using NIC number ; select the “NIC Number” option; after type you want NIC number and then press “Find New” button.

(3)If you want to search a customer by using society code ; select the “Society no” option; after type you want society code and then press “Find New” button.

This window can use to search society informations which is shown by Figure C.5.

The screenshot shows a software window titled "SocietySearch". At the top center, it says "Database : Colombo". Below this, there is a "Search By:" label followed by a dropdown menu currently showing "Containing Text". To the right of the dropdown are three buttons: "Find Now", "Print", and "Close". On the left side, there are three radio button options: "Name" (which is selected), "Division Name", and "District Name". Next to the "Name" option is a text input field. Below the "District Name" option is a checkbox labeled "Draft Output". The bottom half of the window is a large, empty grey rectangle, intended for displaying search results.

Figure C.5 Society Search Form

(1) There are three options to search a society ; If you want to search a Society by using name ; select the “Name” option; after type you want name and then press “Find New” button.

(2) If you want to search a society by using division name ; select the “Division Name” option; after type you want division name and then press “Find New” button.

(3) If you want to search a society by using district name ; select the “District Name” option; after type you want district name and then press “Find New” button.

This window can use to search agent informations which is shown by Figure C.6.

The screenshot shows a software window titled "Agent Search". At the top right, there are standard window control buttons (minimize, maximize, close). Below the title bar, it says "Database : Colombo". The main area contains a "Search By:" label followed by a dropdown menu currently showing "Containing Text". Below this are three radio button options: "First Name" (which is selected), "NIC Number", and "Society Code". To the right of the "First Name" option is a text input field. To the right of the "Society Code" option is a checkbox labeled "Draft Output". On the right side of the window, there are three buttons: "Find Now", "Print", and "Close". The bottom half of the window is a large, empty grey rectangular area, likely intended for displaying search results.

Figure C.6 Agent Search Form

(1) There are three options to search an agent; if you want to search an agent by using first name; select the "First Name" option; after type your first name and then press "Find Now" button.

(2) If you want to search an agent by using NIC number; select the "NIC Number" option; after type your NIC number and then press "Find Now" button.

(3) If you want to search an agent by using society code; select the "Society Code" option; after type your society code and then press "Find Now" button.

This window can use to search policy informations which is shown by Figure C.7.

PolicySearch Database : Colombo

Search By: Containing Text Find Now

Customer Code

Agency Code Print

Society Code Draft Output Close

Name

Address

Figure C.7 Policy Search Form

- (1) There are three options to search a policy ;If you want to search a policy by using customer code ; select the “Customer Code” option ;after type you want customer code and then press “Find New” button.
- (2) If you want to search a policy by using agent code ; select the “Agent Code” option ;after type you want agent code and then press “Find New” button.
- (3) If you want to search a policy by using society code ; select the “Society Code” option ;after type you want society code and then press “Find New” button.

This window can use to search claim informations which is shown by Figure C.8.

Claim Search

Database : Colombo

Search By: Containing Text Find Now

Policy No

Customer Code Print

Society Code Draft Output Close

Name

Address

Figure C.8 Claim Search Form

- (1) There are three options to search a claim ; If you want to search a claim by using policy number ; select the “Policy No” option; after type you want policy number and then press “Find New” button.
- (2) If you want to search a claim by using customer code ; select the “Customer Code” option after type you want customer code and then press “Find New” button.
- (3) If you want to search a claim by using society code ; select the “Society Code” option after type you want society code and then press “Find New” button.

This is Teller Transactions options of front office menu which is shown by Figure C.9.

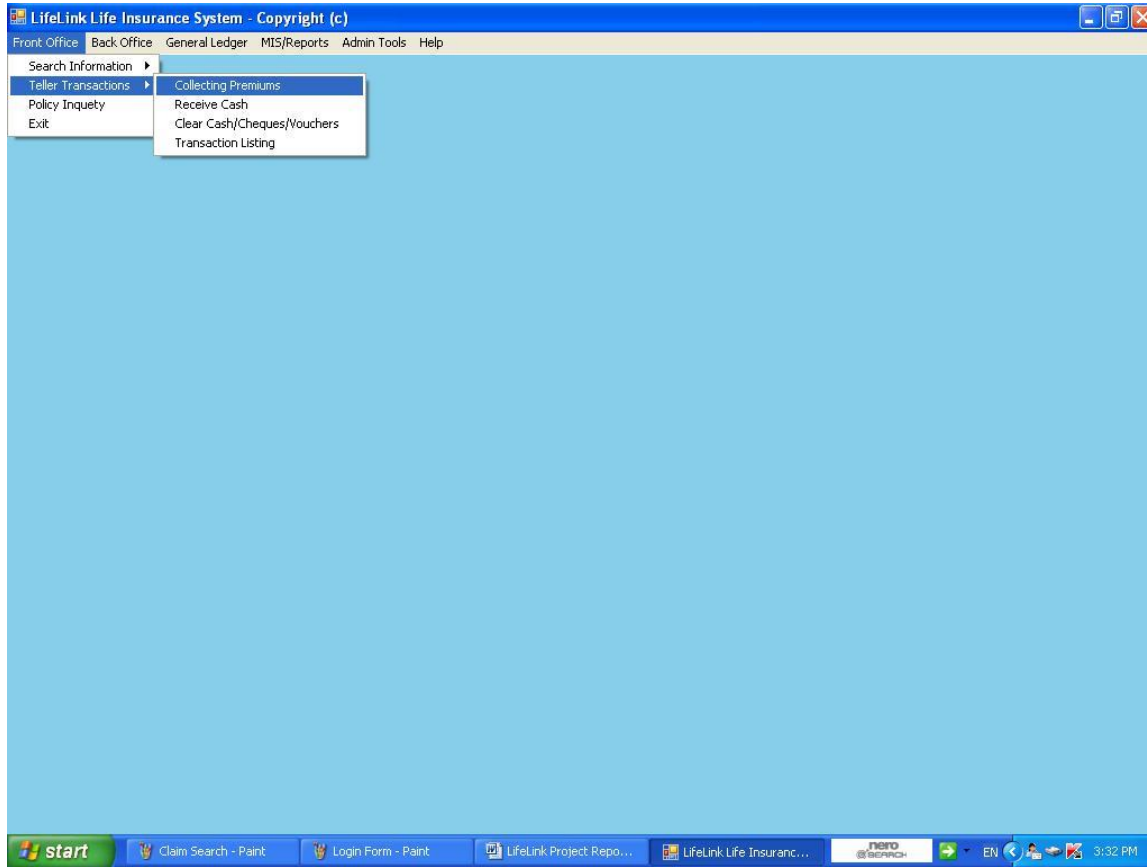


Figure C.9 Popup Menu Form Collecting Premium Form

This window can use to collect premiums which is shown by Figure C.10.

Front Office Transactions - CollectingPremium
Database : Main

Policy Number Customer Number

Client's Name and Address

Frequency Policy Type

Payment Mode Details/Cheque #

Nos Premiums 0 Amount Received 0 Label18

Appropriated As Follows

Unappropriated Balance
B/F 0.00
C/F 0

Premium Details
(MM/DD/YY)
FROM
TO 0

Done Re-Print Cancle Type the Policy Number and Press <Enter> Key Close

Figure C.10 Collecting Premium Form

(1)First type the policy number you want at the particular “Policy Number” area and then press the “Enter” button of your keyboard .Then it will display relevant informations.

(2)After fill the “Payment Mode” and “Amount Received” areas and press the “Enter” button of the keyboard of “Done” button.

(3)Afterthat it will display a report also.

This window can use to clear cash or cheques or vouchers which is shown by Figure C.11.

	Cash	Cheque	Vouchers	Money Orders
Amounts B/F	0	0	0	0
Amounts Recorded	3050	1100	500	400
Amounts in Hand	3050	1100	500	400
Amounts Clear	3050	1100	500	400

Figure C.11 Cash Clear Form

(1) Enter amounts “Amount in Hand” and “Amount Clear” and press the “Done” button. Then it will display a another report.

This window can use to inquiry about various policies which is shown by Figure C.12.

Policy Inquiry Database : Main

Policy Number Customer Number

Policy Information Beneficiary & Payment Information

Date Of Birth (MM/DD/YY) Society Name

Monthly Premium Agency Code

Premium Frequency Open Date (MM/DD/YY)

Premium Maturity Date (MM/DD/YY)

Terms in years Basic Cover

Exp. Claim Rate Maturity Value

Exp. Claim Days Minor Dissable

Accident Cover Partial Dissable

Double Acc. Cover Full Dissable

Search If Error Type the Policy Number and Press <Search> Button Close

Figure C.12 Policy Inquiry Form

(1) First type the policy number you want at the “Policy Number” textbox area and then press the “Search” button or “Enter” button at the keyboard.

This window displays sub menu options of Claim Maintenance menu option which is shown by Figure C.13.

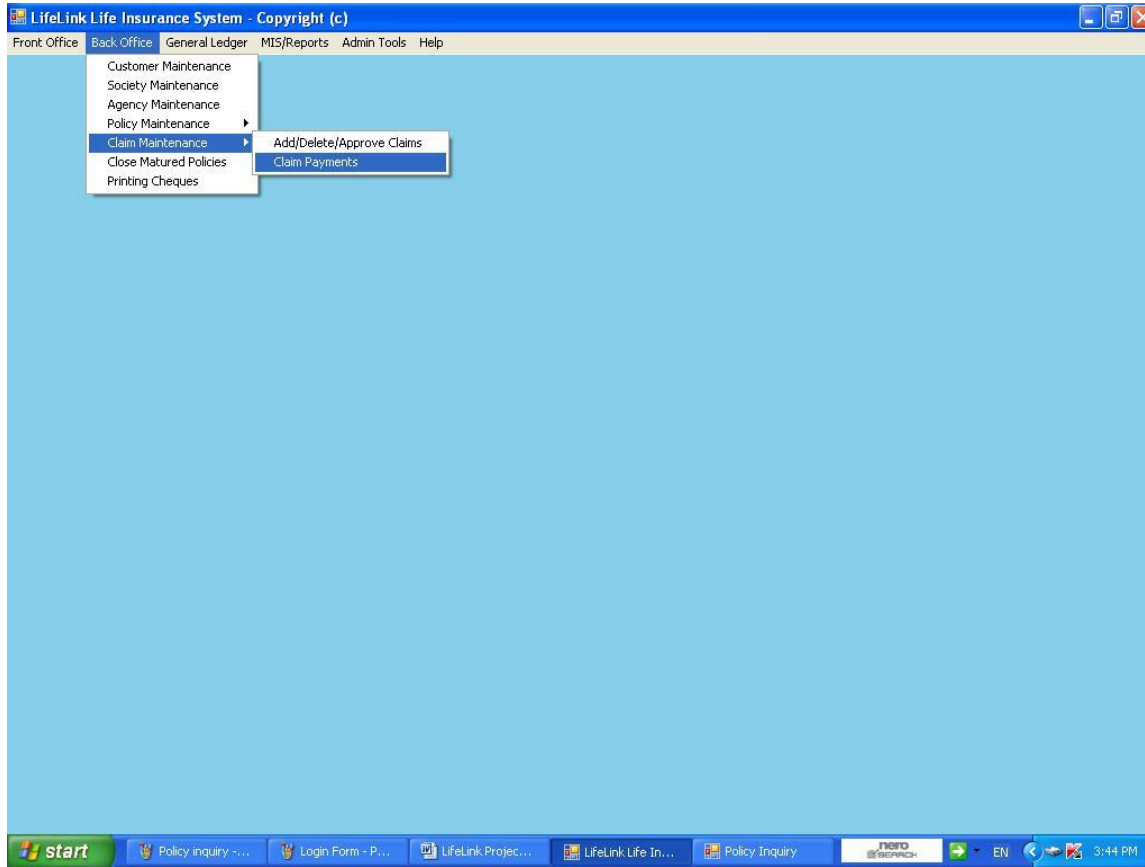


Figure C.13 Popup Menu For Claim Payments Form

These 3 windows display various options of Policy Claim Maintanances window which is shown by Figure C.14,C.15 And C.16.

(a)

Figure C.14 Policy Claim Maintenance Form

- (1)Enter the policy number you want at the “Policy Number” area and press the “Enter” key at the keyboard or press the “Search” button.Then it will display the relevant informations.
- (2)Afterthat select the suitable option which is “Claim on this” option.
- (3)Press “Add” button.Then it will dispay a report also.
- (4)After press “Approve” button also.Then it will display another report.

(b)

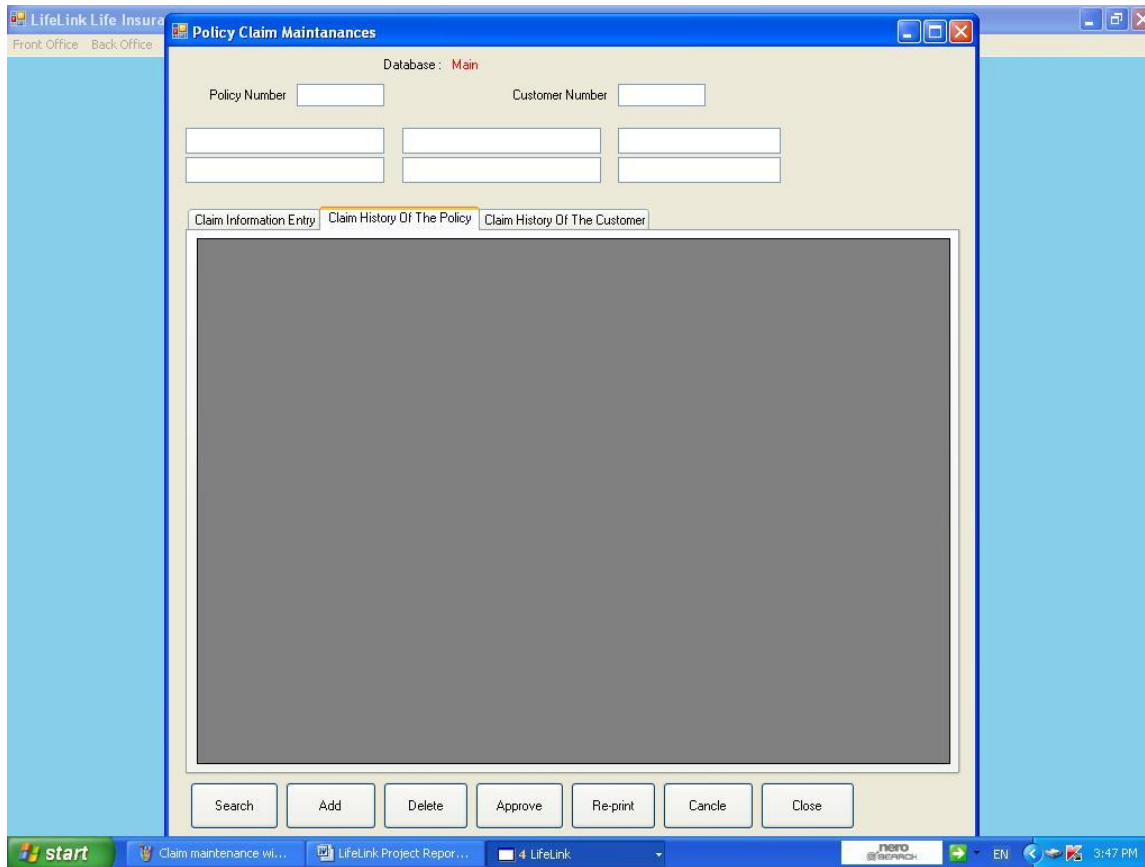


Figure C.15 Policy Claim Maintenance Form

(c)

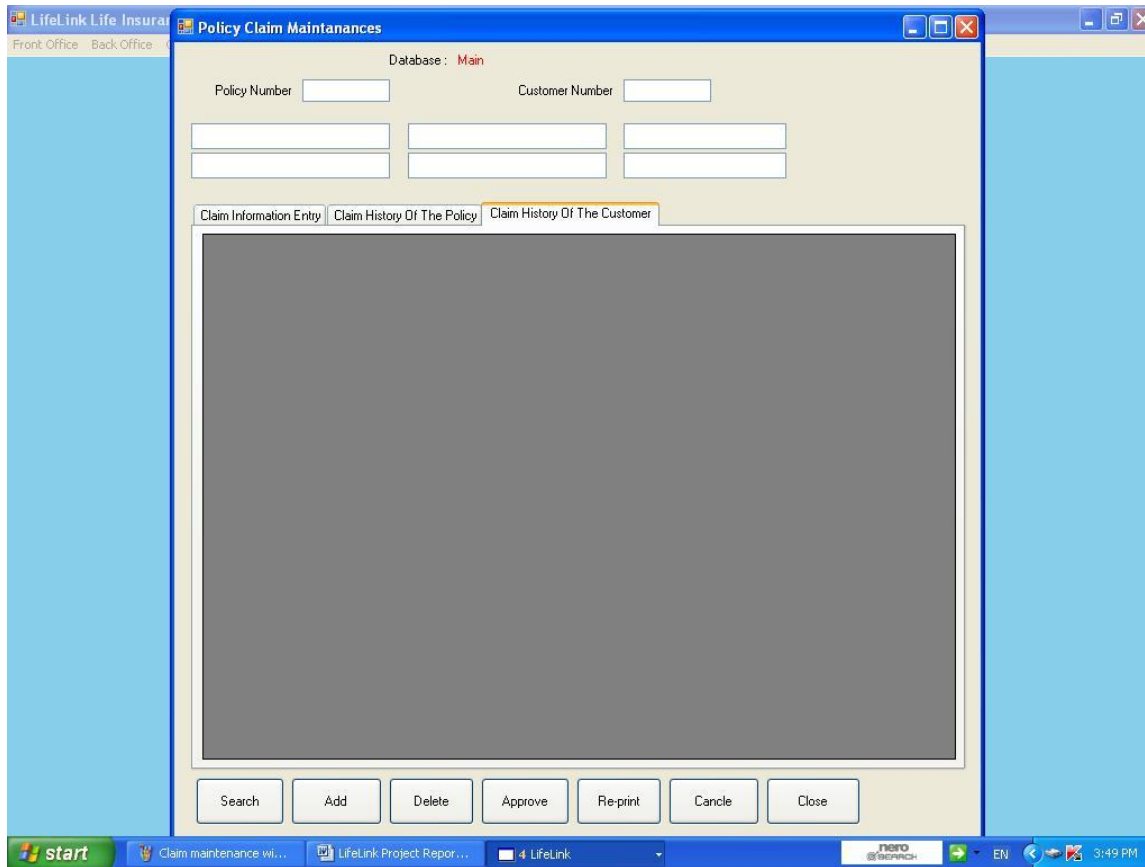


Figure C.16 Policy Claim Maintenance Form

This window displays Claim Payments informations which is shown by Figure C.17.

The screenshot shows a software window titled "Claim Payment" with a blue border and standard Windows window controls (minimize, maximize, close). The window content is as follows:

- At the top right, it says "Database : Main".
- Below that, there are two text input fields: "Claim Number" and "Policy No".
- The "Insured's Information" section contains a "Customer No" field followed by three more empty text input fields.
- The "Claim Information" section contains four fields: "Claim Type", "Requested Date", "Sub Type", and "Amount Approved".
- The "Payment Information" section contains four fields: "Payment Mode" (a dropdown menu currently showing "Cheque"), "Amount Paid", "Cheque Bank", and "Cheque No".
- At the bottom of the window, there are three buttons: "Search", "Done", and "Close".

Figure C.17 Claim Payment Form

- (1) Enter the Claim Number you want and press the "Search" button or press the "Enter" key at the keyboard. Then it will display the particular informations.
- (2) After that enter the "Payment Mode", "Cheque Bank", "Amount Paid" and "Cheque No" and press the "Done" button. Then it will display the report you want.

This window displays various menu options of MIS/Reports menu which is shown by Figure C.18.

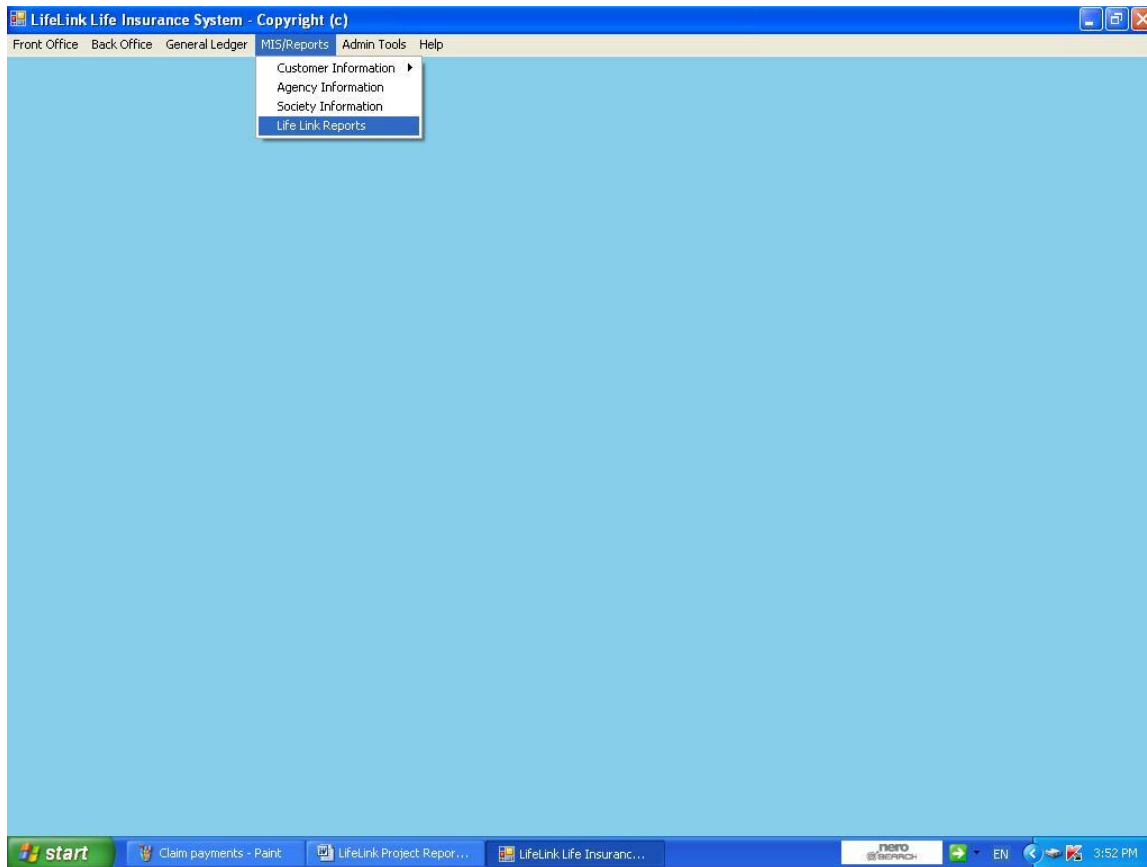


Figure C.18 Popup Menu For LifeLink Reports Form

This window can use to get various reports which is shown by Figure C.19.

The screenshot shows a window titled "Life Link Reports" with standard window controls (minimize, maximize, close). The window contains the following elements:

- A dropdown menu labeled "Select the report name to view/print" with a small downward arrow on the right.
- A text area labeled "Report Description" with a large empty rectangular box below it.
- A "Report Category" section on the right side containing five radio button options:
 - Daily Reports
 - Monthly Reports
 - Ad hoc Reports
 - GL Reports
 - All Reports
- Two buttons at the bottom: "View" on the left and "Close" on the right.

Figure C.19 LifeLink Reports Form

(1) Select the report type you want and press "View" button. Then it will display the report you want.

Appendix D – Management Reports

(01)Collecting Premium

Collecting Premium is the first business process of ALMAO system and finally generate a report which is shown by figure D.1.

සමස්ථ ලංකා අනුපාතය ආරක්ෂණ සංවිධානය
නො 40, කරඬුපහ, කෑගල්ල.

අනේතුක විපත් ප්‍රතිලාභ ක්‍රමය
වාරික ලෙසට 7 4:48:07 pm

සමිති අංකය	MT008
නියෝජිත අංකය	00200
වාරිකය	50.00 15

නිවිසුම් අංකය	12343
දිනය	06/06/2012
ලෙසන් අංකය	460,464

එස්. රසික සන්නය සේනාරත්න
පොලුණුවත්ත
රුචිච්ච
මරියස

ඉදිරියට ගෙන ආ අසම්පූර්ණ වාරික ශේෂය	80.00
මුදල් ලැබීම්	790.00
මුළු එකතුව	870.00
අඩුකළා වාරික	750.00
ඉදිරියට ගෙන යන අසම්පූර්ණ වාරික ශේෂය	120.00

හිඟ වාරික ගණන	150		
වාරික අදාල කාලය	09/20/1998	සිට	13
(ඊලඟ වාරිකය ගෙවිය යුතු දිනය)	12/20/1999	දක්වා	28

DESIGN & PRINTED BY: OVATA FORMS (PVT) LTD. 5748240, 5748019, 0772 941930 The Ultimate In Forms

Figure D.1 Collecting Premium Report

(02)Cash Clear Report

Cash Clear is the second business process of ALMAO system and finally generate a report which is shown by figure D.2.

Tran# : 475,993

Date : 10/27/201

Time : 1:01:53 pm

Teller Clear Cash / Cheque / Vouchers and Money Orders for the Day

Teller# 3 Teller Name : 009

<u>Description</u>	<u>Amount SLR</u>
Cash Cleared	3,050.00
Cheque Cleared	1,100.00
Vouchers Cleared	500.00
Money Orders Cleared	400.00
Total Amounts Cleared :	<u>5,050</u> =====

Amounts Handed Over

Amounts Received

Teller's Signature

Chief Cashier's Signature

Figure D.2 Cash Clear Report

(03)Society Wise Daily Report

According to ALMAO business process; end of the day they get some daily reports. One is “Society Wise Daily Report” and it is shown by figure.D.3.

SANASA ALMAO INSURANCE COMPANY LTD.							
Listing of society Life insurance Premium collations for the day of					10/27/2017		
KE298	RENAPANA KAPPETTIHENA RANAPANA			KAPPETTIHENA			
Receipt No	Policy No	Insured Name		Paied Amount	Agent Code	No Of Prem	
475,984	109334	ආදී	මාලනී රාජසත්ත	රාජසත්ත	500.00	00038	185
475,986	109334	ආදී	මාලනී රාජසත්ත	රාජසත්ත	100.00	00038	183
475,987	27911	සේන	අනුමලකා	කීර්තිය	500.00	00200	184
475,988	109335	එච්	ජයන්තා ලාලනී	සේනේසොලසේ	400.00	00038	186
475,989	27911	සේන	අනුමලකා	කීර්තිය	600.00	00200	172
475,990	109335	එච්	ජයන්තා ලාලනී	සේනේසොලසේ	250.00	00038	181
475,991	109330	සු.එල්.	අනුලාචනී	උඩවෙල ලේකම්ලාසේ	1,000.00	00038	184
475,992	109333	එච්.ආදී.	ධම්මික	කුමාර ජ. සේනා රත්නසේ	1,500.00	00038	179
Total of Daily Income:				4,850.00			

Figure D.3 Society Wise Daily Report

(04) Society Wise Policy Status Report

According to ALMAO business process; end of the day they get some daily reports. Other is “Society Wise Policy Status Report” and it is shown by figure.D.4.

ALMAO Insurance Organization No:40,Karadupona,Kegalle.									
Society wise Life Insurance policy status as at 10/27/2017									
KE003 DIPPITIYA SANASA UNLTD DIPPITIYA			MAHAPALLEGAMA						
PolicyN	Customer Name		Monthly Premium	Premium Type	Agent Code	Nos Due Premium	Account Balance	Inappropriate Balance	Premium Due Date
6749	ANULA DISANAYAKA	DISANAYAKA MUDIYANSE D.M.	0.00	1	00200	0	0.00	0.00	
6750	ARIYAWATHI	MAHAGAMA RALALAGE M.R.	0.00	1	00200	0	0.00	0.00	
1947	NANDADASA	KARANAKA PATHIRANNAI K.P.	0.00	1	00200	0	0.00	0.00	
6515	MALANI CHANDRALATHA	GALGODAWATTA HEWAYA G.H.	0.00	1	00200	0	0.00	0.00	
6516	LEELAWATHI	SUDUHAKURUGE S.	0.00	1	00200	0	0.00	0.00	
6517	MAGILIN	PALUMULLALAGE P.	0.00	1	00200	0	0.00	0.00	
6518	INDRANI KUSUMALATHA	PALUMULLALAGE P.	0.00	1	00200	0	0.00	0.00	
6519	SIMON	IHALAGEDARA HEWAYAL I.H.	0.00	1	00200	0	0.00	0.00	
6520	SIRISENA	WEERASINGHA LIAYNAGE W.L.	0.00	1	00200	0	0.00	0.00	
6521	JINADASA	GURULLAPITI DEWALAGE G.D.	0.00	1	00200	0	0.00	0.00	
6522	RATHNAPALA	KURUPATHHALAGE K.	0.00	1	00200	0	0.00	0.00	
6528	CHANDRIKA PRIYADARS	RMMANDALAMUTUTHAN R.M.	0.00	1	00200	0	0.00	0.00	
6529	KAMALAWATHI	MUTHUKUDAWJESURIY M.W.A.	0.00	1	00200	0	0.00	0.00	

Figure D.4 Society Wise Policy Status Report

(05)Claim Add Report

“Claim Add Report” is the first report of the Claim Process of ALMAO and it tells new claim entry and it is shown by figure D.5.

Sanasa ALMAO Insurance Company LTD- No:40 ,Karadupona , Kegalle.
Claim Information - New Entry on 11/1/2017

Policy Number : 110818

Claim Number : 2,136

Assured's Number : 32268

Assured's Name : SIKURADIPATHI GAN GNANAWATHI S.G.

Address : HETTIMULLA KOTADENIYAWA

Date Of Birth : 07/22/1948

Date Of Open : 01/20/2002

Date Of Maturity : 01/20/2017

Premium Amount : 50.00

Nos Terms : 15

Account Balance : 50.00

Basic Cover : 9,000.00

Nos Premiums Paid: 1

Nos Premiums Due: 0

Claim Category : Hospital Charge

Claim Description : 001-Road Accidents

Claim Amount : 100.00

Figure D.5 Claim Add Report

(06) Claim Approve Report

“Claim Approve Report” is the second report of the Claim Process of ALMAO and it tells approval of the new claim entry and it is shown by figure D.6.

Sanasa ALMAO Insurance Company LTD. No:40 ,Karaduona , Kegalle.
Claim Information - Approve Claim on 11/1/2017

Policy Number : 110818

Claim Number : 2,136

Assured's Number : 32268

Assured's Name : SIKURADIPATHI GAN GNANAWATHI S.G.

Address : HETTIMULLA KOTADENIYAWA

Date Of Birth : 07/22/1948

Date Of Open : 01/20/2002

Date Of Maturity : 01/20/2017

Premium Amount : 50.00

Nos Terms : 15

Account Balance : 50.00

Basic Cover : 8,900.00

Nos Premiums Paid: 1

Nos Premiums Due: 0

Claim Category : Hospital Charge

Claim Description : 001-Road Accidents

Claim Amount : 100.00

Figure D.6 Claim Approve Report

(07)Claim Payment Report

Figure D.7 shows Claim Payment Report. This report shows claim payment details.

Sanasa ALMAO Insurance Company LTD - No.40 , Karadupona , Kegalla.
Claim Information - Claim Payment on 11/1/2017

Policy Number : 110818

Claim Number : 2,136

Assured's Number : 32268

Assured's Name : SIKURADIPATHI GANITHAYALAGE GNANAWATHI

Address : HETTIMULLA KOTADENIYAWA

Date of Birth : 07/22/1948

Claim Category : Hospital Charge

Claim Description : 001-Road Accidents

Claim Amount : 2,000.00

Payment Made By : Cheque

Description : People's Bank

Number : 2354232

Figure D.7 Claim Approve Report

Appendix E – Test Results

Test Cases For User Login Module

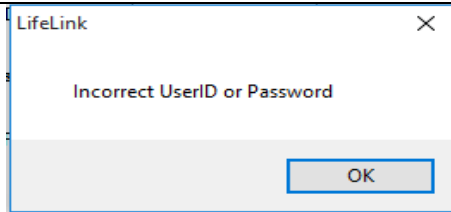
Test No	Test Case	Expected Output	Actual Output	Status
1	When incorrect UserName or Password is entered	Display a message called “Incorrect User ID or Password”		Pass

Table E.1 Test Cases For User Login Module

Test Cases For Collecting Premium Module

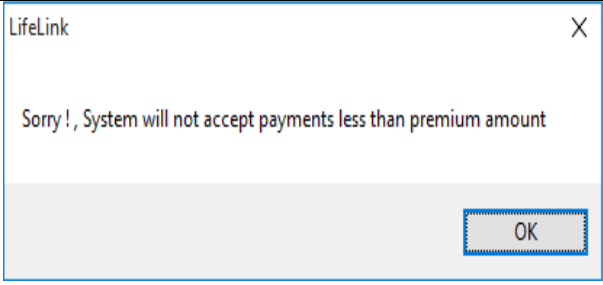
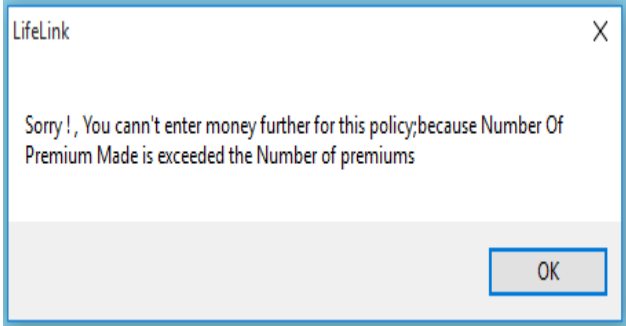
Test No	Test Case	Expected Output	Actual Output	Status
1	When entering a payment which is less than or equal to premium amount.	Display a message called “Sorry ! , System will not accept payments less than premium amount”		Pass
2		Display a message called "Sorry ! , You can't enter money further for this policy;because Number Of Premium Made is exceeded the Number of premiums"		Pass

Table E.2 Test Cases For Collecting Premium Module

Test Cases For Policy Claim Maintenance Module

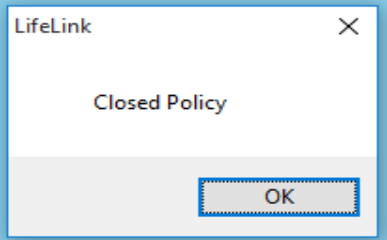
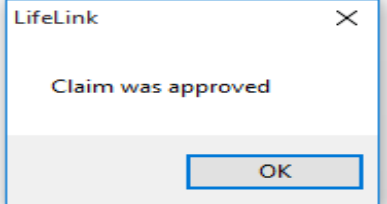
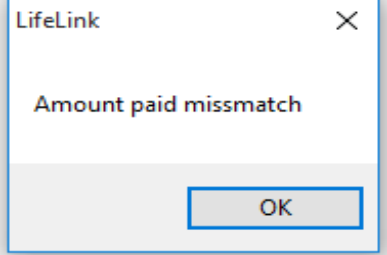
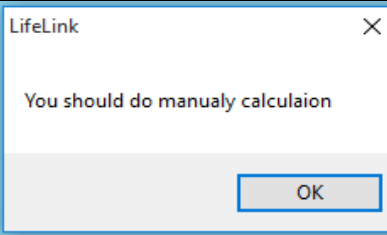
Test No	Test Case	Expected Output	Actual Output	Status
1	When closed policy number is entered	Display a message called "Closed Policy"		Pass
2	If a claim is approved	Display a message called "Claim was approved"		Pass
3	When amount to pay is greater than to Basic Cover	Display a message called "Amount paid mismatch"		Pass
4	When request a claim for "Sahana" type policy	Display a message called "You should do manually calculation"		Pass

Table E.3 Test Cases For Policy Claim Maintenance Module

Test Cases For Ledger Module

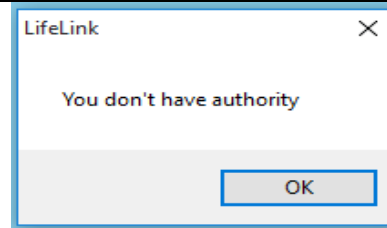
Test No	Test Case	Expected Output	Actual Output	Status
1	When wrong User Level's person is tried to access Ledger	Display a message called "You don't have authority"		Pass

Table E.4 Test Cases For Ledger Module

Test Cases For Policy Inquiry Module

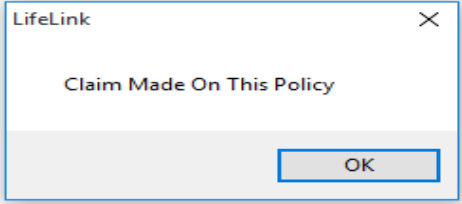
Test No	Test Case	Expected Output	Actual Output	Status
1	When a claim is already has been requested on this policy	Display a message called “Claim Made On This Policy”		Pass

Table E.5 Test Cases For Policy Inquiry Module

Test Cases For Change Password Module

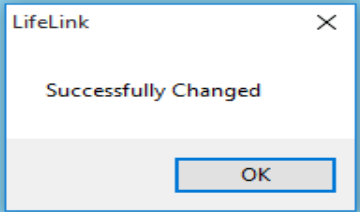
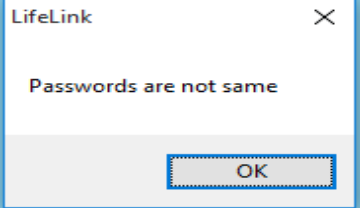
Test No	Test Case	Expected Output	Actual Output	Status
1		Display a message called “Successfully Changed”		Pass
2		Display a message called “Passwords are not same”		Pass

Table E.6 Test Cases For Change Password Module

Test Cases For End Of Day Process Module

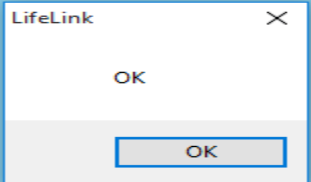
Test No	Test Case	Expected Output	Actual Output	Status
1				Pass

Table E.7 Test Cases For End Of Day Process Module

Test Cases For To Do Policy Active Module

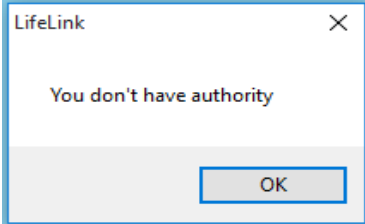
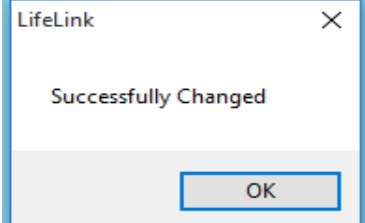
Test No	Test Case	Expected Output	Actual Output	Status
1		Display a message called "You don't have authority"		Pass
2		Display a message called "Successfully Changed"		Pass

Table E.8 Test Cases For To Do Policy Active Module

Test Cases For To Do Policy Close Module

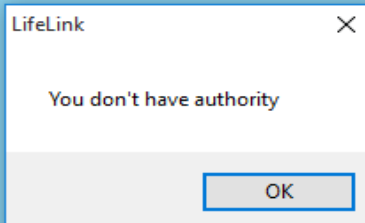
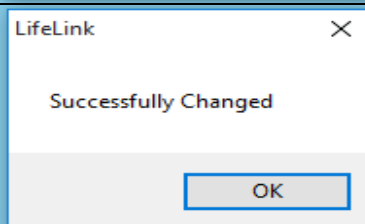
Test No	Test Case	Expected Output	Actual Output	Status
1		Display a message called "You don't have authority"		Pass
2		Display a message called "Successfully Changed"		Pass

Table E.9 Test Cases For To Do Policy Close Module

Appendix F – Code Listing

(01)User Login Event

```
Imports System.Data.SqlClient
Imports System.Data
Imports System.Text
```

```
Imports System.Security.Cryptography
Imports System
Imports System.IO
Imports System.Security
Imports System.Runtime.InteropServices
```

```
·*****
```

```
Public Class LoginForm1
```

```
    Dim MyConn As SqlConnection
    Dim uu As SqlDataReader
    Dim MyCommand1 As SqlCommand

    Dim Form1 As New LifeLink.Form1
```

```
·*****
```

```
Private Sub OK_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles OK.Click
```

```
    PasswordTextBox.Text = EncryptPassword(PasswordTextBox.Text)
```

```
    If (CheckCredentials(UsernameTextBox.Text, PasswordTextBox.Text)) Then
```

```
        MyConn = New SqlConnection("Data Source=" & TextBox1.Text & ";Initial Catalog=" &
        ComboBox1.Text & "; User ID=sa; Password=sic1almao; ")
```

```
        MyCommand1 = New SqlCommand("UPDATE [User] SET LLogDate=@LLogDate WHERE UserID=" &
        UsernameTextBox.Text & """, MyConn)
```

```
        MyConn.Open()
```

```
        MyCommand1.CommandType = CommandType.Text
```

```
        Dim param1 As New SqlParameter("@LLogDate", SqlDbType.DateTime)
        param1.Direction = ParameterDirection.Input
        param1.Value = Label11.Text
```

```
        MyCommand1.Parameters.Add(param1)
```

```
        MyCommand1.ExecuteNonQuery()
        MyConn.Close()
```

```
        Me.Hide()
```

```

Form1.Show()

Dim MyCommand As SqlCommand

MyCommand = New SqlCommand("SELECT AccessLevel FROM [USER]WHERE UserID='" &
UsernameTextBox.Text & "'", MyConn)

Dim MyReader As SqlDataReader
MyConn.Open()
MyCommand.CommandType = CommandType.Text
MyReader = MyCommand.ExecuteReader

Do While MyReader.Read
    Label4.Text = MyReader("AccessLevel")

Loop

MyConn.Close()
Dim MyCommand2 As SqlCommand

MyCommand2 = New SqlCommand("SELECT UserID FROM [USER]WHERE UserID='" &
UsernameTextBox.Text & "'", MyConn)

MyConn.Open()
MyCommand2.CommandType = CommandType.Text
MyReader = MyCommand2.ExecuteReader

Do While MyReader.Read
    Label5.Text = MyReader("UserID")

Loop

Else

    MsgBox("    Incorrect UserID or Password    ")

End If

MyConn.Close()
End Sub

'*****

Function CheckCredentials(ByVal UserID As String, ByVal Pass As String) As Boolean

    MyConn = New SqlConnection("Data Source=" & TextBox1.Text & ";Initial Catalog=" & ComboBox1.Text & ";
User ID=sa; Password=sic1almao;")

    MyConn.Open()

```

```

Dim MyCommand As SqlCommand
MyCommand = New SqlCommand("SELECT UserID,PWard FROM [User]WHERE
UserID=@givenUsername AND PWard=@givenPassword", MyConn)
MyCommand.Parameters.Add(New SqlParameter("@givenUsername", SqlDbType.VarChar))
MyCommand.Parameters("@givenUsername").Value = UserID
MyCommand.Parameters.Add(New SqlParameter("@givenPassword", SqlDbType.NVarChar, 10))
MyCommand.Parameters("@givenPassword").Value = Pass

uu = MyCommand.ExecuteReader()
If Not uu.Read() Then
    Return False
Else
    Return True
End If

```

End Function

```

Public Function EncryptPassword(ByVal Password As String) As String
    Dim PassNew As String
    Dim Cntr As Integer
    Cntr = 1
    Do While Cntr <= Len(Password)
        PassNew = CStr(PassNew) & Chr((Asc(Mid(Trim(Password), Cntr, 1)) + 20))
        Cntr = Cntr + 1
    Loop
    EncryptPassword = PassNew

```

End Function

```

Private Sub LoginForm1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
MyBase.Load

```

Try

```

    Dim objServerDateTime As New ServerDateTime
    objServerDateTime.GetServerDateTime()

```

Catch ex As Exception

End Try

Try

```
Dim objServerDay As New ServerDay  
objServerDay.GetServerDay()
```

Catch ex As Exception

End Try

Try

```
Dim objServerMonth As New ServerMonth  
objServerMonth.GetServerMonth()
```

Catch ex As Exception

End Try

Try

```
Dim objServerDateTimeAnother As New ServerDateTimeAnotherFormat  
objServerDateTimeAnother.GetServerDateTimeAnother()
```

Catch ex As Exception

End Try

Try

```
Dim objServerDateTimeWithSeconds As New ServerDateTimeWithSeconds  
objServerDateTimeWithSeconds.GetServerDateTimeWithSeconds()
```

Catch ex As Exception

End Try

End Sub

End Class

(02)Reporting Code

```
Imports System.Data.SqlClient
Imports System.Data
Imports CrystalDecisions.CrystalReports.Engine
Imports CrystalDecisions.Shared
Imports System.Configuration
Imports System.Text
Public Class Receipt

Private Sub Receipt_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
    Dim myConn As SqlConnection
    Dim strSql As New StringBuilder
    Dim oTable As New DataTable()
    Dim crReportDocument As New ReportDocument

    Label2.Text = LoginForm1.ComboBox1.Text

    myConn = New SqlConnection("Data Source=" & LoginForm1.TextBox1.Text & ";Initial Catalog=" &
Label2.Text & "; User ID=sa; Password=sic1almao;")
    Dim strReportPath As String = Application.StartupPath & "\ & "CrystalReport5" & ".rpt"
    crReportDocument.Load(strReportPath)
    myConn.Open()


    strSql.Append("SELECT Policy.PolicyNo,Policy.Premium,Policy.NextDueDate,Temp.UserID, Temp.Terminal,
Temp.SocietyCode, Temp.TranAmount, Customer.SFirstName, Customer.SLastName, Customer.SInitials,
Customer.SAddress1, Customer.SAddress2, Customer.SAddress3, Customer.SAddress4, Temp.FromDate,
Temp.No1, Temp.No2, Temp.CF, Temp.Total, Temp.Deduct, Temp.BF, Temp.NoOfAreasPremium,
Temp.NoOfPremiums, Temp.TranNo, Temp.TranDate, Temp.TranTime, Temp.AgencyCode FROM Customer
INNER JOIN (Policy INNER JOIN Temp ON Policy.LastTranNo = Temp.TranNo) ON Customer.CustomerNo =
Policy.CustomerNo WHERE Policy.PolicyNo=" & TextBox1.Text & """)

    Dim oDA As New SqlDataAdapter(strSql.ToString, myConn)
    oDA.Fill(oTable)
    crReportDocument.SetDataSource(oTable)
    CrystalReportViewer1.ReportSource = crReportDocument

End Sub
End Class
```


Appendix G – My Service Letter

Figure H.1 shows My Service Letter.



සීමාසහිත සුඤ්ඤ රක්ෂණ සමාගම
வரையறுக்கப்பட்ட சனாச காப்புறுதி கம்பனி
SANASA INSURANCE COMPANY LIMITED
Reg. No. : PB 379

No. 340, 2/1 R. A. De Mel Mawatha, Colombo - 03, Sri Lanka.
Tel: +94(0)11 200 2021 Fax : +94(0)11 200 2022
E-mail: info@sicl.lk Web: www.sicl.lk

19th May 2017

TO WHOM IT MAY CONCERN

This is to certify that Mr.R.M.N.P Bandara Rathnayake (742380513V) of No. "Nirosha" Uda Hewapola, Pillessa was in our employment from 17th of September 2007 to 31st of March 2016 .


The details of his employment are as follows.

Position held	Department / Branch	Period of Service
Software Engineer	Information Technology Department	17-09-2007 to 2015-09-30
Senior Executive	Kurunagala Branch	01-10-2015 to 31-03-2016

We wish him every success in his future endeavors.

This letter is issued at the request of Mr. R.M.N.P Bandara.

SANASA INSURANCE COMPANY LIMITED (GENERAL DIVISION)


W.L. Madusanka
Head of Human Resources
Sanasa Insurance Company, Ltd.
No. 340 2/1, R. A. De Mel Mawatha
Colombo - 03.

W.L. Widanagamage
Head of Human Resources

සීමාසහිත සුඤ්ඤ රක්ෂණ සමාගම உண்மையான காப்புறுதி பதத்திற்கு True Meaning of Protection

Figure H.1 My Service Letter

Appendix H – Client Certificate

Figure H.2 shows client certificate which is presented by ALMAO for Sanasa Insurance Co.

සමූහ සී/ස සමස්ත ලංකා අනන්‍යතා ආරක්ෂණ සංවිධානය
අලමාණි ALMAO

අංක : 40, කරඬුපහ, කෑගල්ල.
 දුරකථන : 035 - 3352155 / 035 - 2222758

කළමනාකරුගේ,
 තොරතුරු තාක්ෂණ අංශය,
 සී/ස සමස්ත රක්ෂණ සමාගම,
 අංක 340 2/1,
 ආර්.පී. ද මෙල් මාවත,
 කොළඹ 03.


2012. 05. 30

මහත්මයාණෙනි,

අනේකුක විපත් වැඩිපටහන සඳහා නව පරිගණක පද්ධතියක් යොදා ගැනීම

ඉහත කරුණට අදාළව නිර්මාණය කළ නව පරිගණක පද්ධතිය, මේ වනවිට පරීක්ෂණ මට්ටමෙන් ක්‍රියාත්මක කර ඇත. ඒ අනුව එමගින් සියලුම කටයුතු (හිල්පත් නිකුත් කිරීම, හිමිකම් ගෙවීම් සම්බන්ධ වාර්තා ලබා ගැනීම) සාර්ථකව ඉටු කළ හැකි බව අනාවරණය විය. හවුලටත් මෙම කාර්යය ඉටුකරගෙන යාමේදී ගැටළු ඇති වුවහොත්, තොරතුරු තාක්ෂණ අංශයේ සහාය ලබාගත හැකි බව නුදුරින් රත්නයක මගඟ පවසයි.

ස්තූතියි,
 මෙය විශ්වාසී,


 කළමනාකරු
 ජේ.කේ.එස්.ජයලත්
 තනතුර -

නම

අත්සන

1) එම්. දිසා වාන්දනි මෙනවිය - හිමිකම් ගෙවීම් පරීක්ෂා කිරීම
 2) ජී.ඒ.සී.එන්. කුමාරි ගුරුසිංහ - වාර්තා ලදුපත් නිකුත් කිරීම
 3) ආර්.පී. වන්දලඟා මිය - වාර්තා ලදුපත් නිකුත් කිරීම

Figure H.2 Client Certificate

Glossary

Database – Collection of data

RUP – Stands for Rational Unified Process. Iterative software development methodology developed by Rational Software Co-operation.

SQL – Stands for Structured Query Language. Help to retrieve data base details.

UML – Stands for Unified Modeling Language developed by Ivar Jacobson, James Rumbaugh and Grady Booch at Rational Software. It is a modeling language.

Use Case Diagram – Usecase diagram represent the how users interact with the system.

Framework – It is a set of class library which provide functionalities.

Index

A

Access iii,1,3,35,43

Agent iii,vi,xi,6,56,57

ALMAOi,iii,iv,xiv,1,2,7,25,26,27,28,30,31,35,81,82,85

B

Business1,4,7,8

Button27,28,30,47,48,50,51,54,55,56,57,58,60,61,62,64,67,69

C

Claimiii,vi,x,xi,xii,xiii,6,16,17,27,28,30,49,58,63,64,65,66,67,74,75,76,78,79

Customer.....iii,vi,x,xi,xiii,4,6,8,12,15,22,25,27,28,31,35,47,48,49,54,57,58,85

Closex,xiii,21,26,30,78,80,81,8

D

Databaseiii,vii,x,1,3,11,23,24,25,35,36,37,43,46,47,51,88

Date26,28,29,30,31,81,83,84,85

Designvii,ix,5,10,11,12,23,24,33,37,46

Developiii,iv,vii,1,2,3,5,8,9,10,22,23,24,31,32,33,35,88

Diagramx,xi,4,5,10,11,46,47,48,49,50,88

F

Form

.....x,xi,4,5,12,13,14,5,16,17,18,19,20,21,24,25,26,27,28,29,30,31,44,51,52,54,55,56,57,58,59,60,61,62, 63,64,65,66,67,68,69,81,82,83,85

H

Hardwarevii,xiii,22,42,43

I

Insuranceiii,iv,1,2,7,35,43

L

Loginviii,x,xi,xiii,2,12,25,27,28,29,30,31,46,51,77,81,83,85

P

Paymentvi,xi,xiii,6,48,49,60,63,67,76,77

Policyvi,x,xi,xii,xiii,6,14,18,21,27,28,30,31,47,48,49,57,58,60,62,64,65,66,73,77,78,,79,80,85

Premiumiii,vi,x,xi,xiii,6,13,31,48,59,60,70,77,85,

R

Reportsiii,vi,ix,x,xi,4,6,20,31,35,50,68,69

Requirementsvi,vii,xiii,4,5,6,8,9,10,22,23,33,35,42,43,

S

Societyiii,vi,xi,xii,6,31,47,54,55,56,57,58,72,73,85,

SQLiii,vii,x,1,2,3,22,23,24,25,26,27,28,29,30,31,39,43,44,81,82,83,85,88

