Abstract

Old Mutual Insurance System is a fully integrated browser-based insurance system that facilitates for faster interaction between potential clients, insured clients and the organisation. The system was developed with the clients at the heart of system and this provided an end-to-end policy administration system with a client-centric approach. The aim of the Insurance Administration System was to reduce application process and insurance claims approval process and address customer services effectively in a shorter time. The Insurance Administration System allows for clients to apply for insurance policies and claims online and also check their monthly premiums without visiting Old Mutual offices in person. Underwriters can view clients' documents online and approve their insurance policy applications and their claims applications within a short period of time. Underwriters can also update the clients' monthly premiums in their respective accounts. The Insurance Administration System was developed using PHP, MySQL and Dreamweaver. To obtain what the end user expected from the Insurance Administration System, the developer used interviews, questionnaires, joint application development and observation. The Insurance Administration System reduced costs, improved efficiency and enhanced controlled user access and authority limits. In the past, the insurance company had to manually maintain each customer records and this required a lot of man power and the amount of time spent after creating each record and updating them was huge. The introduction of the Insurance Administration System made it possible to manage all records with ease, eliminating errors in the process.

Declaration

I, Mihloro Mornisher do hereby declare that I am the sole author of this project, I authorise Midlands State University to lend this project to other institutions or individuals for the purpose of scholarly research.

Signature…………………………. Date …………………………….

Approval

This project entitled Old Mutual Insurance Administration System by Mihloro Mornisher meets the regulations governing the award of the degree of Information Systems Honours of the Midlands State University, and is approved for its contribution to knowledge and literary presentation.

Supervisor…………………………….

Date ………………………………….

Acknowledgements

In partial fulfilment of my degree programme, I would like to express my profound acknowledgement to the following people: The project supervisor Mr Zhou who contributed immensely through out the entire project development process, the management and staff of Old Mutual Private Limited Company for allowing me to carry out my research activities. l would also like to extend my sincere gratitude to my friends Faith Mazuzu, Tarisiro Urayayi and Charlton Gweremwezhe for their support during the research study. Last but not least l would like to thank my mother Mrs Mihloro and my sisters Sandra Mihloro and Shinnon Mihloro for their love and financial support and making me be where l am today. I could have never done it without them. I can safely say," May God continue to bless you".

Lastly l would like to thank the almighty God for the source of strength that he has given me that has brought the following project into existence.

Dedication

"This project is dedicated to my late father, Samson Mihloro who is my major source of inspiration. His loving memories always inspire me to be the best and to bring out the best in me. The greatest lesson that he taught me is that even the largest task can be accomplished if it is done one step at a time."

Table of Contents

List of Acronyms

CABS	Central African Building Society
CBA	Cost benefit analysis
DBMS	Database management system
DFD	Data flow diagram
ERD	Entity relationship diagram
EERD	Enhanced entity relationship diagram
IAS	Insurance administration system
ICT	Information communications and technology
JAD	Joint application development
OM	Old Mutual
OMIG	Old Mutual Investment group
OMLAC	Old Mutual life Assurance Company
ROI	Return on investment
ZSE	Zimbabwe stock exchange

List of Tables

List of Figures

List of Appendices

CHAPTER 1: INTRODUCTION

1.1 Introduction

Old Mutual Insurance Administration System is an online-based administration system that seeks to provide advertisement, efficiency, and control, reduce data redundancy and help management in decision making. It involves all decision makers in the organization, from top management to the shop floor subordinates. The system is designed to enable life insurance companies to effectively implement new insurance products, optimize distribution channels, automate business processes and respond promptly and flexibly to the demands of the quickly developing insurance market. The system is also designed to help in capturing, storing and reviewing customer information; assists marketing and after-sale activities. In this dynamic world and the competitive market growing every second it is vital for a company to have maximised productivity for day-to-day claims processing. The system will empower users by providing integrated client information at their fingertips and a user-friendly interface that support multi-tasking for optimal productivity.

1.2 Background of the study

The insurance industry is growing rapidly and the client base is fast growing hence, the developer chose to study the above mentioned topic because there is need to develop fast, reliable and efficient insurance systems to cope up with the fast growing client base. A web based insurance system that facilitates faster interaction between the client and the organisation is therefore a necessity. With the growing competitive market an organisation needs to be highly innovative and retain its customers for it to survive in the competitive market.

1.2.1 Background of the organisation

Old Mutual is an international long-term savings, protection and investment group, based in London, operating in 33 countries, and serving more than 15 million customers. The group has a reputation for integrity, financial strength and value-for-money and is dedicated to satisfying client needs for wealth creation and protection - in profitable ways. Old Mutual has solid foundations established on one core promise; the commitment to deliver value always to its clients. It has at its core the drive to create, manage and preserve wealth for its millions of clients. It is dedicated to satisfying customer needs and develops products in response to clients changing needs. John Fairbain founded the Mutual Life Assurance Society in the Cape of Good Hope in 1845. The society had no initial capital and was financed solely from the funds of the society's 166 members. Its main objective was to mobilise funds to provide cover to its policyholders. The name Old Mutual was born in 1883 when another mutual society calling itself Colonial Mutual Life Assurance society came on board in South Africa. In order to differentiate the company from the new player a rebranding was done and the name OLD MUTUAL was born. OLD MUTUAL opened its doors for business in Salisbury (now Harare) in 1927. The company grew in leaps and bounces and on 12 July 1999 the company successfully completed its demutualisation exercise marking the birth of the present day Old Mutual Life Assurance Company Limited.

1.2.2 Organisational structure

In order to serve all market niches Old Mutual is strategically structured as a financial powerhouse. It has concerns in every sector of the financial services industry. Old Mutual's principal business comprises of life insurance, asset management, banking and general insurance. The diagrammatic representation of the organisational structure is shown in Figure 1.1 overleaf. It shows the different subsidiaries that are under Old Mutual Zimbabwe Ltd. Old Mutual life assurance business is conducted through OMLAC, which provides life, disability, retirement savings and investment.

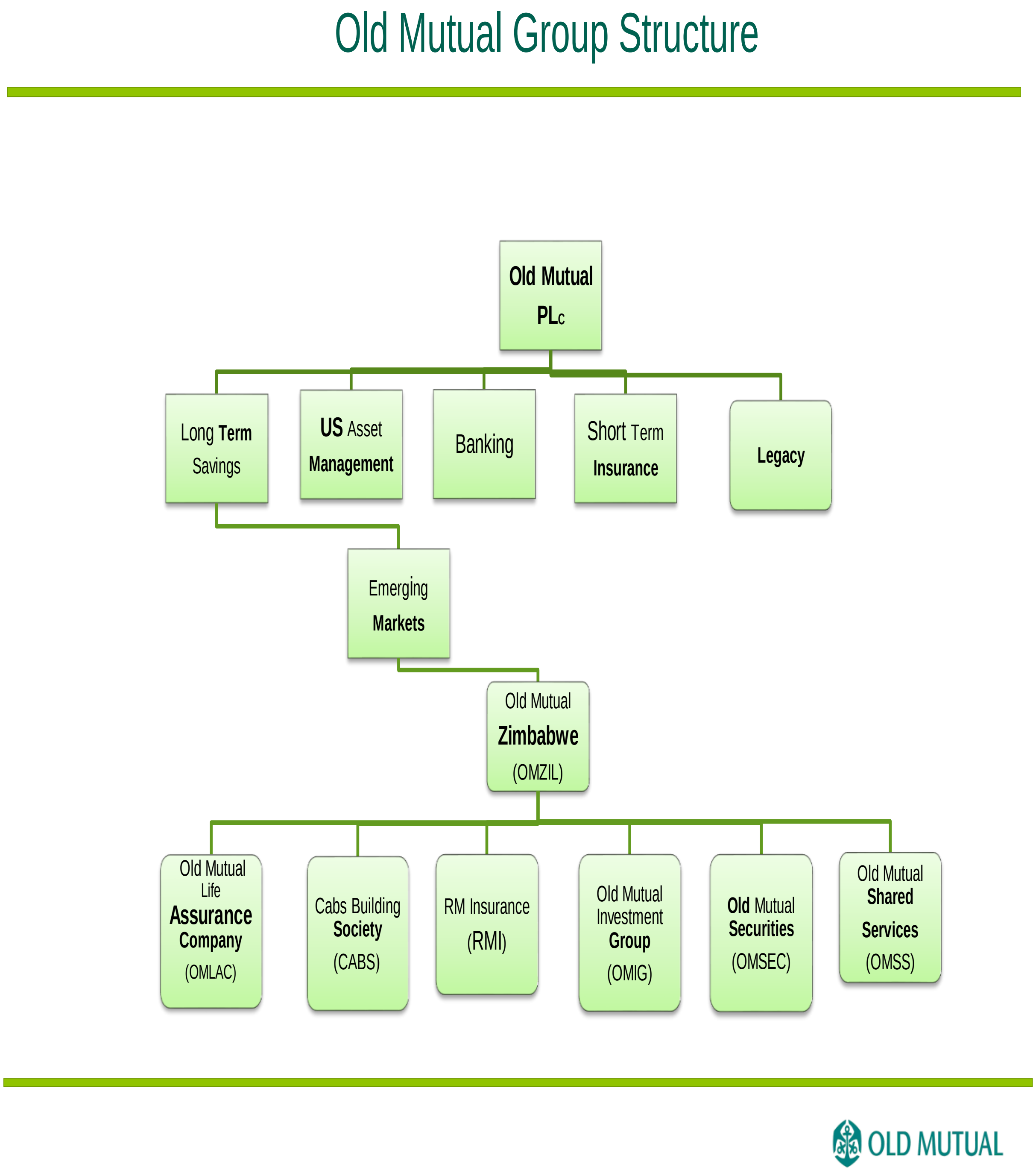

Old Mutual Group Structure
Old Mutual PLc
Long Term Savings
US Asset Management
Banking
Short Term Insurance
Legacy
Emerging Markets
Old Mutual Zimbabwe (OMZIL)
Old Mutual Life Assurance Company (OMLAC)
Cabs Building Society (CABS)
RM Insurance (RMI)
Old Mutual Investment Group (OMIG)
Old Mutual Securities (OMSEC)
Old Mutual Shared Services (OMSS)
OLD MUTUAL

Figure 1.1 Old Mutual Organisational Structure

1.2.3 Vision

To become the first choice provider of financial services in Zimbabwe as we help our customers build wealth and achieve their lifetime financial goals, while enriching the lives of our employees and the people we serve as well as enhancing stakeholder value.

1.2.4 Mission statement

Old Mutual is an international financial services group with a reputation for integrity, financial strength and value-for-money. The Old Mutual Group is dedicated to satisfying client needs for wealth creation and protection - in profitable ways. The company serves clients individually and in groups through a variety of access channels recognising its client's preferences. Old Mutual develops its products and services in response to client's changing needs and complements them through co-operation with other organisations. The Group operates through a number of financial services businesses. Each has its own focus and is accountable for profitability and growth. The Group provides a powerful driving force behind the development of business that anticipates future client needs and competitive trends. The company plans to extend its market leadership in the rapidly evolving international financial services industry. This requires talented, skilled and committed people. It offers its entire people opportunities for personal growth and achievement. Through its business activities are an active force in socio-economic development. The culture challenge the company faces is to grow its businesses profitably whilst acting with integrity and delivering value to its clients and in this way Old Mutual live up to the trust clients place in Old Mutual.

1.3 Problem definition

The problems that Old Mutual is facing include the following:

1) System inflexibility as clients can only access their balances and statements on a working day, during working hours. Claims can only be done during the week and in some cases if it’s a hospital insurance claim and the client needs money

urgently she/he has to wait till Monday. This result in work overload during Mondays and overloading the system may slow the performance of the system.

2) Time Consuming as clients wait for a long time before they can be given their account balances and statements and also the client has to travel to the nearest Old Mutual branch just to make a claim or to increase the scope of the insurance.

3) If a client makes a claim she or he has to go through a long process for the claim to be approved and signed. The client has to move with his or her documents from one office to the other having the documents signed by different personnel.

4) Standardised reports cannot be produced because each insurance facet has its own data mart. The life assurance has its own data mart and the funeral policy has its data mart and this makes it hard for comprehensive reports to be made.

5) The systems are not integrated and this is causing employees to be inefficient because they cannot do multi tasking. An insurance administrator can only perform one task at a time. If the administrator is an underwriter then it means she or he cannot process claims but only stands for underwriting.

6) The current system does not provide history of claims made by the client and this has caused emergence of fraud.

7) There is data redundancy as each insurance facet keeps its own set of data. One client my have short term insurance, funeral policy and life assurance and because these three insurance facets do not share information then it means they all store the details of the client leading to data redundancy.

8) The current system was built to be policy-centric, not client centric. The system does not provide an end to end policy administration with a client centric approach. There is no relationship between Old Mutual and its clients and this has made customer retention very difficult.

9) Document management with the current system is not secure. All the clients' documents for example birth certificates are stored in cabinets which can be prone to theft and natural disasters without backups. So there is need to automate the document management systems as this increases security and it has the option of backing up the documents.

1.4 Aim

To develop a fully-integrated browser-based insurance policy administration solution that helps manage the entire policy lifecycle for all lines of business, available 24/7 and can be accessed remotely and also allow every authorised member of the organisation to quickly view, schedule and execute everyday tasks

1.5 Objectives

The objectives of the Old Mutual Insurance Administration system include the following:

I. Central repository for client communication, information and document management that will facilitate customer relationship management.
II. Comprehensive process for managing the claims workflow. This means that when a client wants to make a claim he/ she does not have to visit every office for approval and the signing of the paperwork.
III. Electronic approval process based on user authority limits.
IV. To provide a comprehensive set of standard reports, summarised and detailed, for all facets of insurance with real-time data to enable superior performance analysis and better decision making throughout the organisation.
V. The system will provide automatic generation of claims documents to a claim. This will reduce the response time that is needed before a claim can be completed.
VI. A platform that enables clients to view their monthly premiums.

1.6 Instruments

To appropriately implement this system, the developer selected the following tools that were incorporated in the development of the system.

I. PHP

PHP is a powerful open source, server-side scripting language. The developer chose to work with PHP for the following reasons:

Extensions – the source code of PHP can be modified to include custom created extensions and components thereby increasing its extensibility. A number of libraries and extensions to extend its core functionalities are available for download.

Cross Platform – PHP provides high compatibility with leading Operating Systems and web servers thereby enabling it to be easily deployed across several different platforms. This means that for Old Mutual PHP can run on computers with Windows 7 or some with Windows XP.

Security – PHP offers security that helps prevent malicious attacks. This means that the system will be less vulnerable to virus attacks.

II. MySQL

A database is a mechanism that is used to store information or data (Stephens, 2001). The developer chose MySQL as the database platform. MySQL is a scalable relational database with many advanced features and options. It is fast, reliable and cost effective to other databases like Oracle.

Reasons for choosing MySQL:

Cross Platform Operability – It is installable and operable on different platforms including Windows and Linux. The developer can easily connect with different platforms as well and integrate applications developed in different OS and development platforms.

Security – MySQL is secure as all access passwords are stored in an encrypted format restricting and unauthorised access to the system.

Fast development – it has a wider and faster development circle. Patches, upgrades and fixes are developed fast and become available in forums, blogs and sites sit on the internet.

III. Dreamweaver

It is a flexible, simple RAD tool which enables designing graphical web based user interface. Reasons why the developer chose to work with Dreamweaver are as follows:

a) Dreamweaver is customisable: The user may set preferences controlling how and to what extent accessibility is coded. The user can adjust code coloring, what fonts are used for coding and how code is highlighted.

b) Building and editing websites is fast: The Graphical User Interface allows for simultaneous designing and coding. The user can see what their html looks like immediately after writing a code.

c) Basic image optimization may be accomplished and more advanced image processing may be accomplished through the close integration between Dreamweaver and fireworks.

1.7 Justification

Old Mutual can gain a lot of benefits from the implementation of the web based insurance administration system. The system is designed to help agents provide quotes, manage policies and follow up with their clients when necessary. Not only does it provide the information tracking required to manage short and long term insurance policies, it also provides a method of receiving and billing policy payments and payouts. The system will provide up to date information and quick answers to clients. Some of the benefits that will accrue to Old Mutual are outlined on the next page.

1. **Higher quality for customer service** - Support for the sales process through merging with solutions existing in the company such as CRM and the automation of manual processes and applications have a significant impact on increasing the level of satisfaction of customers and insurance partners.
2. **Competitive Growth**- Reducing costs is possible through a better organisational flow of information, automation of processes and the ability to exploit new sales channels through the Internet and hence Old Mutual will gain a competitive advantage over companies like Altfin and Fidelity Life Assurance
3. **Improved reports** - Reporting the overall portfolio of insurance with regard to insurance, financial and statistical data is a key factor in the management of an insurance company. The system provides reports for internal and compulsory reporting, shareholders and stock exchanges.
4. **Reduced operating costs** - One repository of data provides information to eligible departments. Increasing the scope of the data exchanged and the ability to obtain additional feedback without the involvement of employees, increases job satisfaction, which as a result leads to improved efficiency and better customer service
5. **Achieve greater consistency and improve accountability** - in key marketing, CRM, underwriting, claims processing and cash disbursement activities. Reduce errors, training time for new employees and expense ratios. Increase efficiency, productivity and customer service.
6. Transactions automatically flow through to the accounts of reinsures of ceded and recovery reinsurance transactions as well as receivables and payables
7. Online standard reports available with the click of a mouse
8. Flexibility and speed to market, enabling the quick launch of new products, channels and quick adaption to the changing regulatory environment

1.8 Conclusion

With the information at hand, it became a necessity to take positive steps. The developer made a plan towards the development of a computer based automated system that will address the highlighted problems. For a system to be successfully developed and delivered on time there was need for careful planning, identification of the resources needed, project duration and analysis of its feasibility. Finally, the management needs an efficient and effective tool to make certain strategic and operational decisions that will enable the organisation to gain a competitive advantage, and be able to offer better services to its customers and other stakeholders. The preliminary stage was completed and the developer proceeded to the planning phase.

CHAPTER 2: PLANNING PHASE

2.1 Introduction

According to Kendall (1995), the purpose of the planning phase is to perform a preliminary investigation to evaluate an IT related business opportunity or problem. The preliminary examination is a critical step because the outcome will affect the entire development process. The Planning phase will determine how the project team will go about building the information system. This chapter will also establish how the system to be developed will benefit the organisation at all levels (operational, technical, social and economic) i.e. the expected business value to the organisation. Included in this chapter, is the feasibility study that evaluated whether it is viable to develop the Old Mutual Insurance Administration System. Risk analysis was also considered in this chapter because any project is vulnerable to risks.

2.2 Reasons for building the system

Web based Insurance software solution is a now a necessity to Old Mutual because of its rapidly expanding client base not just vertically in the society but also horizontally across the globe. The business has become far more complex than it has ever been before. With the increase in awareness and literacy rate across the country, the customers for various insurances are on the constant rise. The various departments, agents, collection centers now require instant connection to feed this demand. On the top of it global operations make it mandatory that the database is updated for every client across the world at the same time. The proposed administration system will allow Old Mutual to provide innovative and cost effective solutions to the individuals, while catering to a wide range of insurance needs.

Currently Old Mutual has to manually maintain each customer's record and this requires a lot of man power and the amount of time spent after creating each record and updating them is huge. The introduction of the smart and advanced administration system will make it possible to manage all records with ease, eliminating any errors in the process. The proposed system will greatly help the insurance company improve its cost effectiveness, cut down the operational expenses, and provide better customer service.

The proposed system is designed to connect the various departments of the business together - from marketing to claims. This way the moment a new customer signs up, the details of the policy are fed into the system, payment details immediately go into accounts, premium durations and period of payment are all related to customer service, all at one go. Any change in the account is immediately reflected in all departments. Therefore in the event of claims the data is ready and available for processing and payment is faster and service more effective.

Introduction of insurance technology in Old Mutual will improve every aspect of the company. The technology will play a major role in data management process of an insurance agency by providing flawless services from underwriting policies, producing documents to collating various ratings and data. The insurance administration system will help the insurance agents to immediately respond to the requirements of the customers and manage to cut back the annual expenditure of the organization. The basic purpose of the insurance administration system is to reduce the process of proposals and policies and address the customer services effectively in a shorter time than any other traditional methods. Web based Information technology in insurance will make it easier for the customers too. Online availability of the insurance agencies allow the clients in dealing with application procedures, signing proposals and policies as well as in receiving quotes without even visiting the insurance office in person.

A web based insurance administration system will help Old Mutual in reducing the costs by eliminating the mail rooms, paper files as well as the data entry clerks. The elaborate underwriting, data processing and the rating take place online and the customers or brokers receive the emailed policy documents within no time. With all data online and on screen the need for cumbersome paperwork will really be diminished. Only the most essential documents will be filed manually and in paper while all others will be electronically stored. This just does not mean less paperwork, but also less manual error, precise information transfer and better back up facilities for the numerous valuable data without which the industry would come to a halt.

Management of claims is another significant aspect of the insurance automation. There is specific insurance software for managing claims and quick settlement of dues. The software works just like claim managers to assess, evaluate the real value and approve the settled amount. The software will make it possible to have settlement of multiple claims. What had to be operated through multiple desks involving different issues of relevance will be found in a convergent solution in insurance automation. The insurance policy administration system will facilitate the shopping policies for different types of coverage. The proposed system is vital to Old Mutual since it will reduce the cost of errors and manual work.

Market penetration- This is achieved by using the power of the Internet to advertise and increase awareness of products and also to lift the profile of a company amongst potential customers in an existing market. It also assists financial institutions in wining new customers and retaining existing customers through improved cross selling opportunities and new customer acquisition.

Market development – a web based insurance system improves sales through automated value-added advice, product modelling, cross-selling and leads qualification tools that earn the customer's trust by advocating the best products to fulfill customer needs.

Product development- New Internet-based products or services are being developed which are typically information oriented, such as market reports that can be purchased using electronic commerce.

Diversification-In this sector, new products are developed which are sold into new markets. Insurance Administration System empowers consumers with knowledge, builds trusts with prospects and customises advice to suit consumer's needs.

2.3 Business value

The process of identifying business value is of great significance because it clarifies some of the potentials, which may accrue by adopting the insurance administration system. The business value refers to the success of the organisation in using information to achieve its strategic objectives (Purba, 1995). By developing the proposed system, there will be reduction in quite a number of problems, which are currently faced by the organisation. The system should offer a competitive advantage against its rivals. The system should increase the customer/client base. The organisation should not incur losses when it adopts the new system. Therefore, the new system is supposed to increase efficiency and effectiveness of the business operations. The system is expected to help the organisation realise its main goal of being the leading insurance provider in the country by offering better Internet based insurance services. The main strategic areas of benefit are:

Organizational value: conceptually, the organisation will be viewed as a technological reputable company and thereby improving its image.

Managerial value: it will be seen as a management tool, which will help them to move towards offering quality services to its clients and other stakeholders at large.

Customers' value: they will view it as a system that will improve the organisations' efficiency and effectiveness of both the management and staff at large. It will assure them of a better service and maintain customer relations with their service provider.

Operational value: the system will reduce the time taken to service a client and also on response time to the clients' claims.

Employees' value: it will help in retaining effective employees as well as controlling their activities especially for financial gain.

Security value: the company will not be prejudiced of any funds through fraudulent activities. Also data and information will be kept secure, that is, no unauthorised access to data will be allowed. Moreover, it also ensures data integrity and control as well keeping up-to-date information.

The website to be developed is expected to derive a number of benefits to Old Mutual thereby maximising on client base hence increasing its market share. Improvement on management oriented reports and timely report production will aid decision making process thus helping in the attainment of organisational stated goals. Efficiency is improved since data can be quickly retrieved when required due to its availability. The use of centralised database for data storage reduces paperwork and data redundancy which improves system efficiency and effectiveness. The new system enhances sudden customer care, good relations, satisfaction and retention since clients' matters will be responded to quickly. Eventually this will give competitive edge and improved organisational standards and professionalism. It is also going to ensure the organisations` image as being a modern organisation using modern technology.

2.4 Feasibility study

The main purpose of feasibility study is to study the feasibility of e-insurance services in Old Mutual Insurance Company. In other words this phase seeks to investigate the state of being feasible of presenting insurance services electronically. Feasibility study seeks to address the question: "Is it feasible or viable to carry out the project taking into consideration the available resources including the time constraint? (Purba, 1995). A key part of the preliminary investigation is a feasibility study that reviews anticipated costs and benefits and recommends a course of action based on operational, technical, economic, and time factors. For the system to be considered feasible; the anticipated benefits should outweigh the overall costs (development and operational costs). Feasibility study is categorised into: technical, economical, social and operational feasibility. The developer worked with the representatives from departments who are expected to benefit from the proposed system.

Benefits of a feasibility study:

Increases chances of the project being a success: The study will identify the hard parts of the project. It will help identify dependencies in the planning i.e. trials that need to be done/questions that need to be answered before key decisions on eradication design can be made. This will allow time for the developer to address all of the issues before the operation starts. This will reduce project risk and the likelihood of surprises later in the project. Issues can be planned for and dealt with.

Informed decision making: Feasibility study manages stakeholder expectations about how much the project will cost and how long it will take. It also ensures the decision and commitment of time and money to the project is based on accurate information, and also it enables adequate resources and timeframes to solve issues.

Reduces wastage of money: Projects that are too difficult will be stopped early, rather than later when more money has been spent or before expectations are raised too far. Concluding that a project is not feasible is not a bad outcome, as it: avoids wasting time and money on a project that would later fail and enables the developer to identify what extra work needs to be undertaken to make it feasible.

Having identified the need for carrying out Feasibility study, the developer then examined different types of feasibility.

2.4.1 Technical feasibility

Technical feasibility analysis analyses the availability of technology and resources (Overton, 2007). Questions that were answered during technical feasibility analysis included the following:

I. **Is the proposed technology or solution practical?**

- Does the company currently possess the necessary technology?

- Does the company possess the necessary technical expertise and is the schedule reasonable for this team?

- Is relevant technology mature enough to be easily applied to the problem?

II. What kinds of technology will be needed?

III. Is the required technology available “in house”?

- If the technology is available does it have the capacity to handle the solution? If the technology is not available can it be acquired?

Some of the computer hardware and software needed for the implementation of the system are already available in the organisation. However, there is need to acquire a state of the art Dell Power Edge R410 server to host the system, six additional HP Compaq 500B Personal Computers and Macromedia 8 software package, a requirement that can easily be met considering the amount of the funds that have been set aside for the system development.

Software Availability– the crucial software required to run this project includes Mozilla Firefox Browser ESXi Installer, VMware viclient, Wamp Server, Macromedia 8.0 and Windows XP professional. All these software except the operating system are open source offered free of charge for developers to use. Hence, no license fees to be worried about. A network is already present that links Old Mutual branches together.

Table 2.1 Hardware and Software requirements

Quantity	Item
1	Database server (Dell Power Edge R410)
6	Underwriters Computers (HP Compaq 500B)
50	Tape drive 20/40
	Networking material (Patch Panel- 24Port, 20m UTP CAT 35 Fly Leads patch codes-RJ 45)
	Software packages (Macromedia 8, Mozilla firefox ESXi Installer, VMware viclient, Wamp Server)

Table 2.2 Database Server (Dell Power Edge 1600s) Specifications

ITEM	MINIMUM	RECOMMENDED
Processor	2.4GHz	2.6GHz Intel P4
Cache Processor	256Kb	512Kb
Memory	2000MB	3000MB
Hard Disk Drive	135GB	200GB
Network Card 3com	10/100	10/100
DVD/CD r/w Drive	48x Read,12x Write	52xRead, 24x Write

Table 2.3 Client Computer (HP Compaq 500B) Specifications

ITEM	MINIMUM	RECOMMENDED
Processor	1.79GHz	2.4GHz Intel P4
Memory	2000MB	3000MB
HDD	111GB	150 GB
Network Cards	10/100	10/100

Technical Expertise

This study also looked at the capabilities of the personnel that will develop, maintain and support the project directly or indirectly. This helped in determining whether to implement the project or not. Personnel that are present at Old Mutual and their respective responsibilities include:

Business analyst: Emphasise on the business issues addressed by the system: value of new system; identification of problems and opportunities; revision of business processes and policies.

Systems analyst: Emphasise on Information System issues of the system: how Information Technology can be used to support business processes; design of new business process and Information Systems; and enforcement of Information Systems standards.

Infrastructure analyst: Technical issues associated with integrating new system components to existing technical infrastructure. Change management analyst emphasise on facilitating organisational adaptation to new system. Helping to identify and overcome resistance to change and assuring adequate training and documentation of new system.

Project manager: Ensuring that progress is made on the project: time schedules and budgets are met; supervision of project team; and manage relations with project sponsor and users**.**

The Information Technology team at Old Mutual is effective, the team is proficient and experienced and they will be able to implement and maintain the system at very low costs. The developers of the system are competent to produce the proposed system. Some of the organisational staff have minimum required technical skills to operate the proposed system. No external employees will be needed for the implementation of the system. The development team has comprehensive knowledge and appreciation of the development software, (PHP, MySQL and Visual Studio) to be used, thus reducing the risk of project failure.

Evaluation: Upon completion of the technical feasibility, the developer concluded that since Old Mutual is able to acquire hardware and upgrade some of the hardware and also that the technical team is very competitive, so the proposed system is technically feasible. Once the technical feasibility is established, it is important to consider the monetary factors also. Since it might happen that developing a particular system may be technically possible but it may require huge investments and benefits may be less. For evaluating this, the developer embarked on economic feasibility of the proposed system.

2.4.2 Economic feasibility

Economic analysis is the most frequently used technique for evaluating the effectiveness of a proposed system. It is also known as Cost / Benefit analysis, the procedure is to determine the benefits and savings that are expected from a proposed system and compare them with costs, (Kerzner, 2011). If benefits outweigh costs, a decision is taken to design and implement the system. Otherwise, further justification or alternative in the proposed system will have to be made if it is to have a chance of being approved, (Overton, 2007). This is an outgoing effort that improves in accuracy at each phase of the system life cycle. The feasibility study presents both tangible (e.g., increased productivity, low operating cost, etc.) and intangible benefits (e.g., improved organizational planning, improved asset utilization, etc.) in a formal way.

a) Development costs - Development costs are incurred during the development of the system, are one time investment. Development costs for the proposed system include acquisition of the state of the art database server Dell Power Edge R410 and additional underwriters' computers HP Compaq.

b) Operating costs **-** Operating costs are the expenses required for the day to day running of the system. This includes the maintenance of the system. Operating costs for the proposed system include stationery, salaries and wages, maintenance of computers and the database server, computer consumables and user training.

An illustration of the costs for the Insurance Administration System breakdown is shown in Table 2.4 overleaf.

Table 2.4 Development Costs Table

Personnel		$
1	Business Analyst (56 days @$30/day)	1 680
1	System Analyst (40 days @$25/day)	1 000
3	Programmers (60 days @$30/day)	5 400
2	Infrastructure Analyst (20 days @$25/day)	1 000
1	Database Specialist (40 days @$25/day)	1 000
1	Project Manager (50 days @$30/day)	1 500
Projected New hardware and software		
2	Database server (Dell Power Edge R410)	20 000
6	HP Compaq 500B	4 000
4	Tape drives	20 000
3*30m	RJ 45 Network Cable	500
Total development costs		**56 080**
Projected Annual operating costs		
	Maintenance of database server	5 000
	Stationery	1 500
	Computer consumables	3 000
	Labour	15 000
	Maintenance of computers	5 000
	User training	4 000
	Fees	2 000
Total projected annual operating costs		**35 500**

Benefits

A benefit is the excess of income over expenses (Layard, 1999). Benefits include increasing income, decreasing costs. Benefits can be tangible or intangible, direct or indirect. Outlined below is a table of intangible and tangible benefits of the proposed system for 3 years.

Table 2.5 Tangible and Intangible benefits Table

Benefit	2013	2014	2015
Tangible Benefits			
Recovered profit	50 000	80 000	120 000
Labour Cost Reduction	15 000	10 000	15 000
Increased productivity	10 000	20 000	25 000
Increased customer base	15 000	25 000	30 000
Intangible Benefits			
Increased employee morale	7 000	10 000	12 000
Improved asset utilisation	5 500	8 000	8 500
Estimated total benefits	**102 500**	**153 000**	**210 500**

After having identified the cost and the benefits to the proposed system, a cost benefit analysis was then conducted.

Cost benefit analysis

A CBA compares costs and benefits using equal terms. It provides a clear indication of net cost or benefit of a specific area or regulation, helping justify decisions at various levels and it also simplifies complex concepts and processes (Whitten et al, 1986). The purpose of a cost/benefit analysis is to answer questions such as: Is the project justified (benefits outweigh costs)? Can the project be done, within cost constraints? What is the minimal cost to attain a certain system? Cost benefit analysis (CBA) is a technique for assessing the monetary social costs and benefits of a capital investment project over a given time period (Demarco, 1979).

Table 2.6 Cost Benefit Analysis Table

	$	**$**	**$**
	2013	**2014**	**2015**
BENEFITS			
Estimated annual benefits	102 500	153 000	210 500
COSTS			
Total development costs	56 080	-	-
Total projected annual operating costs	35 500	35 500	35 500
Increase in maintenance costs	-	5 000	15 000
Increase in computer consumables	-	400	500
Net benefit / loss	**10 920**	**112 100**	**159 500**

With reference to the above CBA table the project was considered to be viable since the benefits for the three years exceed the corresponding costs.

Payback period

It is time taken to break even or pay back the initial investment (Layard, 1999). The investment with the shorter payback period is recommended because a shorter payback period is viewed as less risky; the longer the payback period, the more uncertain are the positive return (Stickney, 2000). Projects with the shortest payback period will be chosen.

Advantage - it is easy to compute

Disadvantage - it does not take into account the time value of money

Calculation:

Year 0 91 580/ 102 500*12

= **10 months**

Evaluation: It takes 10 months for the project to pay back the initial capital. From the above calculation, it has been reflected that all the costs to be incurred will be returned back in a period of less than a year (10 months). The developer concluded that the project is economically viable.

Return on Investment (R.O.I)

ROI can be used to compare the net profit against the investment required. ROI is used to calculate the viability of the project (Stickney, 2000). It is the most widely used cost benefit analysis technique and is calculated using the percentage of profitability.

$$\text{ROI} = \frac{\text{Project Outputs} - \text{Project Inputs}}{\text{Project Inputs}} \times 100\%.$$

Where the project outputs are all of the benefits of the project quantified in terms of cost savings and revenue generation, and the project inputs are all of the costs of the project including the operating costs (Stickney, 2000). A project with a positive ROI is recommended (Borman, 2011).

Advantage - It provides a simple way to calculate a measure of return on capital

Calculation

$$\frac{(102\,500-(56\,080+35\,500))}{(56\,080+35\,500)})\quad *100\%$$

10 920/91 580 *100% = **12%**

Evaluation: R.O.I is positive. Ratio of income generated to investment is sufficient and favorable. In relation to these anticipated benefits the project was given the priority to continue since the benefits to be projected by the project, exceeded the anticipated costs. Since the net benefit in the previous years were below par value as a result of operational intangible costs, the management and workers agreed to approve the proposed system for continuation.

2.4.3 Social feasibility

Social feasibility study is a process that provides a framework for prioritising, gathering, analysing, and incorporating social information and participation into the design and delivery of projects (Overton, 2007). It ensures that infrastructure project development is informed and takes into account the key relevant social issues, and incorporates a participation strategy for involving a wide range of stakeholders. This is a determination of whether a proposed project will be acceptable to the people or not. This determination typically examines the probability of the project being accepted by the group directly affected by the proposed system change. The system to be developed is not going to eliminate any human effort that is currently running the manual system. Instead the system will be there to lessen burden of the management.

The introduction of the new system does not seem to bring much 'earthquake' to the social structure of the firm neither does it compromise morals and the general organisational culture. The threatening area is that there might be staff downsizing since some manual tasks are now being automated thus the workers will be redundant. This may break work groups and social groups which in turn affect employee morale thus reducing worker productivity. Nevertheless this action reduces the firm's pay bill and the firm does not lose its structure.

Evaluation: Socially the system is feasible as there will be no loss of jobs, and the system will boost employee morale since all the employees are familiar with the Mozilla Firefox browser to be used and the platform on which it will be running thus Windows 7.

2.4.4 Operational feasibility

After the project has been deemed economically, socially and technically feasible, the next step was to determine whether it is operationally feasible or not. During operational feasibility study, it is determined whether the system will operate in the way that user wants. Operational feasibility is mainly concerned with issues like whether the system will be used if it is developed and implemented. Whether there will be resistance from users that will affect the possible application benefits? This is mainly related to human organisational and political aspects (Karolak, 1996). In exploring operational feasibility the developer used PIECES frame work. (Performance, Information, Economics, Control, Efficiency, Services).

PIECES FRAMEWORK

The PIECES framework is a checklist for identifying problems with an existing information system (King, 2000).

Performance: The proposed system will improve throughput, the amount of work performed over some period of time and also improve response time, the average delay between a transaction and a response to that transaction.

Information: The current system is facing some loopholes in terms of accessibility of information. Information about the same client is stored in different polices of insurance. There is no integration of the clients information with many insurance polices. Each insurance policy has its own database of clients and this leads to duplication of data since one client can have many policies. Information produced from the current system is not timely to its subsequent use. Information storage at the moment, all the registration and administrative processes are stored in file cabinets which means mostly all have access to it and may be misfile or pages of files can be missing. The system will allow sharing of information to all applications. All insurance policy applications and claims will make use of a single database, so information entered in one application is immediately updated across the whole platform. When a client is registered her information is immediately entered in the database and the proposed system makes use of a document management system.

Economics: There is need to reduce Economic or cost impacts. The current system at Old Mutual is resulting in excessive costs thus the proposed system will explore new markets and increase the company's clientele resulting in an increase in the organizational profits.

Control (and Security): The current system does not provide comprehensive support ad hoc querying, there for the new system will provide adequate response time

Efficiency: The new system will be faster and more accurate in provision of information and services

Service: A computerized system will have the ability to cope with large volumes of files and records.

To get to check on whether the relevant stakeholders are in support of the project, a random check using interviews and questionnaires when necessary was done. The responses obtained are:

Management (Chief Executives and Directors): Management is fully in support of the project and they promised to mobilise their subordinates to support the project in every way they can.

Management (Finance): Management in the finance department is more concerned about what the company gets from the project. They are not concerned about what it takes to make the project. After a serious look at the benefits and the cost benefit analysis the managers fully supported the project. They realised it was beneficial to the organisation.

Management (I.T.): they are confident the system will be the organisations' best product ever and they are confident they will manage to implement and maintain it well.

Users (Administrators): they are the ones who actually pushed and initiated the project since they were now overwhelmed by the clerical paper work they were handling in the traditional insurance system. They decided to look for another innovative way of reducing their workload. They are in full support of the project and they are actually mobilising every other worker to support the project.

Users (Clients): They are in full support of the project since they want to reduce costs in communication and transport. Most of the clients use the internet most of the time so it won't be very difficult for them to get used to the system.

Evaluation: From the random samples of interviews and questioners carried out, the developer concluded that the project is operationally feasible since all the stakeholders are putting all they can to make the development of the Insurance Administration System a success.

2.4.5 Schedule feasibility

A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. Schedule feasibility is a measure of how reasonable the project timetable is (Demarco, 1979). The deadline for the project is reasonable and it can be concluded that the project will be completed within the specified time.

2.4.6 Market research study & analysis

This is one of the most important sections of the feasibility study as it examines the marketability of the product or services and convinces readers that there is a potential market for the product or services (Karolak, 1996). If a significant market for the product or services cannot be established, then there is no project. The developer established the market and also through the website it will make it possible to market the product thus establishing an efficient Customer Relationship Management. The proposed system is marketable.

Overview of feasibility study

Technical feasibility was conducted and it was concluded that the project was technically feasible since most of the hardware requirements are affordable and also some of the hardware requirements are present at Old Mutual. Also the IT personnel at Old Mutual are competitive and can work towards the successful implementation of the project. Economic feasibility was done and it was concluded that the project would yield a profit within a short period of time of 10 months and thus the project is economically feasible. The proposed system will be completed within the specified given deadline and also it will be possible to market the product through the internet. After the completion of the feasibility study, the developer then identified risks of implementing the Insurance Administration System.

2.4.7 Risk analysis

This section considered all possible risks and setbacks likely to occur during the development of the project, the contingency measures to avoid these risks as well as trying to come up with counter measures for the risks if they occur.

Operational/Transaction Risk: arises from fraud, processing errors, system disruptions, and the inability to deliver products or services, maintain a competitive position, and manage information. The organisation will require proper management of information systems and the right capacity to service its customers. Contingency and business resumption planning is necessary for the organisation to ensure that they can deliver products and services in the event of adverse circumstances

Time constraint risk: Time has been a major wondering risk for the developer, since the developer has got very limited time against the demands of the system, thus might not deliver the system within the specified timeframe. The defined timeframe need to be flexed a bit so to accommodate the cumbersome demands and challenges of the system that may be faced so as to deliver a standard system.

Legal Risk: is the risk to earnings or capital arising from violations of, or non-conformance with, laws, rules, regulations, or ethic standards. The need to ensure consistency between paper and electronic advertisements, disclosures, and notices increases the potential for legal violations. Regular monitoring of the Old Mutual Insurance Administration System will help ensure compliance with applicable laws, rules, and regulations.

Having clearly reviewed the risks that are possibly going to be encountered by the project developing team, a work plan was developed showing all the activities involved in developing the web based insurance policy administration system. Before a schedule can be made, the developer needs to know the Software Development Life Cycle model to use. For the Insurance Administration System, the developer chose the Waterfall model.

Water fall model

A Waterfall SDLC model is a sequential software development process in which progress is regarded as flowing increasingly downwards (similar to a waterfall) through a list of phases that must be executed in order to successfully build a computer software (Lewis, 2000). The Waterfall model defines several consecutive phases that must be completed one after the other and moving to the next phase only when its preceding phase is completed.

Advantages

The most important advantage of the waterfall model lies in the fact, that there is minimum planning overhead for the steps that are to follow, since the activities in each of the phase is carried out upfront, it is feasible that one does not have to plan for the entire phase. There is certain amount of discipline that is enforced as one has to only look into one phase of the process at any given point of time. In other models it is often difficult to nail the start and end of the said phase, which is not the case with the waterfall model. The project does not slip on its schedule. The number of resources working on the project does not keep on increasing with each passing day, as the planning for the same is done at the start of the phase itself.

The waterfall development allows for departmentalisation and managerial control. A schedule can be set with deadlines for each stage of development.

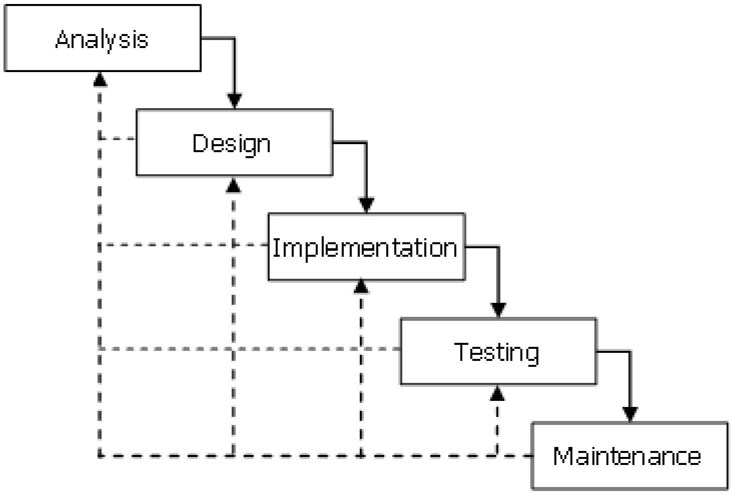


Figure 2.1: Waterfall Model

The developer identified stages of the waterfall model that will be undertaken for the completion of the project. The stages are explained on the next page.

Analysis Phase: Often known as Software Requirements Specification (SRS) is a complete and comprehensive description of the behavior of the software to be developed. It implicates system and business analysts to define both functional and non-functional requirements. Usually, functional requirements are defined by means of use cases which describe the users' interactions with the software. They include such requirements as purpose, scope, perspective, functions, software attributes, user characteristics, functionalities specifications, interface requirements, and database requirements. In contrast, the non-functional requirements refer to the various criteria, constraints, limitations, and requirements imposed on the design and operation of the software rather than on particular behaviors. It includes such properties as reliability, scalability, testability, availability, maintainability, performance, and quality standards.

Design Phase: It is the process of planning and problem solving for a software solution. It implicates software developers and designers to define the plan for a solution which includes algorithm design, software architecture design, database conceptual schema and logical diagram design, concept design, graphical user interface design, and data structure definition.

Implementation Phase: It refers to the realisation of business requirements and design specifications into a concrete executable program, database, website, or software component through programming and deployment. This phase is where the real code is written and compiled into an operational application, and where the database and text files are created. In other words, it is the process of converting the whole requirements and blueprints into a production environment.

Testing Phase: It is also known as verification and validation which is a process for checking that a software solution meets the original requirements and specifications and that it accomplishes its intended purpose. In fact, verification is the process of evaluating software to determine whether the products of a given development phase satisfy the conditions imposed at the start of that phase; while, validation is the process of evaluating software during or at the end of the development process to determine whether it satisfies specified requirements. Moreover, the testing phase is the outlet to perform debugging in which bugs and system glitches are found, corrected, and refined accordingly.

Maintenance Phase: It is the process of modifying a software solution after delivery and deployment to refine output, correct errors, and improve performance and quality. Additional maintenance activities can be performed in this phase including adapting software to its environment, accommodating new user requirements, and increasing software reliability.

Having identified all the activities to be carried out during the life cycle of the project, a schedule for the activities was constructed.

2.5 Work plan

A work plan was developed that shows all the activities and the time span for each activity that will be carried out during the Insurance Administration System development cycle. The work plan was represented using a Gantt chart. A Gantt chart is a detailed graphical representation of project tasks. It helps to divide project formation into separate parts and keeps track of daily project completion tasks done individually and by the team (Kerzner, 1995). It has horizontal bars plotted to represent a schedule. The horizontal bars show the start and end time of an activity, and the length of the bar shows the activity’s duration. Gantt Charts are a way to graphically show progress of a project. Management of a project is made easier if it is viewed as small manageable items where the dependencies are visually illustrated, parallel processes are discovered, the overall processing time determined and progress tracked.

Table 2.7 Project Time Allocation Schedule

Work	Starting date	Completion date	Duration (weeks)
Documentation	04/02/13	18/04/13	11 weeks
Project Proposal	04/02/13	11/02/13	1
Planning	11/02/13	25/02/13	2
System Analysis	25/02/13	11/03/13	2
Design Phase	11/04/13	01/04/13	3
Testing & Implementation Phase	01/04/13	15/04/13	2
Evaluation & maintenance	15/04/13		Ongoing

PHASE/WEEK	1	2	3	4	5	6	7	8	9	10	11
Project Proposal	■										
Planning		■	■								
System Analysis				■	■						
Design Phase						■	■	■			
Testing&Implementation Phase									■	■	
Evaluation& Maintenance											■
Documentation	■	■	■	■	■	■	■	■	■	■	■

Figure 2.2 Gantt Chart

2.6 Conclusion

Taking into consideration all the activities carried out in this phase the developer was now in a position to continue with the Insurance Administration System project development. This is because the project was approved by the management and the users, was considered feasible from all aspects. Tasks establishment and scheduling was successfully done. Risks counter measures have been put in place in case the risks surface. Therefore the developer safely carried out the analysis of the current system and all other activities.

CHAPTER 3: ANALYSIS PHASE

3.1 Introduction

Systems analysis is the dissection of a system into its component pieces to study how these component pieces interact and work (Kendall et al, 1988). System analysis is the process of gathering and interpreting facts, diagnosing problems, and using the information to recommend improvements to the system (Denis et al, 1999). The principal objective of the systems analysis phase is the specification of what the system needs to do to meet the requirements of end users. Systems analysis and design focused on what humans need to analyse data input or data flow systematically, process or transform data, store data, and output information in the context of a particular business. Systems analysis and design was used to analyse, design, and implement improvements in the support of users and the functioning of businesses that could be accomplished through the use of computerised information systems. Systems analysis identified the following:

- ❖ Outputs and processing needed.
- ❖ Data required providing this processing and output.
- ❖ Role of people in the process.
- ❖ Security aspects to ensure the efficient continuation of the business.
- ❖ Costs of providing the system.

During system analysis, the developer first gathered as much information as he could. The developer then used different information gathering techniques and presented the results in graphical forms. After gathering information about the system from the users and the involved parties, the developer then did an analysis on the current system. The analysis on the current system was done using the information gathered by the developer. After analysing the current system the developer then evaluated alternatives and lastly the developer did requirements analysis.

3.2 Information gathering methodologies

Information gathering is very essential at the stage of developing a new system; it gives the developer an idea of how the current system operates its advantages and also its short falls. This information came from the end user's point of view; this information determined the successful development of the proposed system since all the facts of how it should function were retrieved from the actual users of the system (Pomberger, 1986). The aim of this stage was to acquire detailed information from the current system users.

For each technique the developer used a sequential process to gather information. Figure 3.0 illustrates the process that the developer used to gather information.

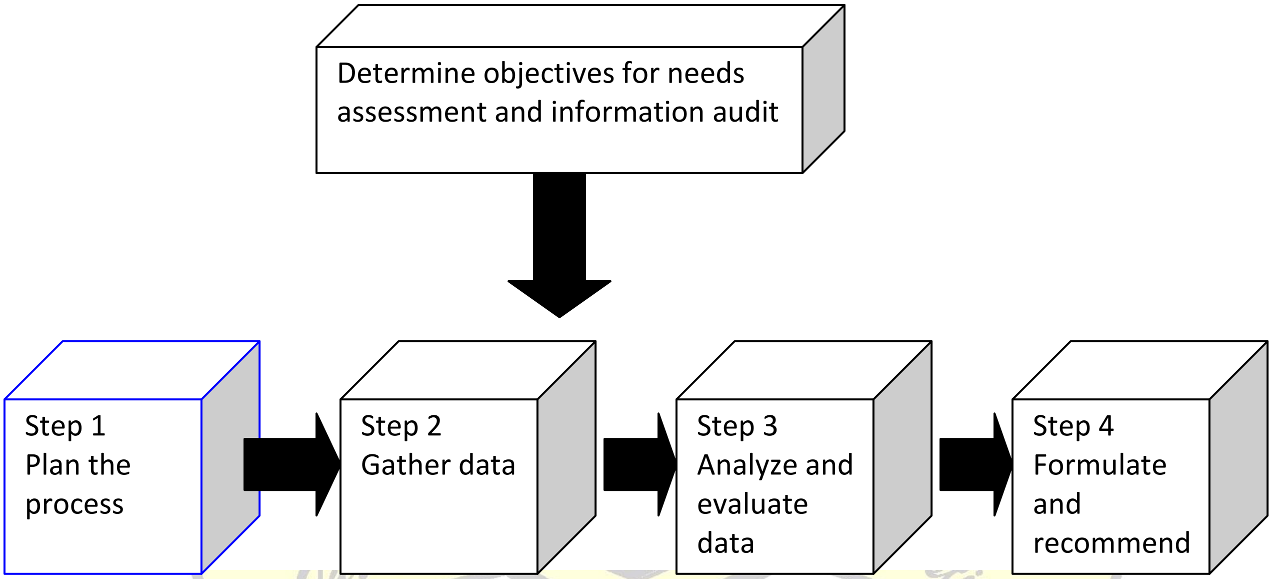

Determine objectives for needs assessment and information audit
Step 1
Plan the process
Step 2
Gather data
Step 3
Analyze and evaluate data
Step 4
Formulate and recommend

Figure 3.1 Information Gathering Process

The developer prepared a detailed document stating the objectives of the research. The developer at this stage answered the question “what is the objective of gathering information”.

The developer then moved to the first step of planning the process. The developer outlined activities that would be undertaken with each information gathering technique. The developer clearly stated how the process was to be carried out. After planning the process, the developer then went ahead with undertaking the information gathering technique. The developer here gathered as much information as needed from the participants involved. The stage involved conducting interviews, questionnaires, document analysis and observations.

After data had been gathered, the developer analysed and evaluated the data. The developer used graphs and pie chart to present the information gathered from the information gathering techniques. After the analysis of the data the developer proceeded to give his / her recommendations based on the information that was gathered. The techniques that the developer used to gather information include:

- Joint application development
- Observation
- Interviews
- Questionnaires

Participants in gathering information included the following people:

1) Insurers - these are the clients to Old Mutual. They gave out information as to whether they were happy with the operations of the current system. Insurers also gave information as to the improvements that they wish to see.

2) Insurance clerks - Assisted the developer in providing information about the current systems including the documentation used and the processes undertaken. They identified particular individual needs of a new system.

3) Information Technology team – the technical team provided information on the operations of the current system. The technical team involved programmers, system analysts, database administrators and the hardware team.

3.2.1 Questionnaires

Questionnaires are a set of questions for obtaining information from individuals. This methodology involves drafting a document containing a number of standard questions and distributing them to existing system users (Jeffrey et al, 2007). Questionnaires were used to collect both qualitative and quantitative data. They were distributed manually and electronically and this made it possible to reach a vast number of people regardless of physical location or geographical dispersion. The aim was to find out general processes, documents used, document movements and volumes of data.

Strategies that were used for the questionnaires are:

Workers questionnaire: This questionnaire was designed for the insurance clerks who are directly involved in the insurance processing so as to understand what changes they would like to see happening in their work routines.

Managerial questionnaire: This was designed for the Fund Managers and the management as a whole in a bid to understand what they expected from their workers to achieve within their work rates and deadlines.

Customer questionnaire: This was given to customers to understand what they expected from the organisation and how they felt needed to be done for the fast approval of their claims.

During the design of the questionnaires, the developer made sure that the questionnaires were brief and user-friendly. There were clear instructions, the questions were related and were in logical order. The wording of the questionnaires was simple to avoid misunderstanding. The developer sent the questionnaires to the clients and the workforce through email, hand delivery, post and a web survey was also used. Email survey involved sending an email in the text of an email or as an attachment, which respondents filled in and sent the answered questionnaire back to the developer. A web survey involved sending people a link to a web page containing a questionnaire that was filled in online. The questionnaire was designed with software that will ensure that skip instructions were accurately followed (i.e. an instruction to 'skip to Question 20 if you answer "yes" to Question 10'). A sample of the questionnaire is shown in **Appendix C.** The use of questionnaires had some loopholes and merits of using them.

Advantages of questionnaires

I. Questionnaires were relatively cheap since stationery and printing costs only were the major costs involved.

II. Email surveying allowed large numbers of respondents to be questioned. Geographical location was not a barrier since all people got the questionnaires through email and the web link

III. They were time saving, since the developer also used emails to distribute some questionnaires

IV. It promoted honest, precise and free participation because of anonymous input hence valid information was obtained.

V. The use of questionnaires allowed the respondent to give answers at their own pace without being rushed hence making sure correct answers were provided.

Disadvantages of questionnaires

I. Some questions were answered wrongly because they would have been misinterpreted, and some were left without answers because they seem ambiguous to the respondent.

II. Questionnaires lacked the integrity of further clarification on seemingly ambiguous questions and responses.

III. There was no control over who fills out the questionnaire, and the developer never knew if the right person had completed the questionnaire because some people are reluctant to answer questions and pass the questionnaires to their friends to answer for them.

IV. Response rates were low and it was difficult to know the characteristics of those who have not filled in the survey and how their non-response will affect the findings.

3.2.2 Interviews

Interviews are a way to get in-depth and comprehensive information by asking face to face questions, which may otherwise not be understood from the onset of questionnaires (Gary et al, 2009). The insurance clerks and the funds manager of the insurance management system were orally (face-to-face) interviewed using an interview guide that will be shown in **Appendix B.**

The developer used the following interview strategies:

Telephone interviews - the interview involved interviewing each insurance clerk, client and the fund manager through a phone call.

Face to face interviews - these were one to one basis with the management, workers, as well as a sample of clients.

Group interviews - these were carried out both intra and inter-department, with the in-cooperation of the management from the supervisors' level to the top management.

Before the interview, the analyst made a list of people to be interviewed and in what order, planned and noted down a list of questions to be asked. A prior appointment with the person to be interviewed was made. Three types of interview questions were used by the analyst to gain a rich understanding of the requirements.

Closed-ended questions: Closed-ended questions enabled the developer to control the interview. Answers to the questions required only a short answer of one or two words (true/false, multiple choice, rating on a scale, or ranking).

Open-ended questions*:* Open-ended questions are those that leave rooms for further elaboration by the interviewee. These questions provided additional information or problems that a user did not like to talk about.

Probes: These were follow-up questions in response to one of the above questions, when the developer was unclear about the answer.

In this research, underwriters, funds manager, system administrator, senior management and users of the current system were interviewed independently. They were selected based on their working experiences and the level of involvement in the various processes and activities done in traditional insurance administration which will be improved through the use of web based insurance administration. The interviews were conducted in a semi-structured format that allowed respondents to express their own viewpoints. Respondents were guided by a list of interview topics about key factors determining the acceptance of the proposed system. Occasionally the respondents raised new issues such as a regulation control and security features of the Insurance Administration System. Respondents were free to discuss at all time. The interview was conducted in English language and interview time varied in length from 30 minutes to one hour. The interview was recorded by both notes taking and taping which facilitated less interruption in the interview when taking notes. The developer listened very carefully to the interviewees and gave the interviewees opportunity to answer and to ask questions.

Advantages of interviews

I. The presence of an interviewer allowed complex questions to be explained, if necessary, to the interviewee.

II. Telephone interviews provided greater levels of monitoring because supervisors could unobtrusively listen in to interviews to ensure that they are carried out correctly.

III. Face to face interviews allowed for visual aids to be used

IV. A friendly environment was created before the interviews by entertaining the interviewees to boost confidence in their expressions and honest answers were obtained.

V. Immediate and direct firsthand information was obtained.

Disadvantages of interviews

I. The cost associated with face-to-face interviews limited the size and geographical coverage of the survey.

II. Interviews were time consuming especially in cases where there were misunderstandings arising and clarifications needed to be made.

III. With telephone interview, it was difficult to ask sensitive questions on the telephone

IV. Interviewers sometimes were biased, which affected the reliability of responses. Such bias emerged from the way in which questions were asked.

3.2.3 Observations

An observation is an information gathering method enabling researchers to learn about the activities of the people under study in the natural setting through observing and participating in those activities. Before the observation the developer obtained a good understanding of the operation carried out in traditional insurance management. All the steps involved in insurance policy application, claims process, the premium process and many other processes were observed getting an understanding of all the steps in the processing cycle noting each output from each procedural step. By directly observing operations and activities, the observer developed a holistic perspective, that is, an understanding of the context within which the current system is operating and how the web based insurance administration should operate. The developer observed the insurance clerks and the funds manager as they carried out their daily duties and as they moved within and around the other financial institutions. The observation card sheet will be shown in **Appendix D**. More so, the developer observed all the steps in the processing cycle, examining each form, record, and report. Consideration was also given to each person working with the system. This fact- finding technique was used to test for the validity of the information collected from the interviews and questionnaires.

Advantages of observations

I. It allowed the developer to discover the relevant information on his/her own without probing it from someone else.
II. It was very cheap as there were no special arrangements carried out.
III. It did not interrupt the work of those being observed.

Disadvantages of observations

I. Not all the activities were observable.
II. Hawthorne Effect: During the observation day, people worked more efficiently than the normal day. Some operations ran less smoothly because people were nervous during observation

3.2.4 Joint Application Development (JAD)

JAD is a structured process in which ten to 20 users meet together under the direction of a facilitator skilled in JAD techniques (Kendall et al, 1988). The facilitator is a person who sets the meeting agenda and guides the discussion but does not join in the discussion. The JAD group met until all of the issues had been discussed and the needed information collected. The developer used electronic JAD which used groupware. Each participant used special software on a networked computer to send anonymous ideas and opinions to everyone else. In this way all participants could contribute at the same time, without fear of reprisal from people with differing opinions. After the completion of the JAD session, the responsible systems developers and information consultants updated the necessary documents and prepared a final report that summarised all discussions, facts, findings, and conclusions

Advantages of JAD

I. It allowed key users to participate in the systems development process, and they felt the sense of ownership in the result and support of the new system.
II. Allowed for the simultaneous gathering and consolidating of information

III. The input from numerous people provided different perspectives on the desired system and generated creative ideas

Disadvantages of using JAD

I. Sometimes people were reluctant to challenge the opinions of others, a few people dominated the discussion and not everyone participated

II. It was expensive to set up the JAD technique

III. Many ideas were generated that additional sessions and meetings are needed to resolve the conflicts

Overview of information gathering

The developer collected, investigated and evaluated the information gathered. From the various information gathering techniques, the developer got a clear understanding of the processes and activities involved in traditional insurance policy management system with the objective of using that information to develop the proposed Insurance Administration System. The questionnaires were used to explore customers' perceptions in adopting web based insurance system in general. This enabled exploration of a large number of people's views, however, the depth of research was limited because the answers were predetermined and standardised. The interview helped to add richness to the findings as it provided a wealth of rich data and reflected people's opinions and attitudes. JAD helped the developer not only to identify the project stakeholders but also helped in knowing the stakeholder expectations.

3.3 Analysis of existing system

The current system at the organisation is not fully computerised. It makes use of the Microsoft Outlook, Word and Excel as software packages and printers as hardware packages. Access Database is used for report generation and a system generated in-house for the record keeping and capturing of the client details. At the present moment, most corporate and individuals are looking for companies to buy insurance policies from. Insurance policies include legal cover, hospital cover and life insurance. When a client wants to apply for an insurance policy or to make a claim or check on their account status they have to visit Old Mutual in person, make a phone call or send an email. Queries pertaining to the clients' accounts are done by sending an email to the insurance policy clerks or the administrator or by making a phone call to the insurance clerks.

Old Mutual has different insurance facets that maintain their own database of clients. Life cover insurance is handled by the Old Mutual Life Assurance Company, short term insurance e.g. car insurance, property insurance is handled by the RM Insurance. Retirement benefits are handled by the Employee Benefits department. All these organisations store details of their clients. One client may have all the insurance policies and because the different policies are being handled by different administrators, the client will end up having many accounts.

Below is a brief description process of the current system:

New clients - The client visits the nearest Old Mutual branch and then fills an application form stating the type of insurance policy that he/she wants. If the client wants to insure for a car or any other property, he /she has to visit the nearest RM branch. If the client wants to make a retirement benefit he/she visits the Life Assurance Company. New clients can download application forms on the website www.oldmutual.co.zw and print the application form. Fill in the form and send the form back using the address on the form. For the client to make monthly payments of the insurance policy he/ she has to visit the nearest Old Mutual branch and make the payment.

Insurance claim process- for a client to make a claim he/ she has to be a member of that insurance for some reasonable time. Claims that clients make include hospital cover claim, legal cover claim, short term insurance claim and life cover claim. For a client to make a claim he visits OM with the relevant documents as proof to the claim. Claims are analysed by the funds manager. The funds manager takes in the claim application form and the feedback may come after days or weeks while in some cases the client will really need the cash urgently. For feedback a call is made to the client or an email is sent to the client to come and collect the confirmation receipt.

Inputs: The inputs to the existing system are:

I. Client details (both Personal and Contact).

II. Insurance applicant's received through mail or post.

III. Insurance claims applications received through an email or hand delivered

IV. Transaction details – the disbursed amounts and the account status

Processes

I. Application process for insurance policies

II. Claims application process

III. Insurance claims approval process

IV. Account status enquiry- when the client enquires about their account.

V. Registration of new clients

VI. Report preparation

Outputs

I. Reports on the account status of the clients

II. Client records

III. Receipts approved claims

IV. Reports on the performance of funds.

V. Applicants file storing insurance policies

VI. File storing approved and disapproved insurance policies

VII. Insurance claims approved and disapproved files

3.4 Process analysis

The process model is a formal way of representing how a business system operates (Joseph, 2003). It illustrates the processes and activities that are performed and how data moves among them. An activity diagram is drawn to clearly illustrate the process analysis of the current system. This shows the activities that take place in the system.

3.4.1 Activity diagram

The activity can be described as an operation of the system (Lucey, 1997). The control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Activity diagram is used to show message flow from one activity to another. An activity diagram is used to represent the inputs, processes and outputs of the current system. It is used to model processes, data, inputs and outputs of a system and the sequential work flow by focusing on action sequences and respective action initiating conditions. Figure 3.2 overleaf is an activity diagram of the current system.

The activity starts when a client visits Old Mutual offices to create an account. When an account has been created, the client then applies for an insurance policy that he or she desires. The client submits the application forms to the insurance clerks and waits for a response. When clients' accounts have been successfully created, they can start transacting. The client can now pay for the monthly premiums and make insurance claims. Application for the insurance claims can only be done by visiting the Old Mutual offices and submit the claim application form and the required documents.

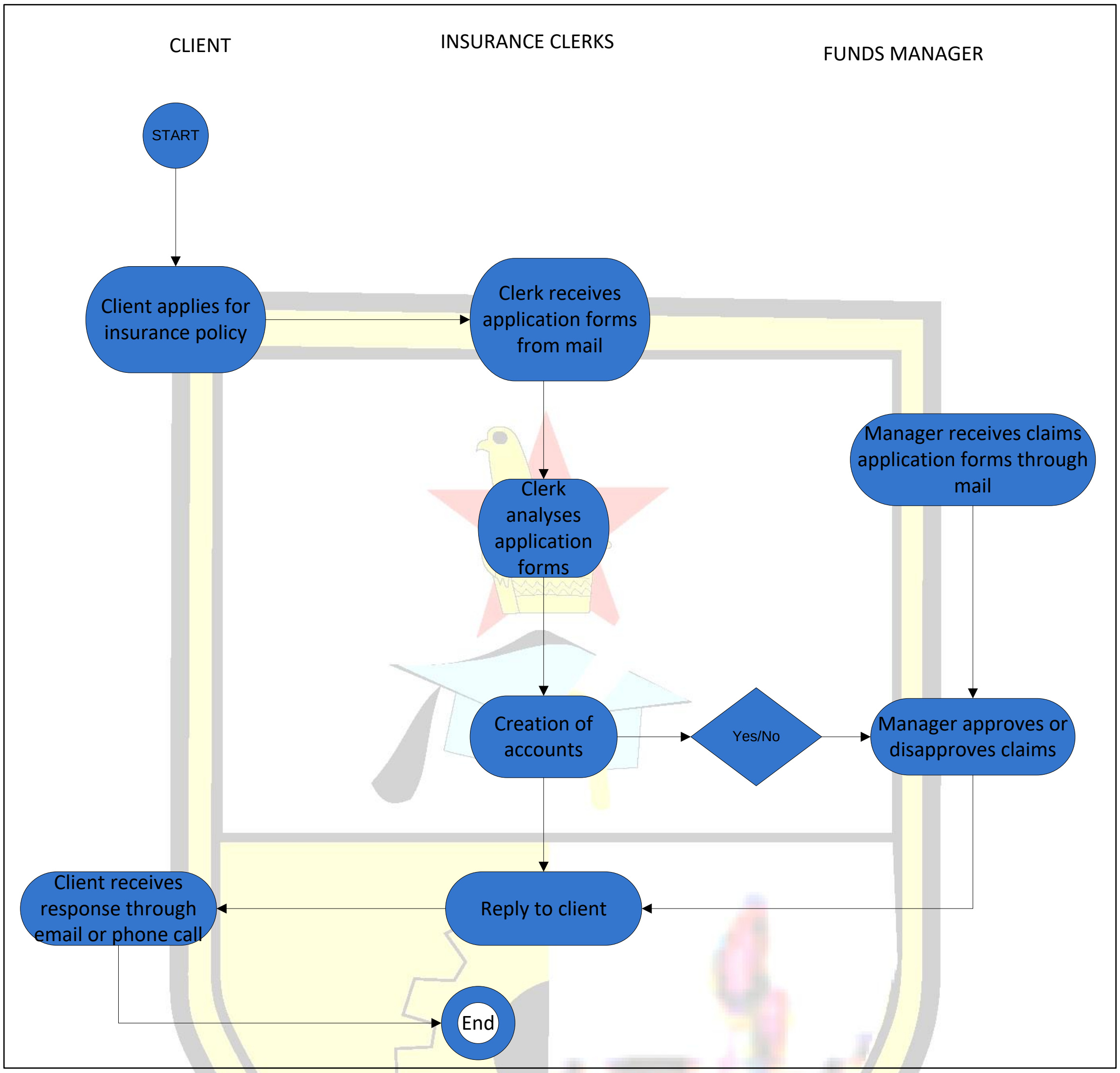
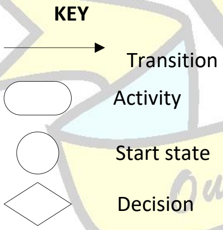

CLIENT
INSURANCE CLERKS
FUNDS MANAGER
START
Client applies for insurance policy
Clerk receives application forms from mail
Manager receives claims application forms through mail
Clerk analyses application forms
Creation of accounts
Yes/No
Manager approves or disapproves claims
Client receives response through email or phone call
Reply to client
End

Figure 3.2 Activity Diagram for the Current System



3.5 Data analysis

The aim of this section is to illustrate the processes and activities that are performed in the current system and also the data flows of the system. It also provides a better visual picture of what is really happening in the organisation. Two diagrams are involved in the illustration process and these are:

- Context diagram
- Data flow diagram

3.5.1 Context diagram

The context diagram for the current system is an outline of the system boundaries (Denis et al, 1999). The context diagram reflects the degree to which the system relates to the external environment and the fact that it is an open system. The context diagram focuses on the relationships with external entities and identifies the information that is exchanged between these external entities and the business system under review.

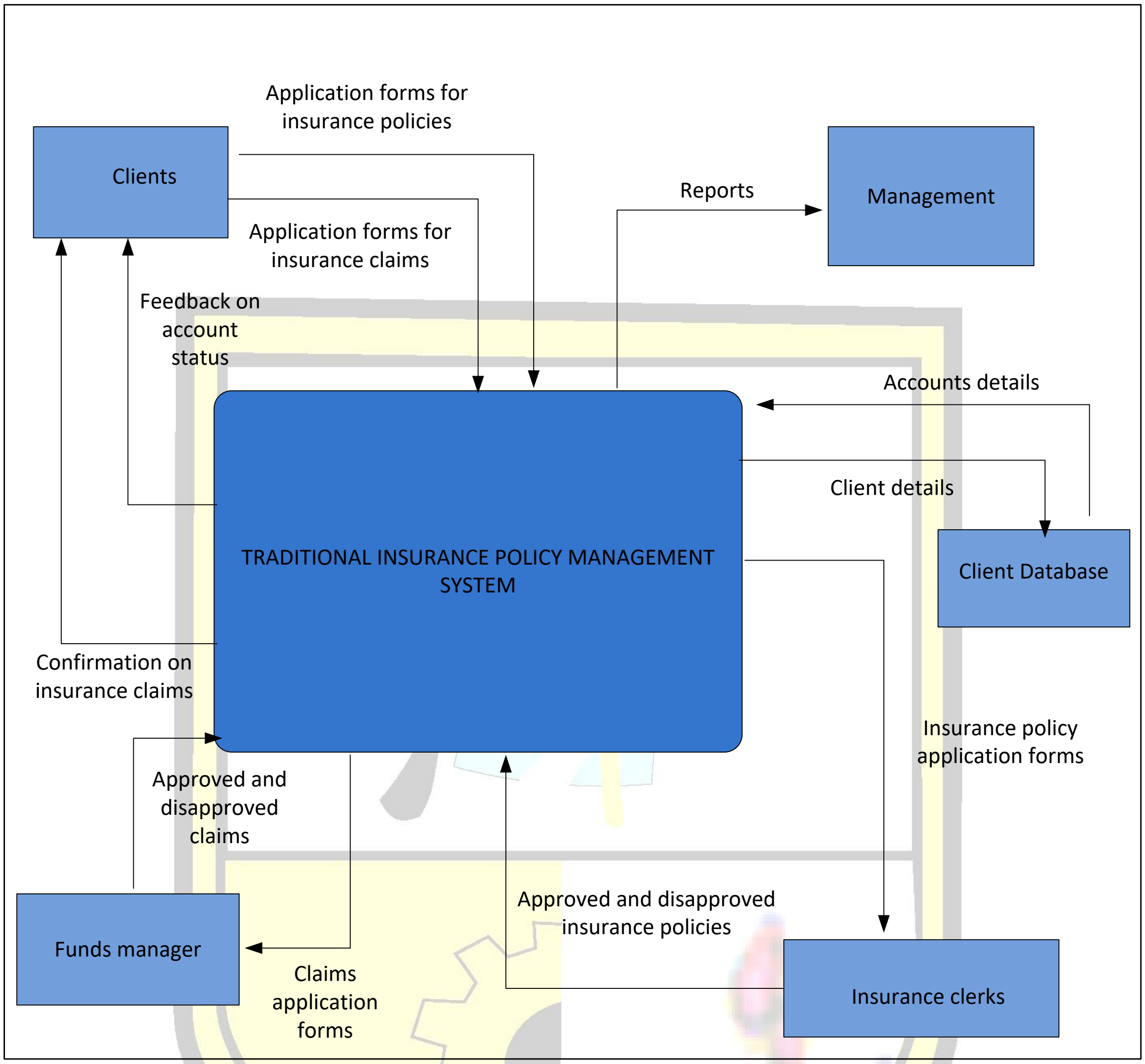

Application forms for insurance policies
Clients
Management
Reports
Application forms for insurance claims
Feedback on account status
Accounts details
Client details
TRADITIONAL INSURANCE POLICY MANAGEMENT SYSTEM
Client Database
Confirmation on insurance claims
Insurance policy application forms
Approved and disapproved claims
Approved and disapproved insurance policies
Funds manager
Claims application forms
Insurance clerks

Figure 3.3: Context Diagram for Current System

Key

▬ Process

■ Entity

→ Data Flow

3.5.2 Data flow diagram

Data flow diagram is a graphical representation of the flow of data through an information system (Kendall et al, 1988). Rather than showing the strict order of execution of steps, it shows how processes depend on one another for information. It shows data flow from external into the system and shows how the data moved from one process to another process. Data Flow Diagrams were used to compare the new system and the old system. With this comparison, the developer found the gap between two systems and the effectiveness of the improved system.

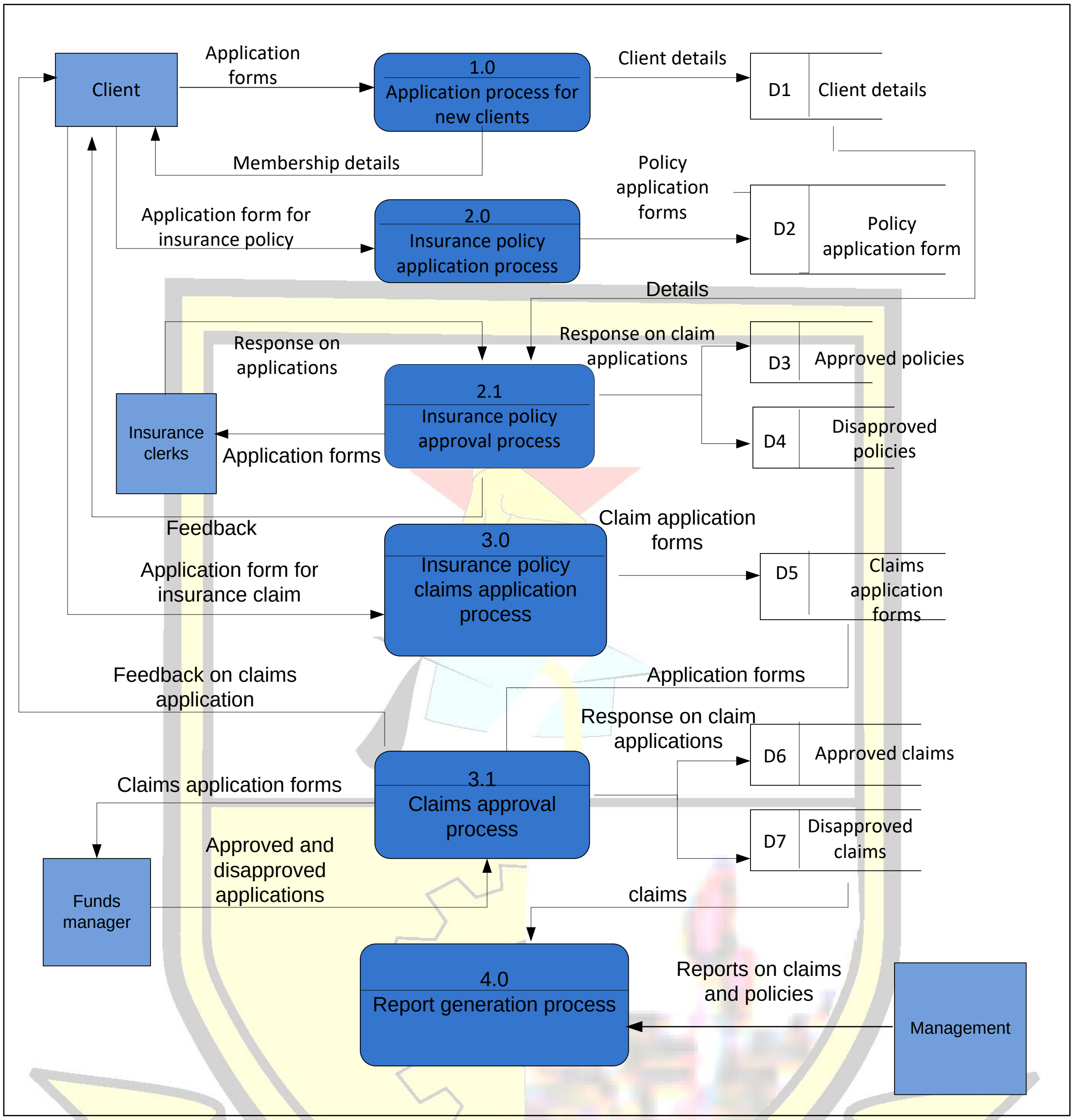
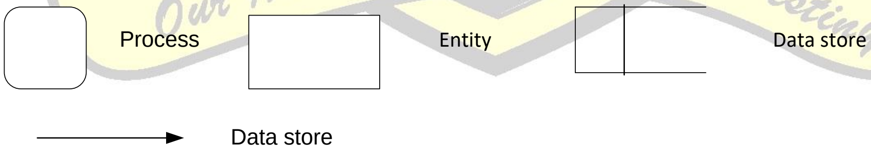


Figure 3.4: Data Flow Diagram for the Current System

Key



3.6 Weaknesses of the current system

The current system at Old Mutual has weaknesses that are costing the organisation a lot. Clients currently cannot view their monthly premiums and this has led to a decline in customer loyalty and client retention. The insurance market is growing rapidly and ICT is ever changing and hence there is need for Old Mutual to keep up with the growing market and the ever changing ICT facilities.

Weaknesses of the current system include the following:

I. System inflexibility as clients can only access their balances and statements on a working day, during working hours. Claims can only be done during the week and in some cases if it's a hospital insurance claim and the client needs money urgently she/he has to wait till Monday. This results in work overload during Mondays and overloading the system may slow the performance of the system.

II. Time Consuming as clients wait for a long time before they can be given their account balances and statements and also the client has to travel to the nearest Old Mutual branch just to make a claim or to increase the scope of the insurance.

III. If a client makes a claim she or he has to go through a long process for the claim to be approved and signed. The client has to move with his or her documents from one office to the other having the documents signed by different personnel.

IV. Standardized reports cannot be produced because each insurance facet has its own data mart. The life assurance has its own data mart and the funeral policy has its data mart and this makes it hard for comprehensive reports to be made.

V. Document management with the current system is not secure. All the clients' documents for example birth certificates are stored in cabinets which can be prone to theft and natural disasters without backups. So there is need to automate the document management systems as this increases security and it has the option of backing up the documents.

3.7 Evaluation of alternatives

With reference to the Feasibility Study that was undertaken in the Planning Phase, it was concluded that the project was viable. Evaluation of the alternatives assisted the developer to choose the best alternative that yielded optimal results. Three alternatives were evaluated and these included:

I. Outsourcing
II. Improvement of the current system
III. In-House Development

3.7.1 Outsourcing

Outsourcing occurs when information systems activities are carried out by a provider outside the organisation (Joseph, 2003). This is when a company buys software package off the shelf from an outside supplier, rather than performing the same work within its own facilities.

Advantages

I. Less expensive - Since costs are spread out through all customers to the vendor hence no need to charge inflated prices.

II. Less Time to implement – Outsourcing a software package requires less time to implement as the package is already designed and available.

III. Bench mark standards – This is because the package is also being implemented by other companies (bench-mark) thus believed to be standard.

Disadvantages

I. It may involve the purchasing of necessary hardware for compatibility with the software being purchased.

II. Vendor relocation would make system support difficult.

III. Outsourcing may not appropriately differentiate service delivery from those of competitors i.e. it does not lead to the gaining of competitive advantage.

IV. A lot of technical expertise may be needed for the maintenance and installation of the software.

Reasons why Old Mutual did not outsource include:

Lack of innovation and know how: From the feasibility study it was discovered that Old Mutual ICT team was very large. The company has a lot of technical expertise and human expertise and innovation is the greatest asset that any organization can develop and retain. Thus, outsourcing would lead to a loss of innovation and know-how in applying technology and also offshore outsourcing of information systems services could pose the greatest risk to the organization.

Loss of Technical Knowledge: When a service is outsourced, clients gradually lose their understanding of the service over time. Even if the provider delivers innovative services to the client, a large proportion of the new knowledge required remains in the hands of the provider and cannot be transferred to the client. There will be a decline in employee morale since they will no longer feel their importance in the organisation. Outsourcing may eventually lead to employee turnover.

Wait Time: With an in-house computer team, the response time for a server crash or virus attack would be almost immediate. Whereas outsourcing may lead to a delay in operations since the company will have to wait until the company can prioritize Old Mutual. Even regular staff seminars or tutorials may have to wait until the company performing the outsourced work can schedule one of its staff members to address Old Mutuals' issues.

Evaluation

From the mentioned benefits and drawbacks of outsourcing the system i.e. buying an Off-The-Shelf software package, it was concluded that it's not worth to consider this option because the identified demerits far outweigh the identified benefits. This is to say this alternative was safely rejected as it might cause even more loss to the organisation hence violating the purpose of Old Mutual to be a profit-making organisation.

3.7.2 Improving the current system

This involves identifying processes with weaknesses and enhancing them. By improving the current system it implies no change to the original operations but rather making some changes or improvements in areas of need. For an insurance claim to be approved, the client has to move from one office to another for signatures before the client can get the claim disbursement. The movement of forms from one office to another for approval is time consuming. If the organisation decides to improve the current system, it might target the insurance claims part and try to solve that area. Improvement might include reducing the insurance claim activities or having the claims process be done electronically. Another weakness of the current system is that clients can only make insurance claims by visiting Old Mutual offices, and the developer might improve that particular part by having a system that will allow for the client to have any time and anywhere access to their accounts.

Improvement to the current system entails improving certain processes and leaving other processes as they are. The developer analysed the option of improving the current system and identified some strengths and drawbacks of improving the current system at Old Mutual.

Advantages

I. **Less costly-** No additional hardware, software and extensive training for the system users as they are already familiar with the existing system

II. **Time saving-** The system will not be developed from scratch so it will not take long to implement

Disadvantages

I. **Problem Inheritance –** existing problems might creep into the system unnoticed and might cause undesired results later on.

II. **Complex Problems -** Some of the problems encountered might be very difficult to rectify for example the performance of the system and the amount of data it can hold.

Evaluation:

Improvement of the current system was not considered viable as it might cause the emerging of the current problems in future due to problem inheritance. In such scenarios it results in this option ending up even more costly than it is currently.

3.7.3 In-house development

This involves the use of the organisation's financial resources, use of the company's technical experts, data capture clerks and the management in order to design a system that best suits user requirements (Jeffrey et al, 2007). The organisation's IT expertise will be responsible for the project's design and user training. The administrator will be responsible for maintenance, system support as well as managing underwriters. At the end of the project system documentation will be produced and can be used for future reference for system support. The firm can create a system that is specifically for Old Mutual. The system to be developed should be able to curb all the problems currently being faced by the current system.

Advantages

I. **User satisfaction**- satisfies unique user requirements since the system will be tailor made as per the identified requirements

II. **Cheap in the long run**- the cost of maintenance is within reach for the organization since the system will be developed by an internal IT staff.

III. **Competitive advantage –** this will be gained since there will be unique differentiating features from the system operations of the competitors

Disadvantages

I. **Additional costs** - These are costs that arise from the acquisition of additional hardware and software at the initial stage of the project

II. **Time consuming -** There are so many phases incorporated in system development so it takes time to develop.

III. **User training –** Since the users will not be familiar with new system there is need for intensive user training which might be costly to the organisation.

The following outlines the reasons as to why the system was developed in-house:

I. The proposed system will satisfy unique user requirements

II. It meets the organisation’s long term goals.

III. Minimize changes in business procedures and policies

IV. Users of the system will assume ownership of the system since their requirements are incorporated during system development and this increases employee morale

V. The system has room for expansion in respect with the changing technology

VI. Maximises utilisation of resources.

VII. Is less costly to implement.

VIII. Small tasks can be integrated in the system thereby reducing manpower.

Evaluation

Given the problems that are being faced by the company at present, it was safely concluded that its worth to develop a new system because of the associated benefits or advantages which far outweigh the demerits associated. Considering the general advantages attached to the other alternatives it leaves those of developing a new in-house system far better a profitable option. One of the main benefits that are going to be enjoyed from developing the new system will be the satisfaction of unique user requirements as well as reduction in vendor dependency syndrome since the developers will be internal.

3.8 Requirements analysis

Requirements analysis is a very important stage in the system analysis phase and development of the software solution because this is where the requirements of the clients and the client domain are noted and taken care of (Gary et al, 2009). If this part is not done well, then there will be need to correct the software solution in the future. Requirement Analysis was divided into 2 categories which are functional requirements and non- functional requirements. The developer did the requirements analysis of the proposed Old Mutual Insurance administration system. To help clearly give a detailed explanation of the actors in the proposed system, the developer used a use case diagram.

3.8.1 Functional requirements

Functional requirements may be defined as the product capabilities or things that a product must do for its users (Lucey, 1997). Functional requirements clearly define how software behaves to meet user needs. Functional requirements define of features i.e. inputs, processes, outputs and stored data needed to satisfy the system objectives. The proposed system's functional requirements are outlined below:

Management of clients' online accounts with security: The proposed system will be secure to reduce the risk of fraud. Each insurer must have their own username and password and insurance administrators must be able to change their passwords on a monthly basis. The proposed system must have different access controls for different levels.

Online Trading: The proposed system should have an interface for old and new clients to apply and access their accounts through the comfort of their homes without visiting the nearest Old Mutual branch. Online trading includes applying for different insurance policies, view their statements online and make insurance claims with a click of a button.

Central Database: Use of a central database to enable real time and online data capturing and manipulation to facilitate access to client details from any computer connected to the internet

Integration of all insurance facets: The administration system should allow for the integration of all insurance facets. This means that if a client applies for the different insurances, all the information pertaining to all the insurances will be available in that single account. Integration of all insurance facets into one account of a client means that there will be a reduction in data redundancy.

Validation and Verification: The insurance administration system will have facilities for integrity checks and to allow clients to verify data before it is captured into the system. Validation and verification will reduce fraud and also increase the correctness of information entered into the system.

Report Generation: The proposed system will have efficient reporting capabilities to facilitate the generation of timely reports.

Backup: The system shall provide periodic backups of the transactions so as to enable the recovery of the information in case of hardware or software failure

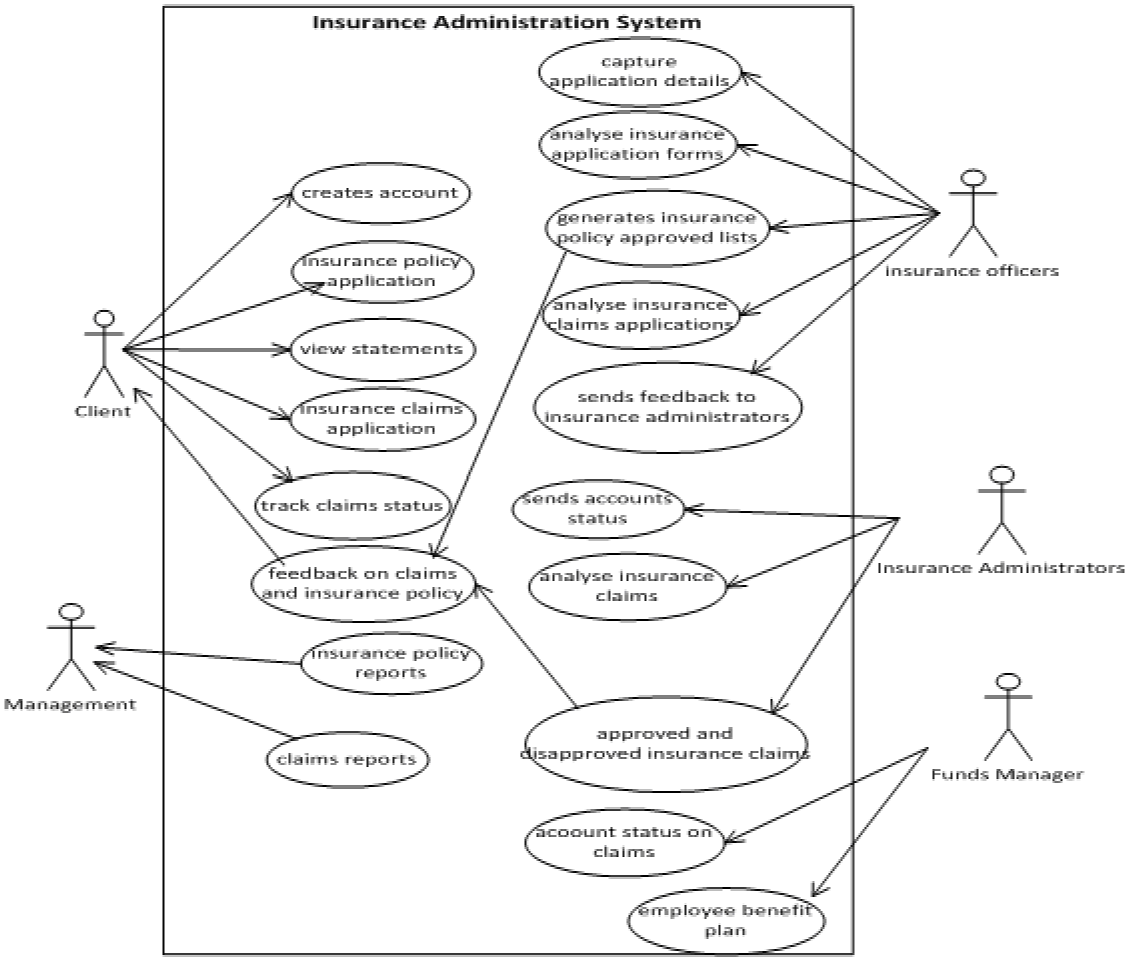
The developer used a Use Case diagram to illustrate the functional requirements of the Insurance Administration system.

Use Case Diagram

A Use Case diagram is a sub class of behavioral diagrams (Pomberger, 1986). It shows how a system interacts with the external entities. Use Case Diagram shows what the developer wants the system to do. One of the major benefits of this diagram is communication. The two main components of a use case diagram are use cases and actors. A Use Case Diagram helped the system developer to discover the requirements of the target system from the user's perspective. The usage of use case diagram includes:

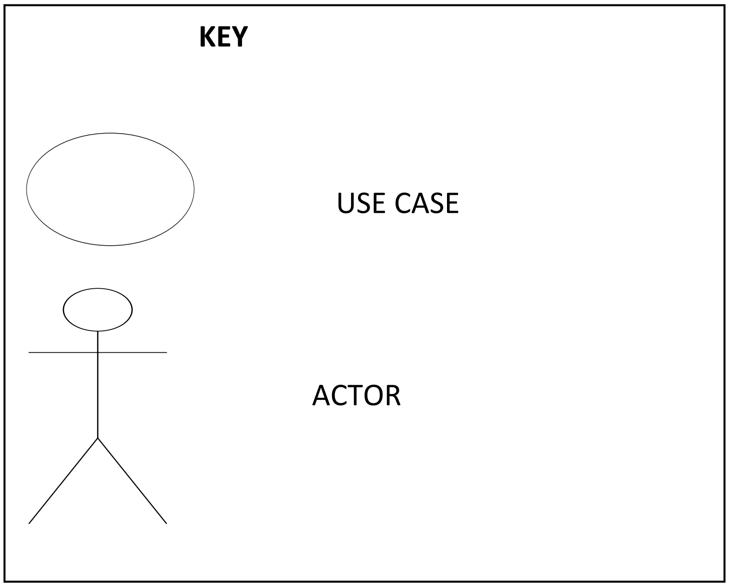
a) the behavior of a system from a user's standpoint

b) Functional description of a system and its major processes.



Insurance Administration System
capture application details
analyse insurance application forms
creates account
generates insurance policy approved lists
insurance officers
insurance policy application
analyse insurance claims applications
view statements
Client
sends feedback to insurance administrators
insurance claims application
track claims status
sends accounts status
Insurance Administrators
feedback on claims and insurance policy
analyse insurance claims
Management
insurance policy reports
approved and disapproved insurance claims
Funds Manager
claims reports
acoount status on claims
employee benefit plan

Fig 3.5: Use case diagram



3.8.2 Non-functional requirements

Nonfunctional requirements are the quality attributes, design and implementation constraints, and external interfaces which a product must have. Examples of quality attributes include availability, maintainability, performance, portability, reliability, robustness, security, scalability, testability, usability, and others.

Hardware Consideration: The hardware of the proposed system must be capable of retaining all relevant information and providing speedy service.

Increased insurer confidence: Investor confidence is normally boosted by the ability to initiate fair, transparent and secure transactions. The system is expected to boost investor confidence which in turn will result in more insurance policies being bought.

Usability Requirements: The proposed system should have a user interface with self explanatory menus that make it easy for the user to understand the functionality depicted by menu items.

Timeliness: The proposed system is expected to significantly improve on the processing speed of carrying out the claims approval process.

Back up: The database system should be backed-up continuously twice a day in order to minimize data loss in the case of a catastrophic failure. The system administrator should maintain the system and its databases to ensure integrity and availability of information.

User access levels: The system must have a log on feature limiting individual users to those functions they have a need to access according to their responsibilities.

3.8.3 Constraints

In the development of the system the following maybe encountered during the different stages of development.

Technical constraints

I. Not enough manpower to concentrate on different modules of the system.

II. The company will have to hire IT personnel to maintain the system or they can consult from the system developer for maintenance

Time constraint

More time is needed to complete the project as it encompasses the coordination of all modules that will be used within the system.

3.9 Conclusion

In this chapter the developer managed to understand in depth of how the current system works, how positively and negatively it affects the operations of the organisation. The developer outlined why building an in house system is an advantage to the organisation after evaluating the many alternatives that are at hand. With the aid of a context diagram, activity diagram and the data flow diagram the developer managed to identify the weaknesses of the current system and realised how the proposed system should operate in order to rectify and eliminate the weaknesses available. After all has been done the developer undertook the design phase which was mainly concerned with the designing of the proposed system.

CHAPTER 4: DESIGN PHASE

4.1 Introduction

This phase is mainly concerned with designing of the proposed system. That is how the system will work, how software and hardware interact and those tasks that focus on the specification of a detailed computer solution were looked at. The major development effort for this project was to design, build and test a dynamic and interactive insurance policy administration system that will aid the recording, storing and tracking of insurance policies. Activities that were undertaken at this stage include:

I. Selecting any software packages which will be part of the system

II. Designing the components of the system database structures, inputs, outputs, internal processing, system interfaces, technical environment and the overall system architecture

III. Developing procedures for conversion from existing system to new system

Processes that the developer undertook include:

Prepare for System Design- the existing project repositories were expanded to accommodate the design work products, the technical environment and tools needed to support system design were established, and training needs of the team members involved in system design were addressed.

Define Technical Architecture- the foundation and structure of the system was identified in terms of system hardware, system software, and supporting tools and the strategy were developed for distribution of the various system components across the architecture.

Define System Standards- common processes, techniques, tools, and conventions that were used throughout the project were identified in an attempt to maximise efficiencies and introduce uniformity throughout the system.

Create Physical Database- the actual database to be used by the system was defined, validated, and optimized to ensure the completeness, accuracy, and reliability of the data.

Prototype System Components- various components of the solution was developed and demonstrated in an attempt to validate preliminary functionality, to better illustrate and confirm the proposed solution, or to demonstrate proof-of-concept.

Produce Technical Specifications- the operational requirements of the system were translated into a series of technical design specifications for all components of the system, setting the stage for system construction.

4.2 System design

System design transforms the detailed requirements of the definition stage into a complete, detailed specification of the system (Ryan, 2001). This section answers how the new system will work, the inputs, processes and outputs from the system and also the interactions and relationships among entities. It describes the desired features and operations in detail, including screen layouts, business rules, process diagrams, pseudo code and other documentation of the proposed system.

4.2.1 Overview of how the proposed system will work

Clients

The system should allow the client to browse the internet and register with Old Mutual to create an online account. During the registration process, the client has to supply all the personal details about him or her.

After registering online, on the internet, the client can then apply for an insurance policy that he or she wants. To apply for the insurance policy the client has to upload certificates that include car certificates, employer details, bank details, a copy of the salary and household property valuation forms depending on the type of insurance policy that he or she is applying for. The client submits the application form online and waits for feedback from Old Mutual.

If the application is approved, then the client is deemed to be an insured client. She or he pays monthly premiums to Old Mutual as cash or through the bank. He or she then logs in into her account and checks if the premium has been updated. The insured client can also increase the number of insurance policies that he or she has. The insured client does this by logging in their accounts and go through the same application process as the one mentioned earlier on except that he or she now uploads different documents depending on the type of insurance policy that he or she is applying for.

The insured client may log in and apply for a claim. During claim application the client will upload the required documents and give details pertaining to the claim. He or she then submits the claim online and waits for the response from the underwriters

Underwriters

The underwriters view the application forms submitted according to their area of expertise. Legal underwriters can only view legal insurance application forms and life cover underwriters can only view life cover application forms and so on. The underwriter analyse the applications and downloads the uploaded documents. Upon completion for the analysis of the policy application, the underwriter then update on the clients' account whether the application has been approved or rejected.

The underwriter receives the claims applications and does an analysis to check whether the rightful documents have been submitted. If the application is incomplete or the client has submitted few documents then the underwriter rejects the claim and update on the clients' account. If the claim application is complete and the rightful documents have been submitted then the underwriter escalates the claim application to the fund manager for further analysis and update on the clients' account.

Underwriters receive cash or bank statements from the clients and they update the premiums on the clients' accounts.

Fund manager

The fund manager logs in to view the claims submitted by the underwriters. He or she does the further analysis of the claim and give feedback to the client whether it has been approved or rejected. If the claim has been approved the fund manager gives the client information as to how the cash will be given to him or her.

4.2.2 Context diagram

A context (level 0) diagram documents the system's boundaries by highlighting its sources and destinations (Kendall et al, 1988). It provides both data to and gets data from the fund manager, the insurance administrators, client and the insurance manager. The powerful reason for creating a context diagram is its ability to visualise the implications of different boundary assumptions. The goal is to provide an overview of how the users and system would interface within the proposed system and what information would be passed back and forth. Documenting the system's boundaries by drawing a context diagram helped the developer, the user, and the responsible management visualise alternative high-level logical system designs.

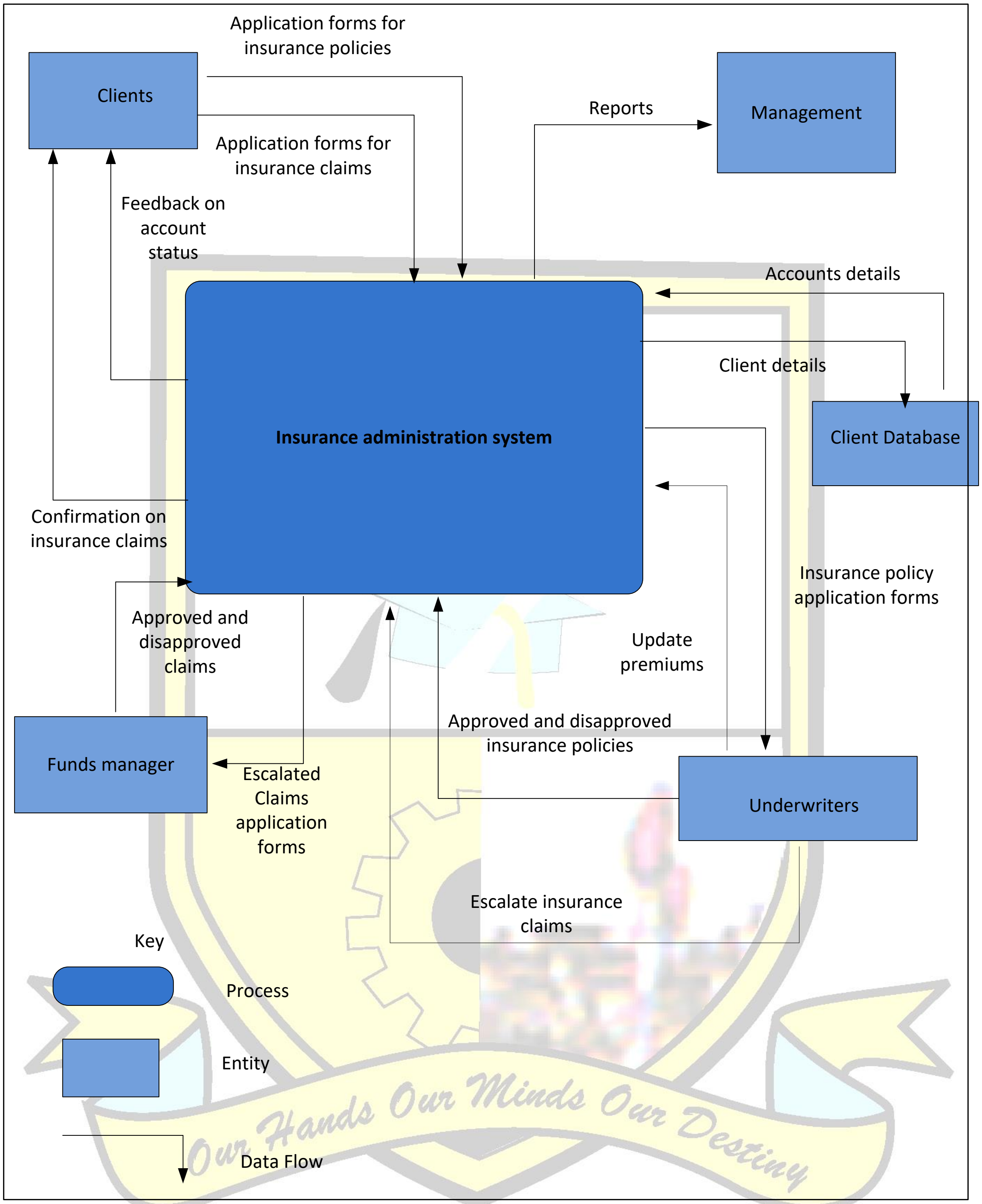

Application forms for insurance policies
Clients
Reports
Management
Application forms for insurance claims
Feedback on account status
Accounts details
Client details
Client Database
Insurance administration system
Confirmation on insurance claims
Insurance policy application forms
Approved and disapproved claims
Update premiums
Approved and disapproved insurance policies
Funds manager
Escalated Claims application forms
Underwriters
Escalate insurance claims
Key
Process
Entity
Data Flow

Figure 4.1 Context Diagram for the proposed system

4.2.3 Data flow diagram

A data flow diagram shows the system's primary processes, data stores, sources, and destinations linked by data flows. A data flow diagram consists of four elements which are the process, flow, store, and terminator. Data flow diagrams (DFDs') reveal relationships among and between the various components in a program or system (Dowell, 2006). DFDs are an important technique for modeling system's high-level detail by showing how input data is transformed to output results through a sequence of functional transformations. DFD is a picture of the movement between external entities and the processes and data stores within a system (Shelly, 2009). Generally, system's primary processes are independent, and thus, are separated from each other by intermediate data stores that suggest how the data are held in some way between processes. The elements of a data flow diagram lead directly into physical design, with processes suggesting programs and procedures, data flows suggesting composites, and data stores suggesting data entities, files, and databases.

Reasons for using the data flow diagrams

Better communication: A logical diagram is easier to use when communicating with the users of the system, because it is centered on business activities.

More stable system: Systems formed using logical data flow diagrams are often more stable than those that are not because they are based on business events and not on a particular technology or method of implementation

Flexibility and maintenance: The new system will be more flexible and easier to maintain because its design is based on a logical model and the business functions are not subject to frequent change. Physical aspects of the system change more frequently than do business functions.

Conversion to physical model: Once the logical data flow diagrams are created, the creation of physical data flow diagrams becomes easy.

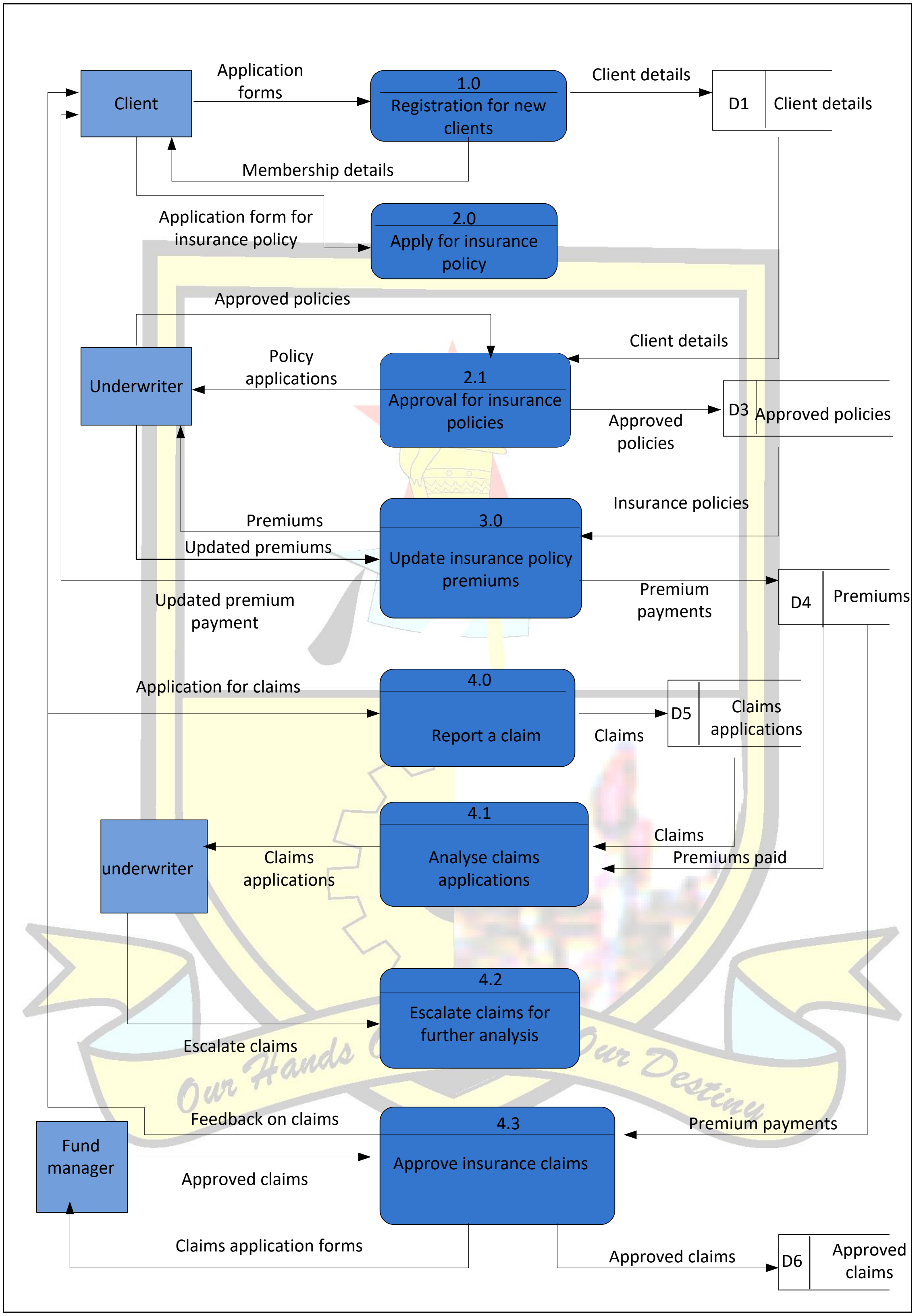
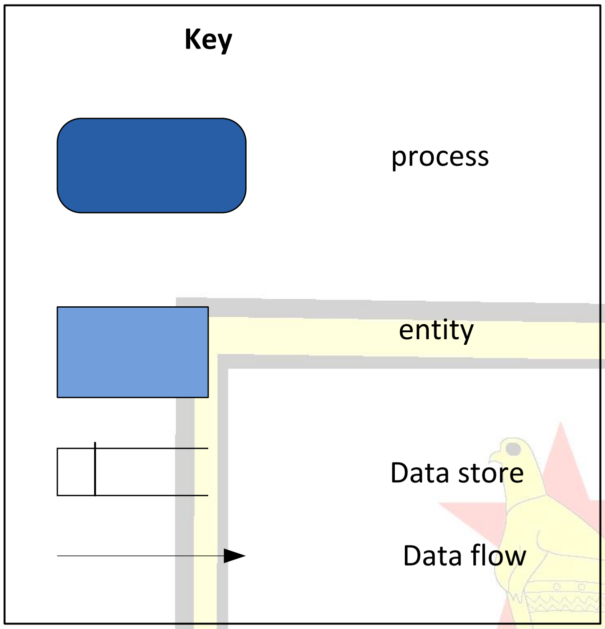


Figure 4.2 Data Flow Diagram for the proposed system



4.2.4 Logical flow chart

A logical flowchart is a diagram that represents the organization and flow of a certain process (Kendall et al, 1988). It shows steps as boxes or levels that connect by lines or arrows. Each box contains a thought or action, and the arrows provide flow and direction. Flowcharts provided a streamlined way to organise, analyse, design and document.

Reasons for using the logic flow charts

- Logic flow charts are used to provide procedure manuals, which can be followed when carrying out certain tasks.
- They provide a pictorial representation of a series of action and therefore are very easy to understand and follow.
- A flowchart provides a visual representation of a concept and can help make it clear. In many ways, a flowchart serves as a blueprint for the information to be presented.

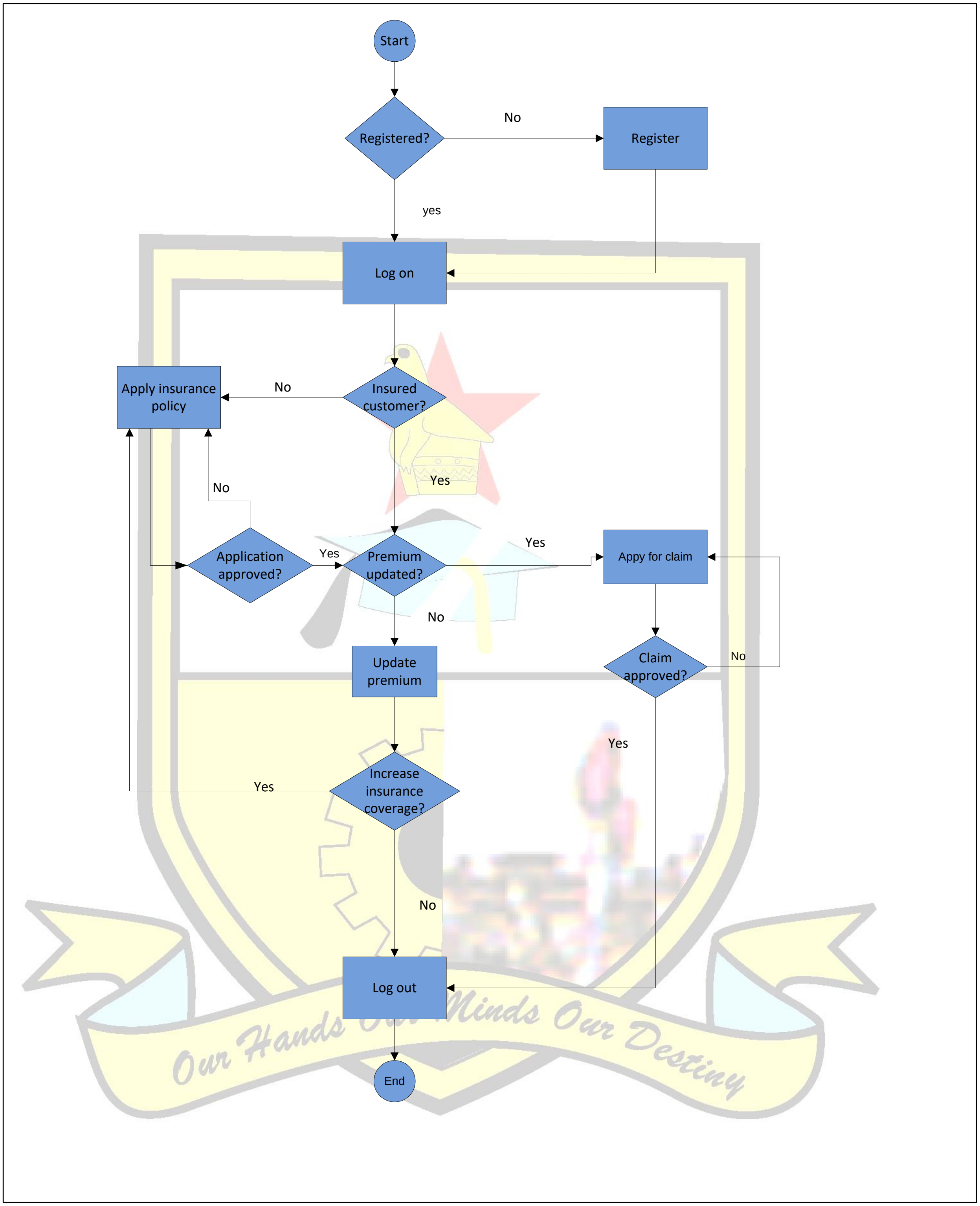
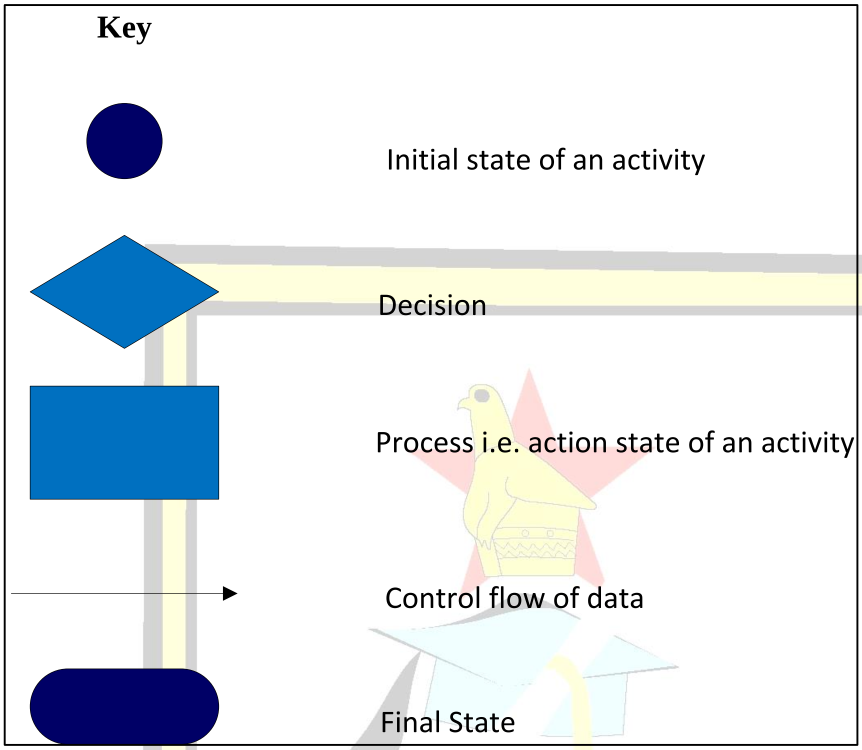


Figure 4.3 Logical Flow Chart for the proposed system


Key
Initial state of an activity
Decision
Process i.e. action state of an activity
Control flow of data
Final State

4.3 Architectural design

Architectural design is the process for identifying the subsystems making up a system and the framework for sub-system control and communication (Sommervill, 2004). The output of this design process is a description of the software architecture. Architecture design looks at the hardware setup and configurations of the proposed system and shows where the major components of the system will be connected and how the several machines and people that make up the system will be connected.

Advantages of architectural design are:

Stakeholder communication: architecture may be used as a focus of discussion by system stakeholders.

System analysis: it can be used for analysis to evaluate if the system could meet its non-functional requirements.

Large scale reuse: the architecture may be reusable across a range of systems.

The insurance administration system was developed using three-tier client-server architecture with: the presentation on the client side, the application on the web server and data management on the database (DB) sever.

The first tier component is responsible for presentation and is the user interface. The user interface to insurance administration allows the user to interact with the system. It is a simple text-oriented interface. These client components enable the user to interact with the second-tier processes in a secure and intuitive manner. Application Server supports several client types. Clients do not access the third-tier services directly. The second-tier process, commonly referred to as the application layer, manages the business logic of the application, and is permitted access to the third-tier services. Application Server provides the application logic layer in three-tier architecture, enabling client components to interact with data resources and legacy applications. The application logic layer is where most of the processing work occurs. Multiple client components access the second-tier processes simultaneously, so this application logic layer must manage its own transactions. The third-tier which is the DB-server tier is to manage persistency of certain data/information and to execute the database transaction services. These services are protected from direct access by the client components. Interaction must occur through the second-tier processes.

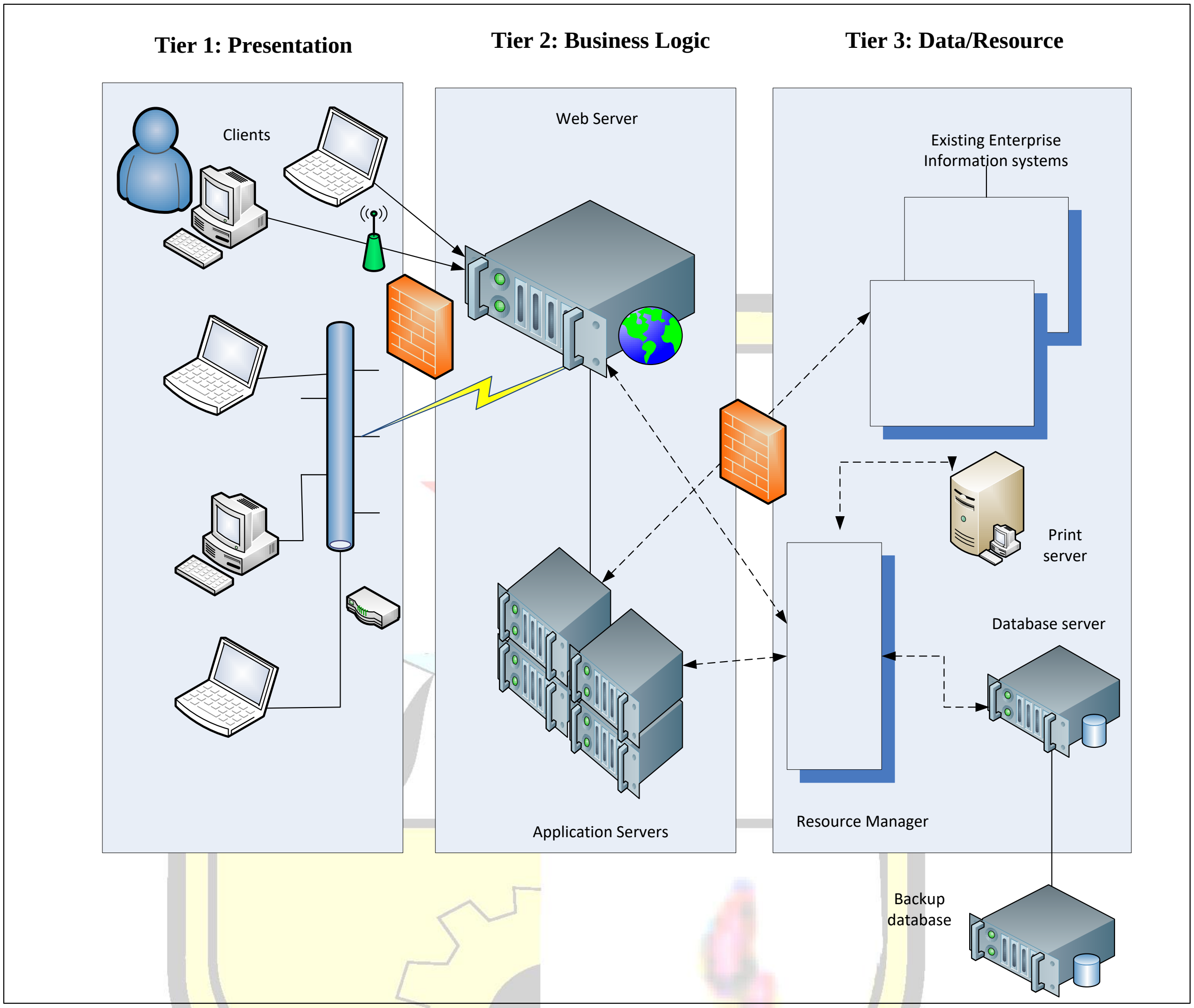
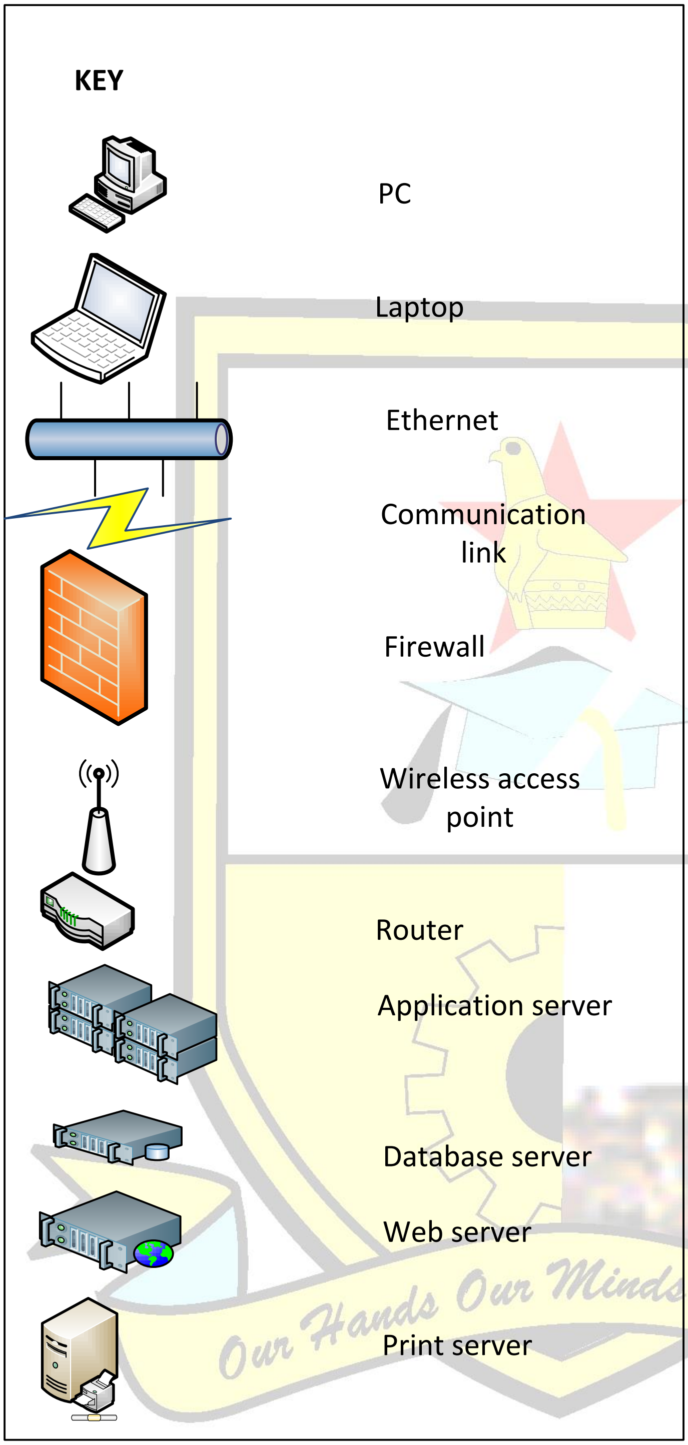


Figure 4.4 Architectural design of the proposed system


KEY
PC
Laptop
Ethernet
Communication link
Firewall
Wireless access point
Router
Application server
Database server
Web server
Print server

4.4 Physical design

This is concerned with the design of how the hardware and software components of the proposed system are going to be laid out and how they will be interacting. It was mentioned earlier on in the feasibility study that some hardware and software specifications had already been met and they were readily available for the new system to function well, thus no additional costs are needed other than the ones identified before. The database will be connected to the Wide Area Network (WAN) to enable easy access by the clients from different geographical locations so as to facilitate the carrying out of transactions successfully at the clients' convenience.

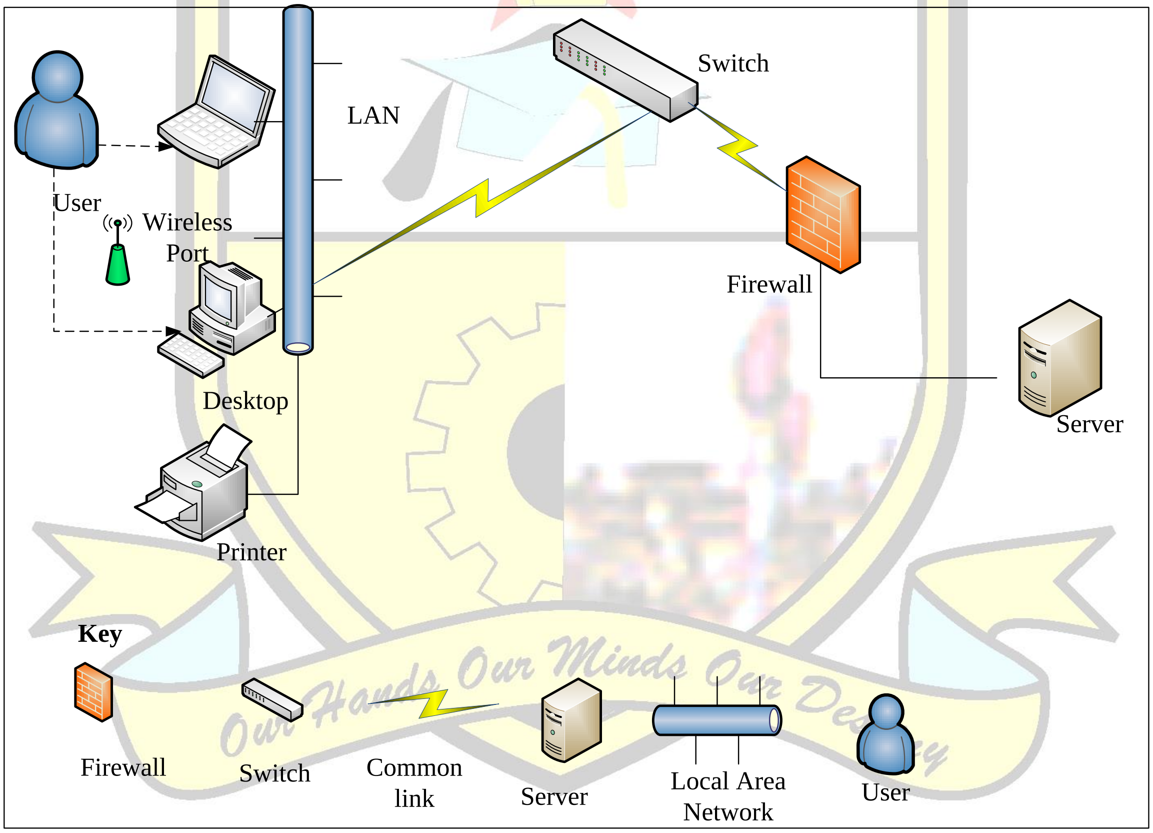


Figure 4.5 Physical Design

The diagram above shows the overall physical and setup design for the new system. There is one machine that functions as the application server; the users' workstations access this application server through the OMs' LAN. It is the application server that communicates with the database server from which all the manipulations and data requirements are handled.

Table 4.1 Hardware and software specified

Hardware	Software
Database server (Dell PowerEdge R410)	(ESXi Installer, VMware vsphere client)
Underwriters' computers (Compaq 500B)	Microsoft Windows Server 2008
Tape drives	Wamp (manager) server
Patch Panel- 24 Port	Macromedia Dreamweaver 8.0
20m UTP CAT, 35 fly leads, patch codes, RJ 45	Web browser – Mozilla Firefox or Internet Explorer

4.5 Database design

A database is not merely a collection of related files arranged in a logical way, but a central source of data meant to be shared by many users for a variety of purposes. This software also allows the retrieval of data and the generation of reports. The database design's goal is to completely capture all a business' data storage needs; its product should be an accurate and easy-to-use database that performs well (Elmosri et al, 2011). The implicit goals of database design are to meet all the information (contextual) requirements of the entire spectrum of users in the development of the Insurance Administration System and to provide a natural" and easy-to-understand structuring of the information content

Three very basic phases of database design that the developer undertook include:

I. **Requirements gathering-** this involved conducting meetings with the end users to establish the requirements for the proposed database. Information obtained include business rules and processes, Information about the current database being used and future needs of the business as related to the database

II. **Data modeling-** The developer visually represented the data for Old Mutual, and then eventually converted the business model into a data model. The data model generated was used to ultimately create the tables, views, and other objects that comprise the database.

III. **Database design and normalization-** The business model (logical model) was converted into a physical model (tables).

From the requirements gathering the developer then used a relational database. The relational database model was then used in designing the database. This approach was chosen because of its simplicity in describing data elements and the relationships between them. Data was entered and stored via web forms. The data was held in such a way that changes to the structure of the database did not affect any of the programs that access the data.

Benefits of the relational model are as follows:

• Data is accessed very quickly.

• The database structure is easy to change.

• The data is represented logically; therefore users need not understand how the data is stored.

• It is easy to develop complex queries to retrieve data.

• It is easy to implement data integrity.

Reasons for using the database management system (DBMS)

- Improved security of the database by the use of the password since only the administrator will be able to access private and confidential information.
- Data independence is supported i.e. data at one level can be changed without affecting the other levels of the database

- Database Management System (DBMS) provides backup and recovery services.
- Data consistency is maintained by implementing the database management system and information is validated as it will be inputted in the database.
- Program is not tied to physical data storage characteristics.

4.5.1 Database architecture design

The ANSI/SPARC architecture was used to design the database. The database consists of three schemas which are the conceptual layer, physical layer and application layer. The schema is a description of the database specified during database and the instance is the data that is stored in the database at a given time (Dennis et al, 1999). The database architecture design of the proposed system is shown in Figure 4.6

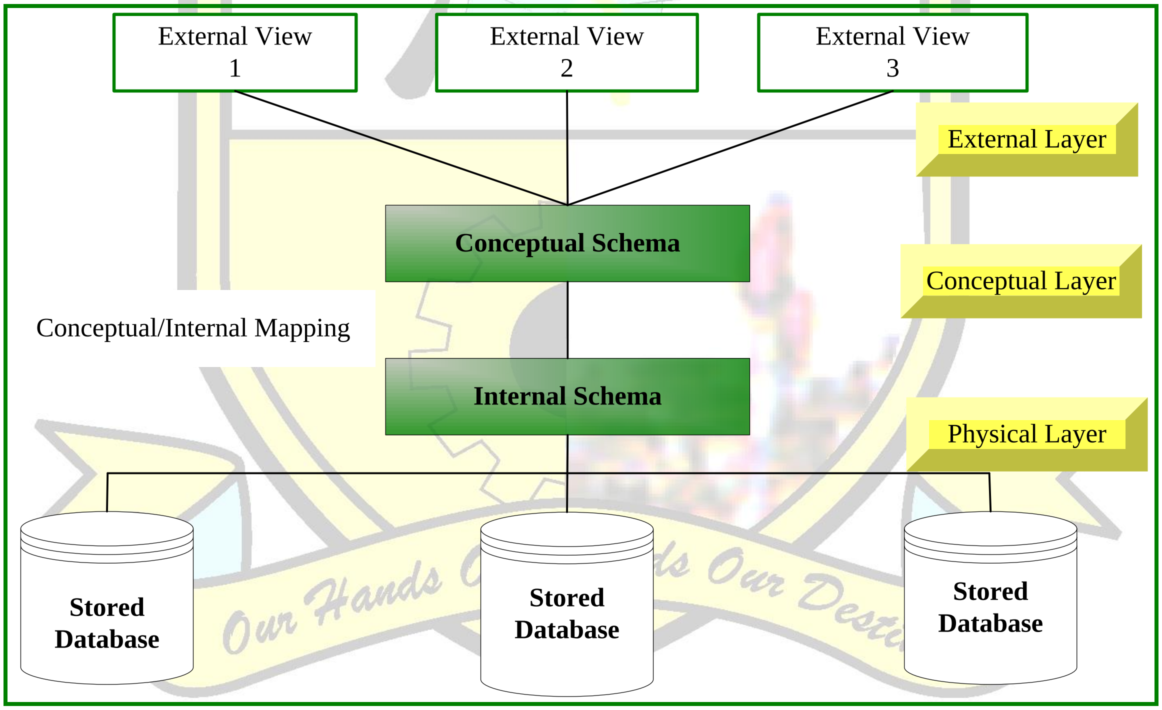


Figure 4.6 Database Architecture (3 Level Schema)

External Level

This is how the users view the database. External level describes that part of the database that is related to the users. Different users may generate many views and some of them are completely different from the way the database is built. Each user can access the data, but have their own view of the data independent of other users. Conceptual schema changes do not affect external views. A view by the underwriters is different from the view of the clients and the administrator. Legal underwriters log in the system and they only view legal related issues. The legal underwriter only activates legal applications forms and also analyse legal claims applications before escalating the claims to the funds manager or back to the client. The fund manager only views the claims that have been escalated by the different underwriters. Each underwriter views the insurance policy that is assigned to him or her. Legal underwriters cannot view hospital insurance policies. Each underwriter is confined to their area of specialisation.

Conceptual Level

This is the community view of the database and describes what data is stored in the database and its relationships. Conceptual level contains the logical structure of the database as seen by the Database Administrator (DA). It acts as the complete view of the data requirements of the organisation. Single shared data representation for all applications and users which is independent of physical data storage. Users do not have to understand physical data representation details.

The DBA can change the storage structures without affecting users or applications. Physical data independence - conceptual schema is not affected by physical changes such as adding indexes or distributing data. Data for the Insurance Administration System will be organised in such a way that the clients' details will be stored in its own data store; the application forms for the insurance policies will be stored in its data store. Claims that will be applied for will be stored in a different data store as well. Premiums will be stored in another data store different from the three data stores. The administrator creates relationships to the stored data in the different data stores. When a client wants to view his premiums with their respective insurance policies he or she is able to do that due to the relationship that is created between the clients' details, insurance policy and the premium paid.

Internal Level

This is the physical representation of the database and has an internal schema that describes how the data is stored in the database. It is the physical representation of the database on the computer. How the data is stored. It provides standard facilities for interacting with operating system for space allocation and file manipulation. Different insurance policies will be stored in accordance with the respective insurance policy facet. The premiums that will be paid will be stored in such a way that the client is able to view the premium with respect to the insurance policy that he or she has.

4.5.2 Entity relationship diagram

An Entity Relationship Diagram (ERD) is a snapshot of data structures. ERD shows the entities in a database and relationships between tables within that database (Dowell, 2006). ERD can also be defined as a data modeling technique that creates a graphical representation of the entities, and the relationships between entities, within an information system (Kendall et al, 1988). The patterns helped the developer focus on how the database actually works with all of the interactions and data flows. The main components of an ERD are:

An entity: may be defined as a thing or object which is recognized as being capable of an independent existence and which can be uniquely identified. It is that aspect of the real world which can be distinguished from other aspects of the real world for example a client, administrator.

A relationship: captures how two or more entities are related to one another. Relationships can be thought of as verbs, linking two or more nouns.

Attributes: are the data or features we collect about the entities that better describes an entity.

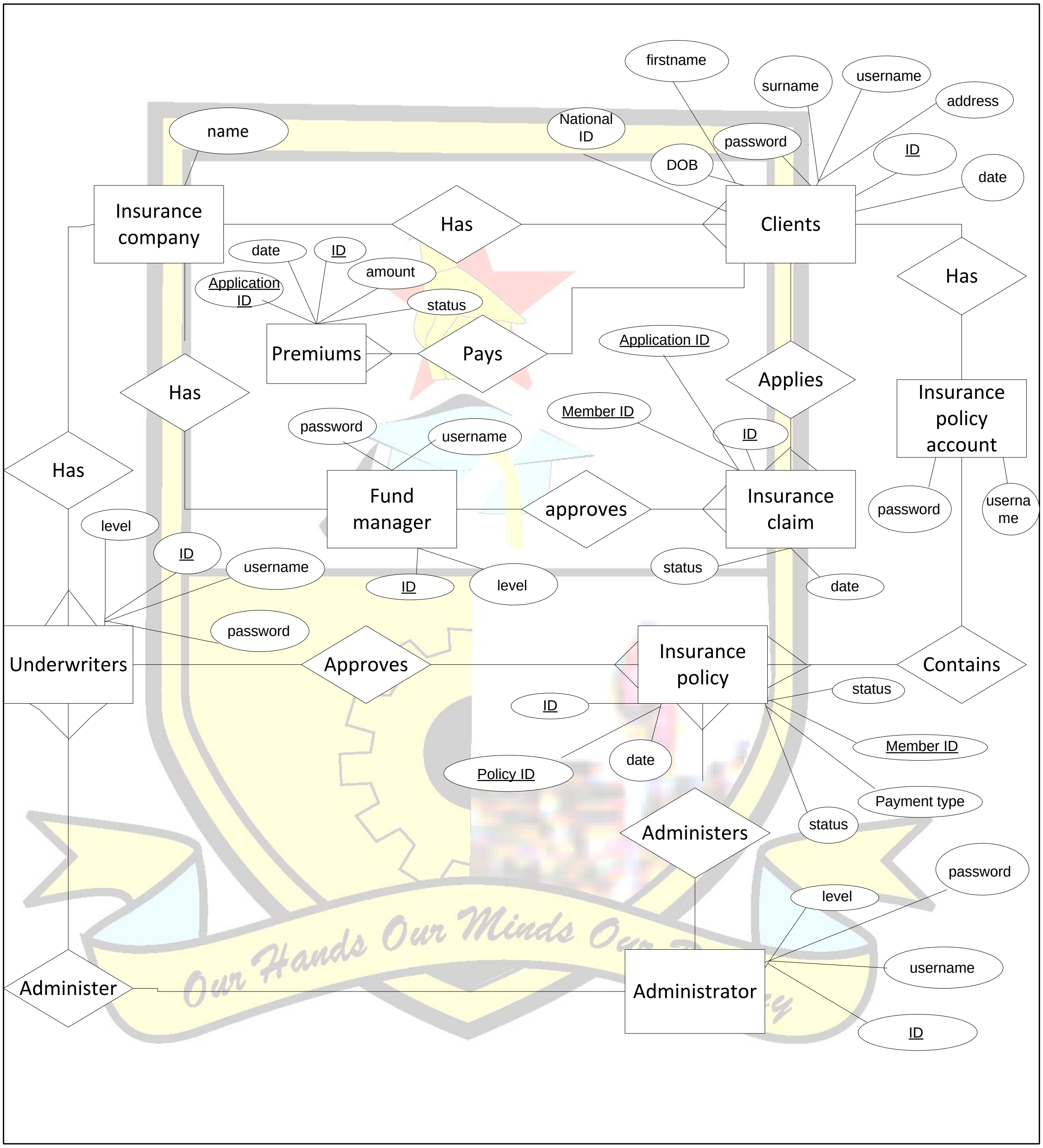
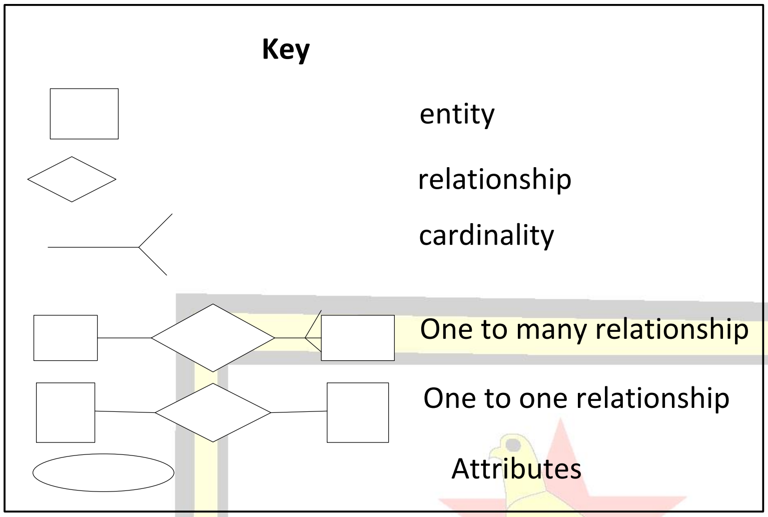


Figure 4.7 Entity Relationship Diagram for the proposed system



4.5.3 Enhanced entity relationship diagram

The EER diagram is a high level or conceptual data model incorporating extensions to the original ER model used in the design of databases. (Elmasri et al, 2011). The developer used the EER because it reflects more precisely the properties and constraints that are found in more complex databases. The EER is created to design more accurate database schemas. It reflects the data properties and constraints more precisely. EER model includes all modeling concepts of the ER model. The EER includes the following:

I. Subclasses and superclasses
II. Specialization and generalisation
III. Attribute and relationship inheritance

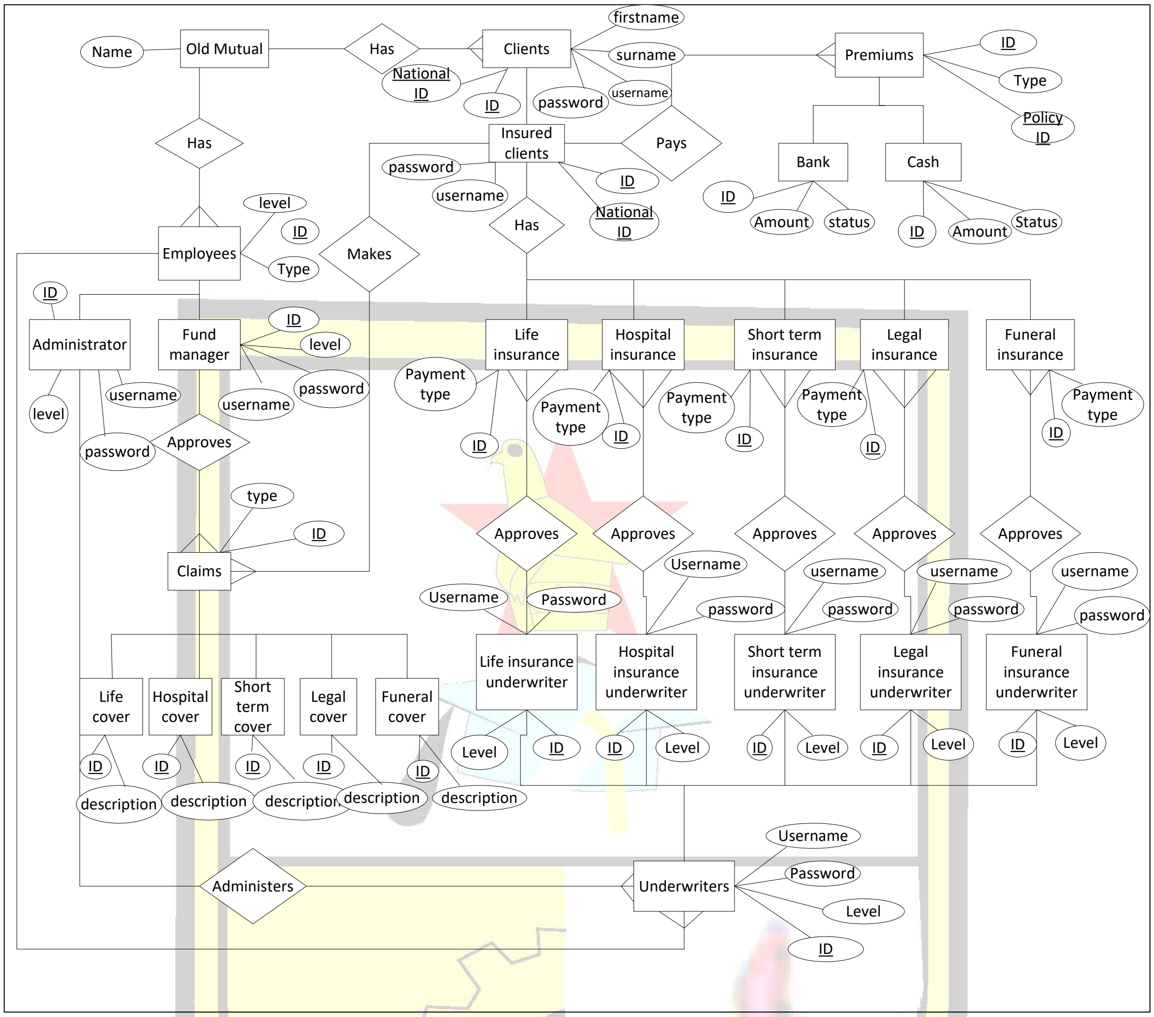
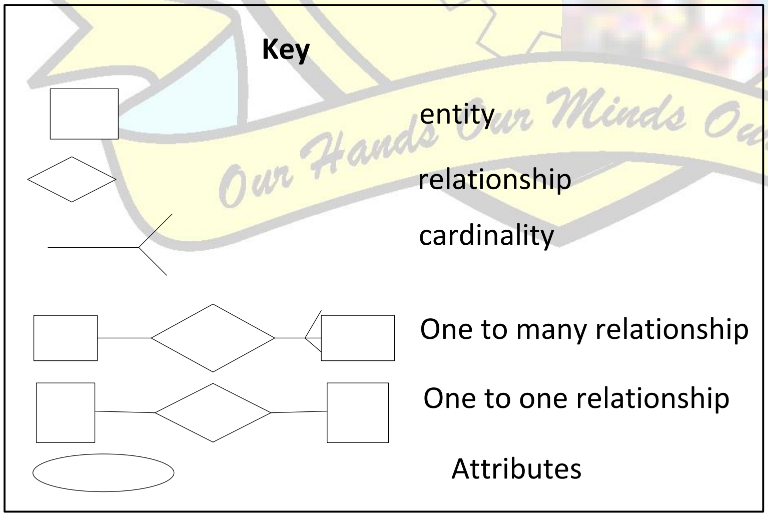


Figure 4.8: Enhanced Entity Relationship Diagram for the proposed system



4.5.4 Database tables

Tables are the fundamental logical building blocks of relational databases. A table is a grouping of columns of data that pertains to a single particular class of people, things, events, or ideas in an enterprise about which information needs to be stored or calculations made (Stephens, 2001). In database design, tables are derived from entities that were defined during logical modeling. Likewise, columns are derived from these entities' attributes.

Table 4.2: Clients' table

Field	Type	Size	Description
Username	Varchar	(100)	Name used to login to the system
Password	Varchar	(100)	Authentication to the username
FirstName	Varchar	(100)	Client's first name
Surname	Varchar	(100)	Client's last name
Nationalid	Varchar	(100)	Primary key, client's national identity number
Address	Varchar	(100)	The clients residential place
Sex	Char	(5)	Client's sex
DOB	Date	(20)	The day to which the client was born
Client ID	bigint	(20)	Client's account number
Date	date	(20)	Date of registration

Table 4.3: Login table

Field	Type	size	Description
Username	Varchar	(30)	Name used to login to the system
Password	Varchar	(30)	Authentication to the username
Level	Number	(2)	Access level

Table 4.4: Policy coverage table

Field	Type	Size	Description
Client id	Bigint	(20)	Client's account number
Policy id	Bigint	(20)	Policy number for the client
Payment type	Varchar	(100)	Method of payment by client
Document	Varchar	(100)	Proof of application
Date	date	(20)	Date of application
Status	bigint	(20)	Status whether approved or declined

Table 4.5 Claim application

Field	Type	Size	Description
Client id	Bigint	(20)	Client's account number
Application id	Bigint	(20)	Automatic generation
Description	Varchar	(100)	Detailed details of the claim
Document	Varchar	(20)	Proof of application
Date	Date	(20)	Date of claim
Status	Bigint	(20)	Status whether approved or declined

Table 4.6 Creation of underwriter accounts

Field	Type	size	Description
Username	Varchar	(30)	Name used to login to the system
Password	Varchar	(30)	Authentication to the username
Level	Number	(2)	Access level

4.6 Program design

Program design takes into account the design of classes, functions, and modules of the proposed system (Shelly, 2009). This phase takes into consideration the design of queries if any are to be used within the new system. The program design therefore involves the designing of modules, classes and functions of the proposed system. The structured design methodology is based on the breaking down of the system into subsystems which are further broken down into modules. The program design is illustrated using three diagrams namely:

I. Class Diagram

II. Sequence Diagram

III. Package Diagram

4.6.1 Class diagram

A class diagram shows how the different entities (people, things, and data) relate to each other; in other words, it shows the static structures of the system (Shelly, 2009). A class diagram was used to display logical classes. The upper section shows the class's name; the middle section contains the class's attributes; and the lower section contains the class's operations (or "methods").

The Class diagram illustrates how the expected states and behaviors of the expected classes of the proposed system will interact with each other.

Class name –refers to people, events, places and things about the system.

Attributes- are properties of the class about which we want to capture information.

Operation- represents the actions or methods that a class can perform.

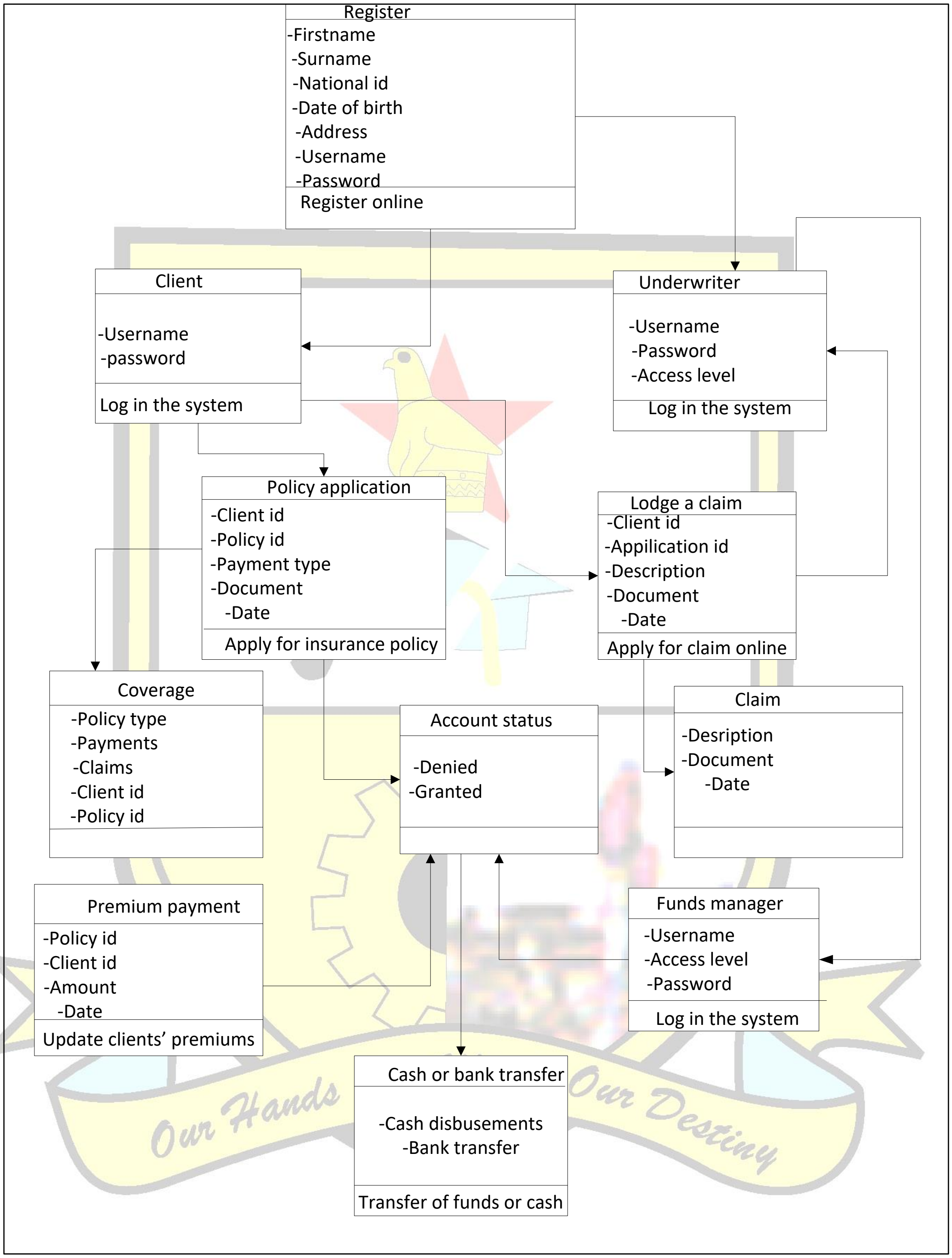


Figure 4.9: Class Diagram for the proposed system

4.6.2 Sequence diagram

A sequence diagram is a visual representation of how objects in a system interact. A sequence diagram is a form of interaction diagram which shows objects as lifelines running down the page, with their interactions over time represented as messages drawn as arrows from the source lifeline to the target lifeline (Navathe, 2011). An important characteristic of a sequence diagram is that time passes from top to bottom the interaction starts near the top of the diagram and ends at the bottom. These interactions are captured in a single scenario, which is a specific path of behavior in a system for a particular use case or set of use cases.

Lifelines- A lifeline represents an individual participant in a sequence diagram. A lifeline will usually have a rectangle containing its object name.

Messages- Messages are displayed as arrows. Messages can be complete, lost or found; synchronous or asynchronous; call or signal.

Execution Occurrence- A thin rectangle running down the lifeline denotes the execution occurrence, or activation of a focus of control.

Self Message- A self message can represent a recursive call of an operation, or one method calling another method belonging to the same object. It is shown as creating a nested focus of control in the lifeline's execution occurrence.

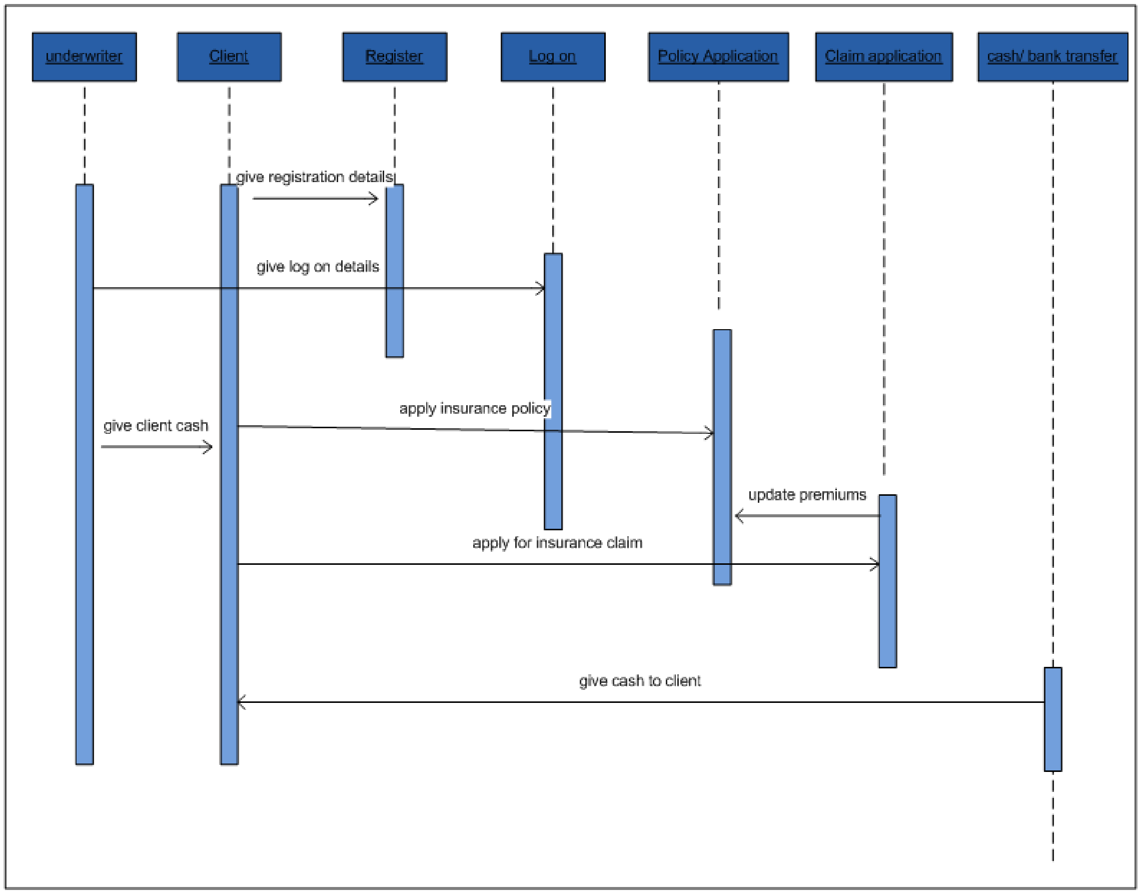


Figure 4.10 Sequence diagram for the proposed system

4.6.3 Package diagram

Package diagrams organise the elements of a system into related groups to minimise dependencies among them. A Package Diagram is a modular structure of the system that depicts the breakdown of the system modules and their interaction (Ryan, 2001). The different classes within the Insurance Administration system were grouped together into packages to reduce complexity and the developer to have a better understanding of the proposed system.

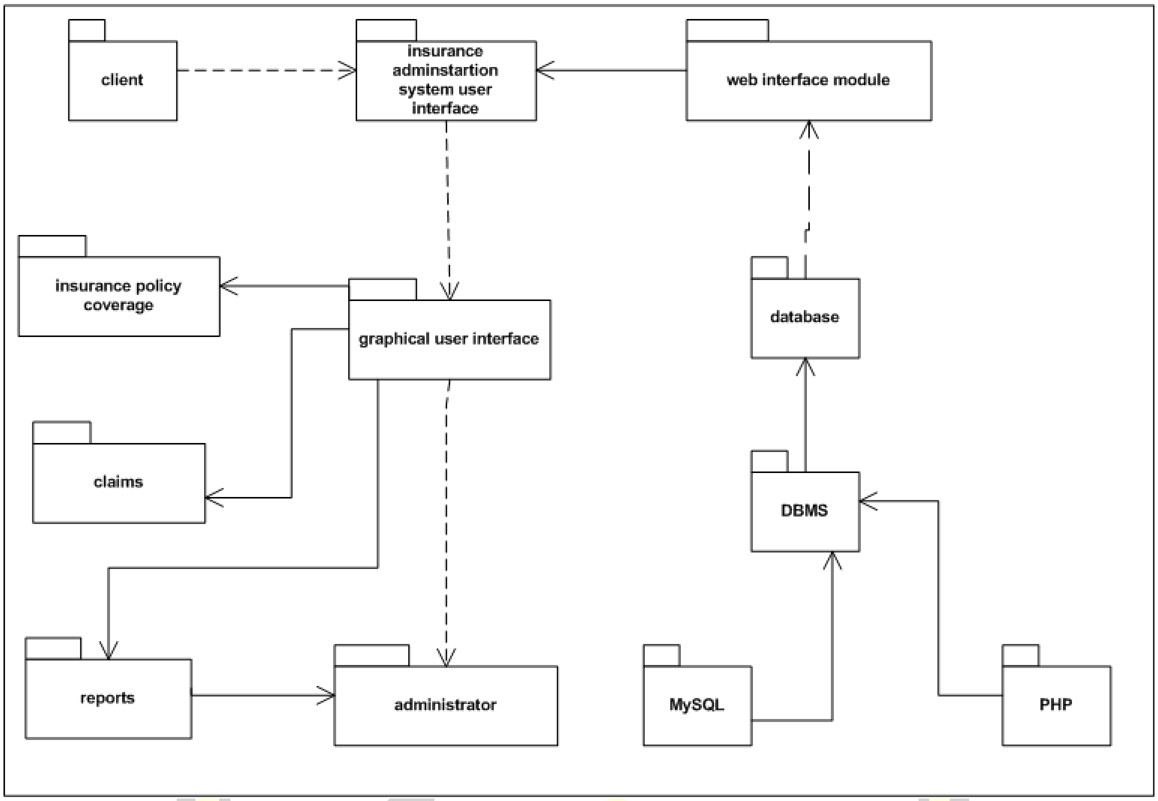


Figure 4.11 Package Diagram for the proposed system

4.7 Interface design

Interface design illustrates the graphical user Interface that interacts with the users each time they use the system, this is designed in a way that allows easy navigation of the entire system. It outlines the design of the menus and forms for data input into the system. It also gives an outline of the structure of the input and output of the system (Sommerville, 2004).

A three main menu is a diagrammatic illustration of what the entities in the system will be doing in the system.

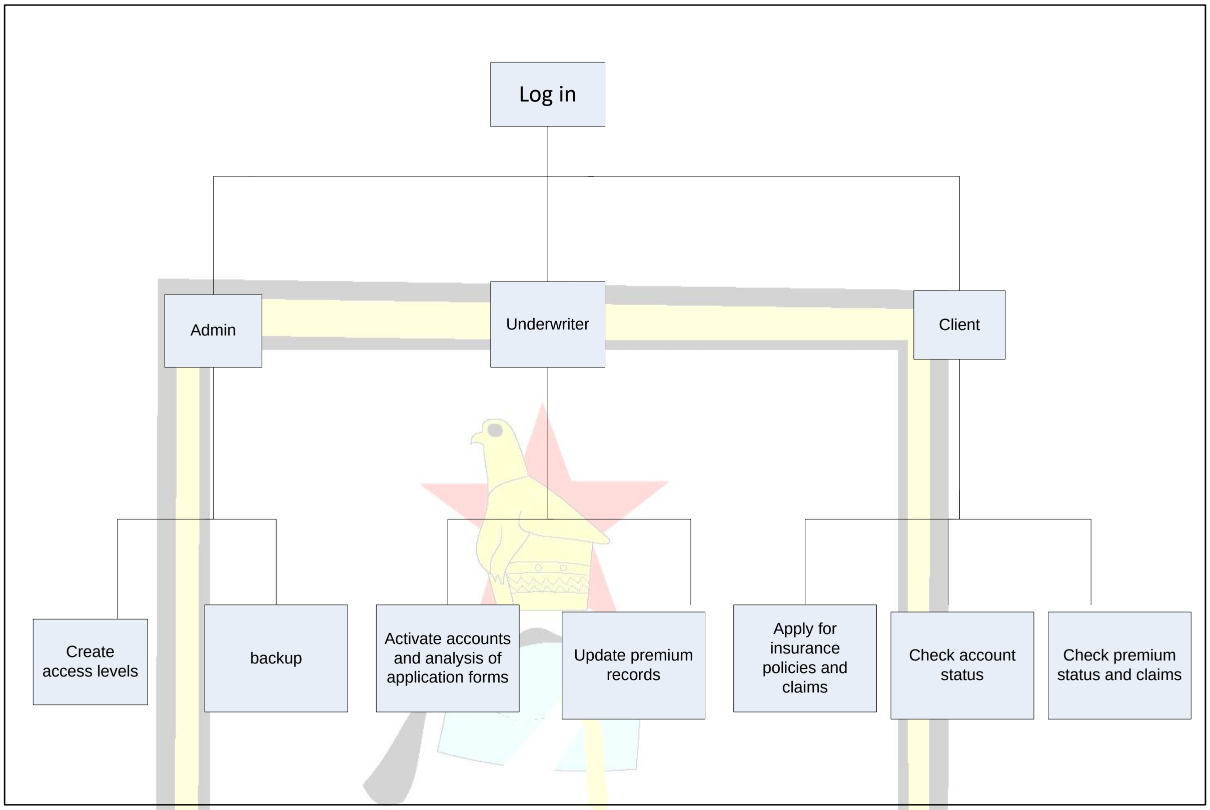

Log in
Admin
Underwriter
Client
Create access levels
backup
Activate accounts and analysis of application forms
Update premium records
Apply for insurance policies and claims
Check account status
Check premium status and claims

Figure 4.12 Three main menu

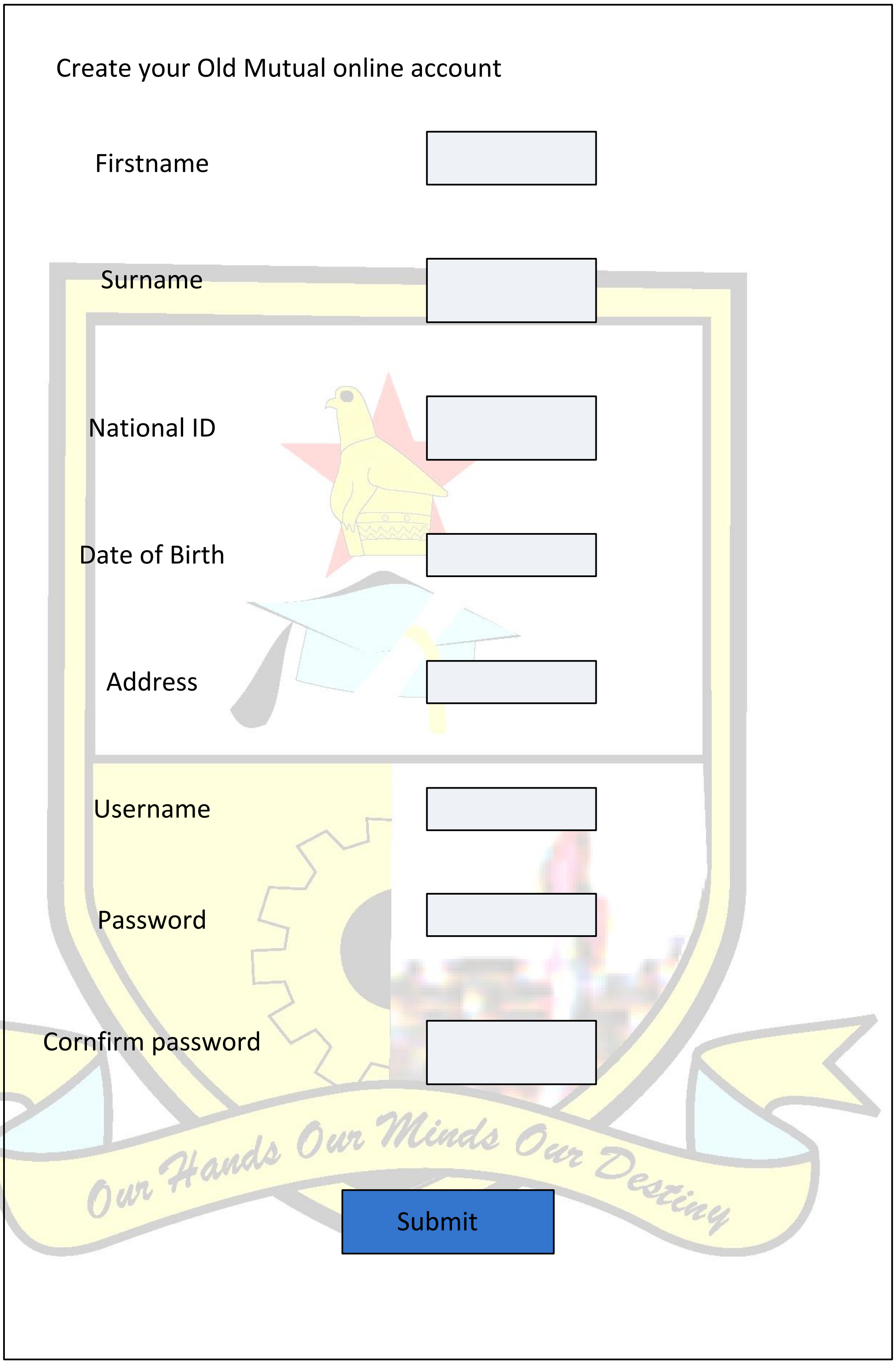
4.7.1 Input design

The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use while retaining the privacy. Input design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

Client registration form

The new client creates his or her account in the client registration form.



Create your Old Mutual online account

Firstname

Surname

National ID

Date of Birth

Address

Username

Password

Cornfirm password

Figure 4.13 Client registration form

Login form

The log in form serves as the entry point into the system. It will accept username and password then validate them against what is in the database. Clients can only manage to login after registering with the OM so that they can apply for insurance policies and also the admin to access client information.

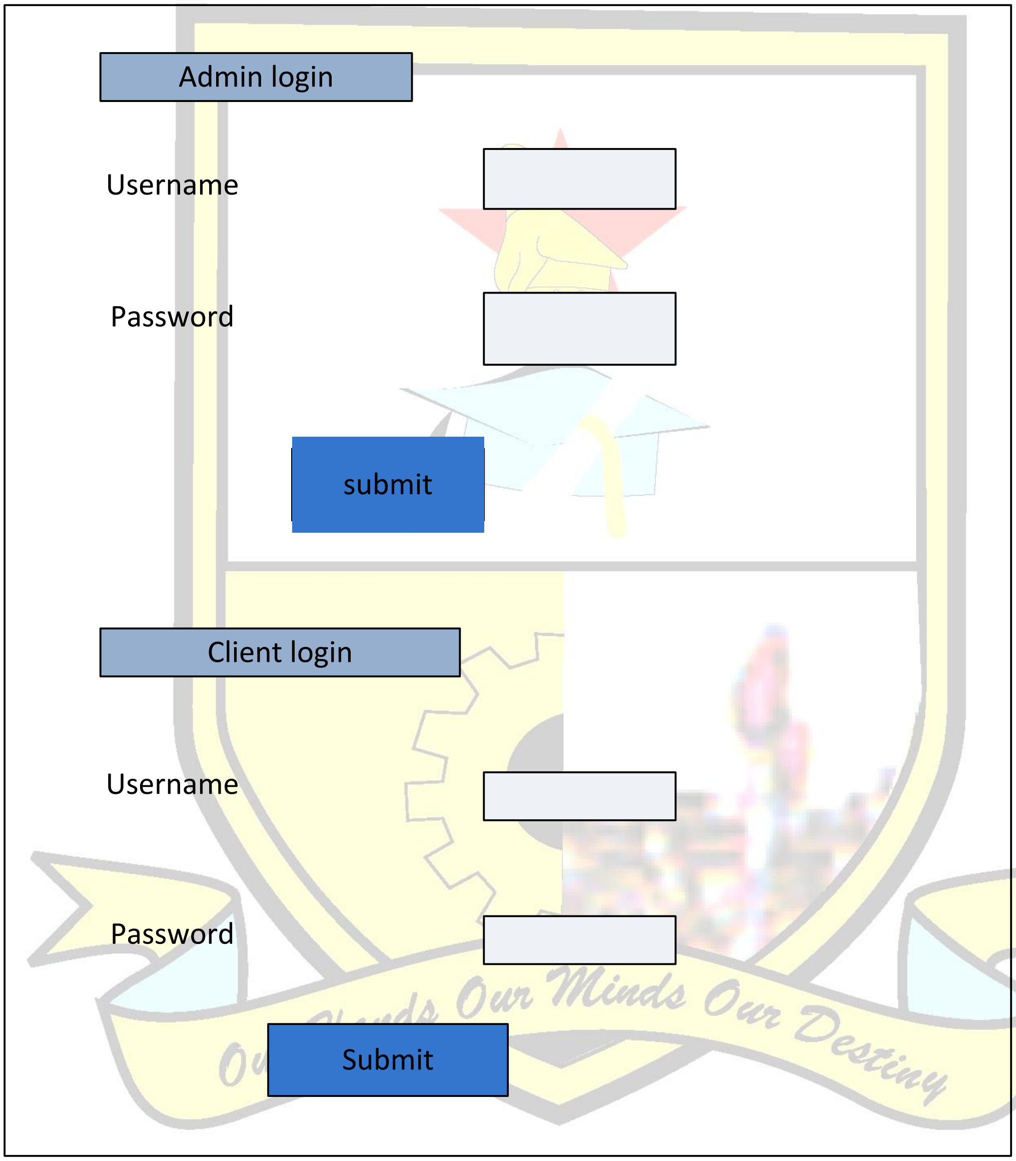
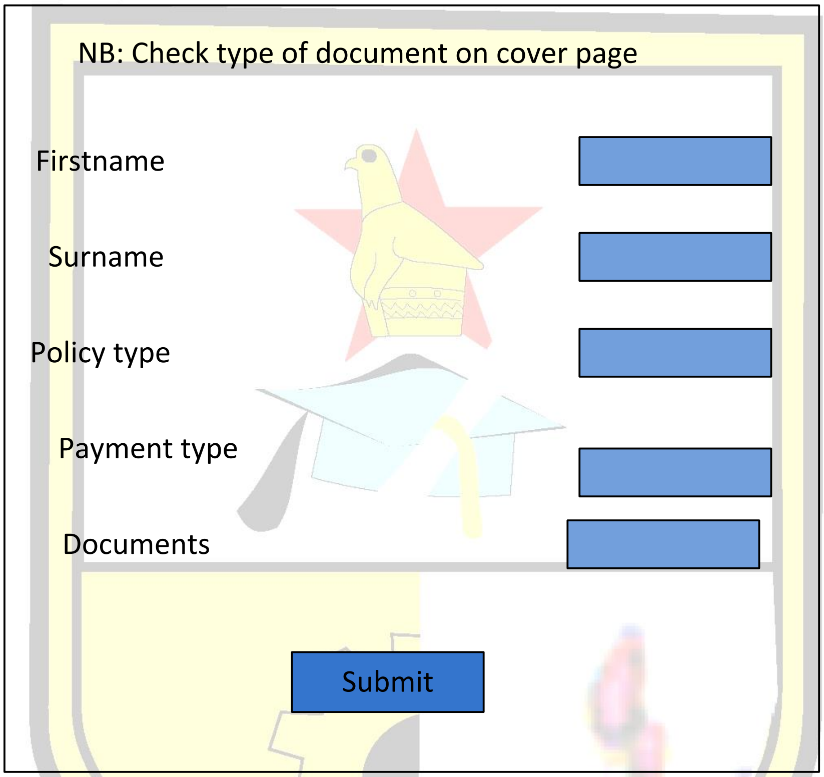


Figure 4.14 Log in Form

Insurance policy application form

The application form will enable clients who are not yet insured to apply for an insurance policy. The form also allows for the insured clients to increase their insurance policies by addition of other insurances.



NB: Check type of document on cover page

Firstname

Surname

Policy type

Payment type

Documents

Figure 4.15 Insurance policy application form

Creation of underwriters form

The form allows the administrator to create and give access levels to different underwriters depending on their area of expertise.

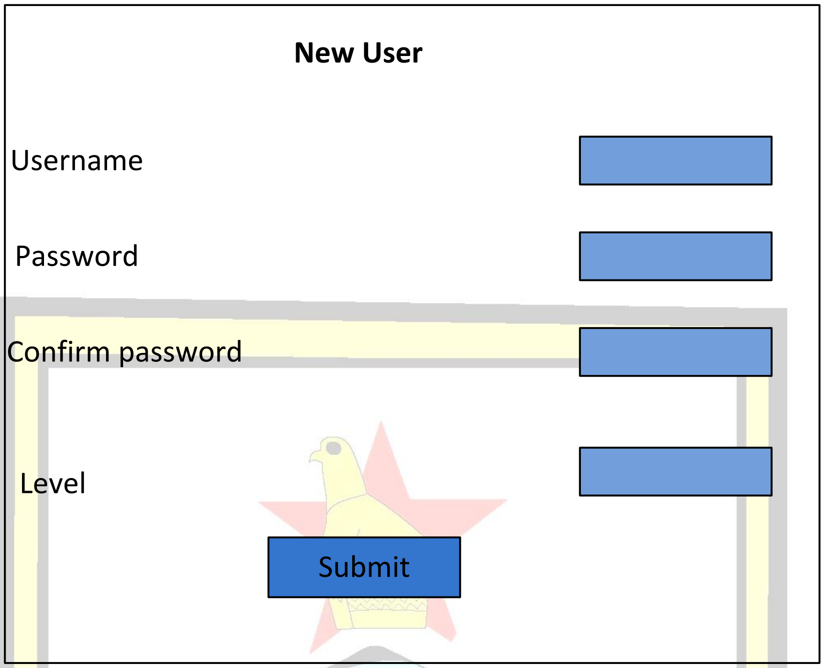


Figure 4.16 New user creation form

Premium payments form

The form allows underwriters to update the clients’ accounts by entering the amount paid for that month.

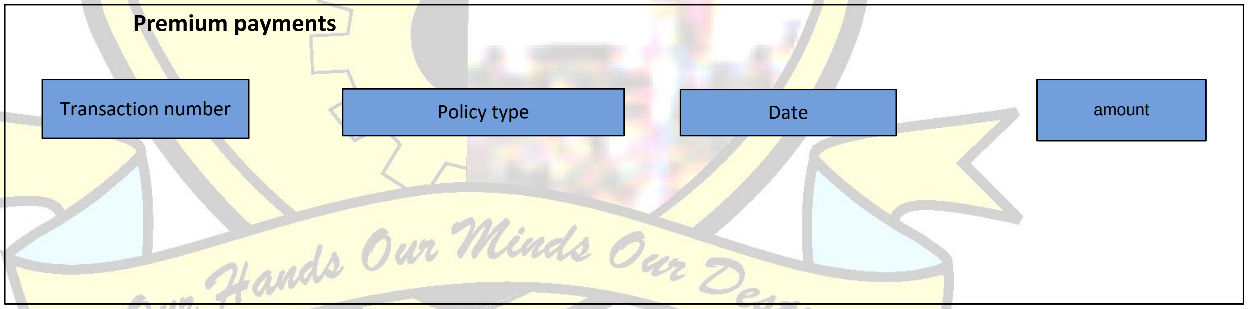


Figure 4.17 Premium payments form

Claims application form

The form allows the client to submit the different type of claims. The client gives the details pertaining to the reported claim.

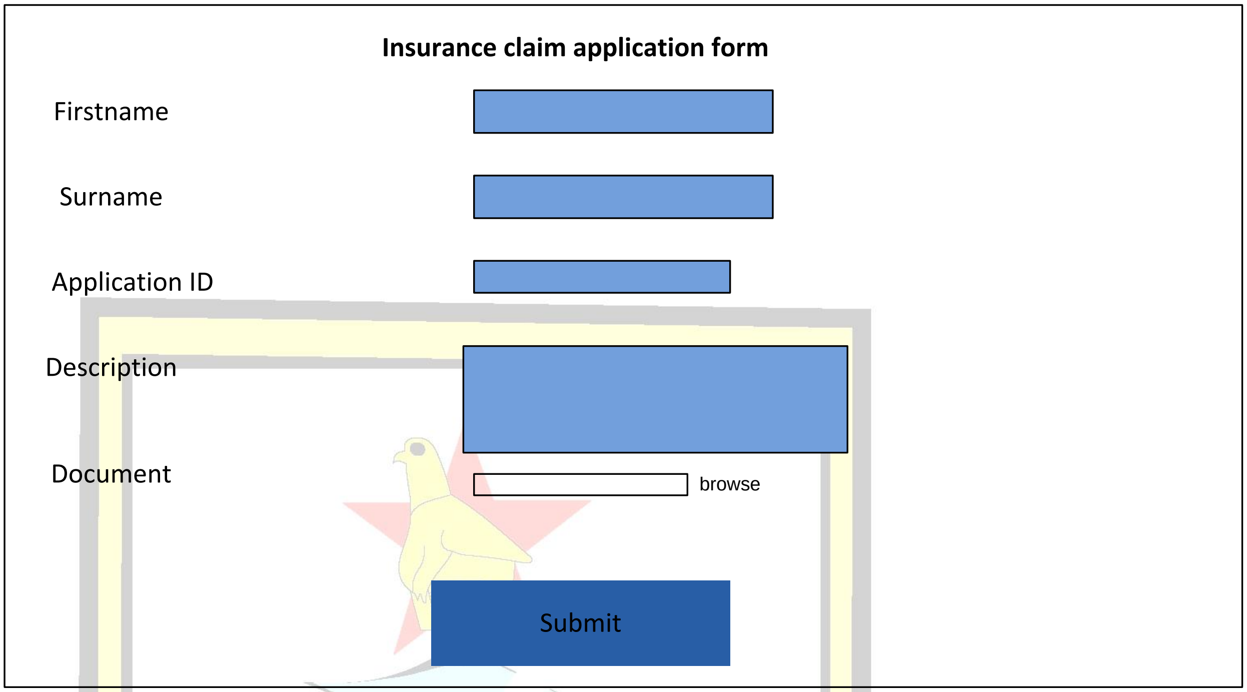


Figure 4.18 Claim application form

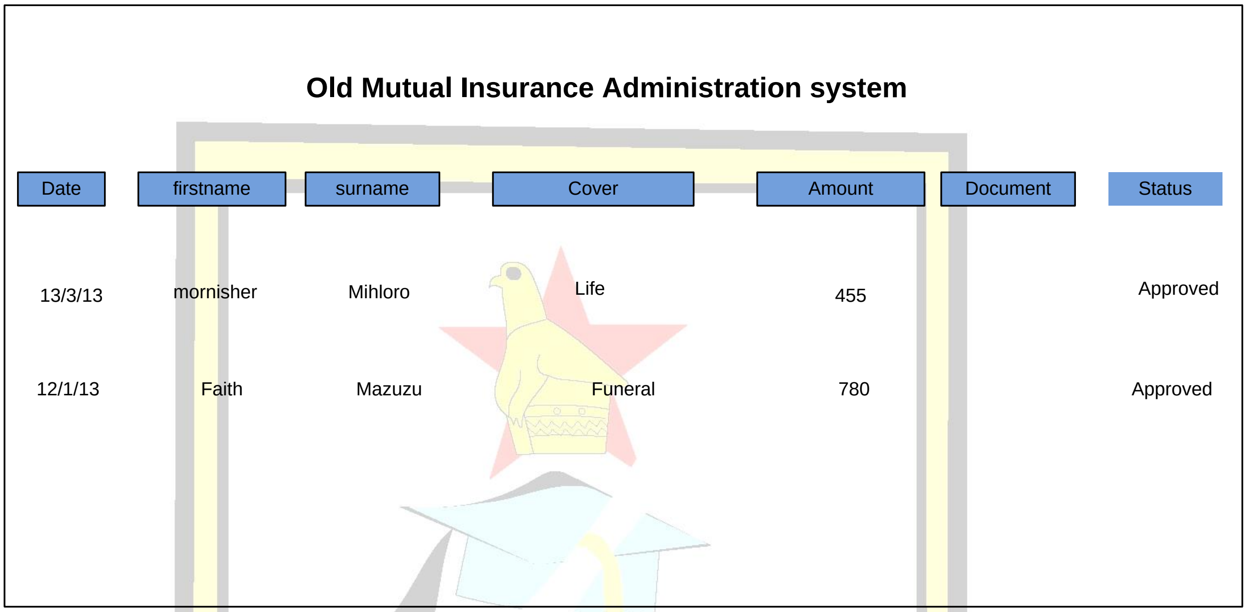
4.7.2 Output design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In output design it is determined how the information is to be displayed for immediate need and also the hard copy output. It is the most important and direct source of information to the user (Dowell, 2006). Efficient and intelligent output design improves the system's relationship to help user in decision-making. The outputs can either be in the form of graphical interface output or electronic reports which can be printed out.

The proposed system's output was designed and developed in an organised, well thought out manner, while ensuring that each output element will enable users to use the system easily and effectively.

Premiums Report

The premiums report allows the administrator to view all the premiums that were paid by the client to the corresponding type of insurance facet.



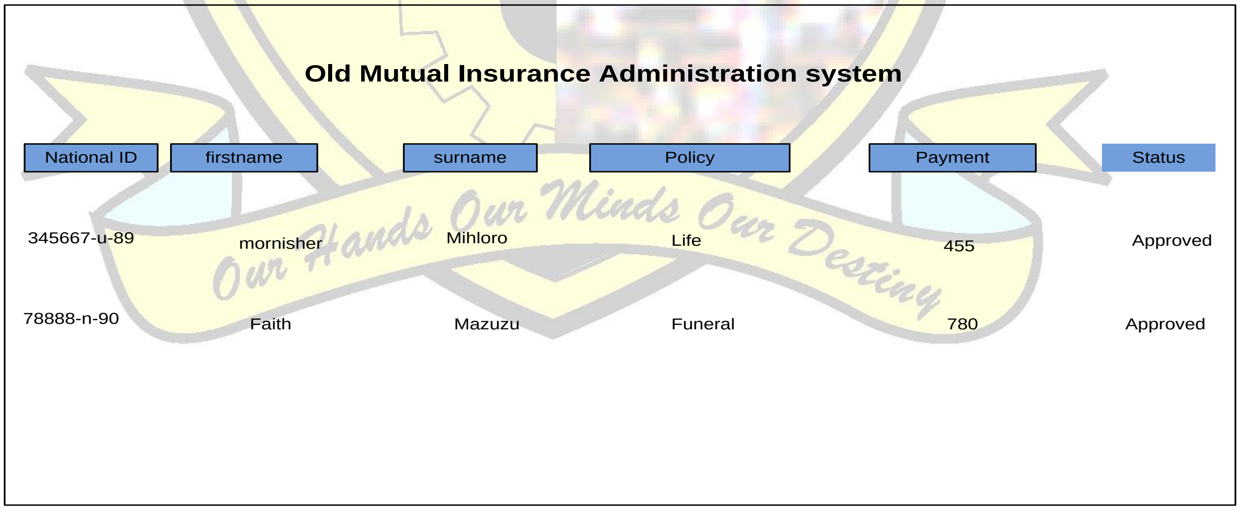
Old Mutual Insurance Administration system

Date	firstname	surname	Cover	Amount	Document	Status
13/3/13	mornisher	Mihloro	Life	455		Approved
12/1/13	Faith	Mazuzu	Funeral	780		Approved

Figure 4.19 Premiums Report

Claims Report form

The administrator views all the claims that were made by the client and the corresponding amounts of the claims.



Old Mutual Insurance Administration system

National ID	firstname	surname	Policy	Payment	Status
345667-u-89	mornisher	Mihloro	Life	455	Approved
78888-n-90	Faith	Mazuzu	Funeral	780	Approved

Figure 4.20 Claims Report

4.8 Conclusion

The design phase enabled the developer to have an overview of the proposed system. All the necessary designing was carried out in this phase i.e. inputs, processes and outputs of the new system were designed. The flow of activities of the proposed system as well as physical setup of the application, which assisted in the coding and construction paved way for the implementation phase. This phase describes the actual system construction, system testing, security, validation, verification, installation and user training.

CHAPTER 5: IMPLEMENTATION PHASE

5.1 Introduction

The implementation stage involves the coding, installation of software, testing, maintenance and training of users by the developer. The system is tested to discover any errors so that corrections can be made before it can be installed. Testing process is repeated several times until all identified errors have been corrected. The system will run parallel to the existing system for two weeks so that no information is lost should the system fail for any reason. Activities that were undertaken in this chapter include the following:

I. Coding
II. Testing
III. Installation
IV. Maintenance

5.2 Coding

This involved turning program logic into specific instructions that were to be executed by the computer system. PHP was used for the application and MySQL database was used as database backend. All the elements in the data dictionary conceptualised during the database design were mapped into the relevant tables.

The system functionality was developed as modules. The modules were finally integrated into one working system. This stage helped the developer gain extra expertise as it introduced her to new areas of programming that she never had a chance to use elsewhere before. Reviews and walkthroughs were conducted to assure that the project was in track, within budget and within the scope.

Pseudo Code

Registering a client

```
Check connection to the Database
    If connected
        Client enter all the required details
        Check for the required fields
    If correct then
        Check whether the record already exists.
    If exist
        Abandon .update
    Else
        Save the new record
```

Underwriter log in

```
Check for connection to the database
    If connected
        Enter username and password
    If correct then
        Go to underwriter home page
    Else
        Try again
    End if
```

Claims submission

```
Log in to the system
Select proceed
    If premium updated then
        Submit to go next
          View required documents
    If happy with the conditions then
        Submit claim application
    Else
        Wait for premium update
    End if
```

Premium update

```
Login to the system
    Select update premium
If unupdated clients’ premiums available then
    Select nationalid
        Enter client’s premium details
Check for premium validity
    If valid then
        Update client details
    End if
Else
        Premium outside range
        Deactivate account
End if
```

Produce Reports

```
Invoke a database query
    If query is valid then
         Refresh report data
         View report on screen
        Set report to be printable
    Else
    Generate informative message to user            End if
```

5.3 Testing

Software testing consists of the dynamic verification of the behavior of a program on a finite set of test cases, suitably selected from the usually infinite executions domain, against the specified expected behavior (William, 2000). Each test has a different purpose, all work to verify that system elements have been properly integrated and perform allocated functions.

5.3.1 Software testing life cycle

Software testing has its own life cycle that meets every stage of the SDLC. The IAS was tested using the seven step testing life cycle which helped visualise the various software testing life cycle phases. Figure 5.1 illustrates the seven step software diagram that was used.

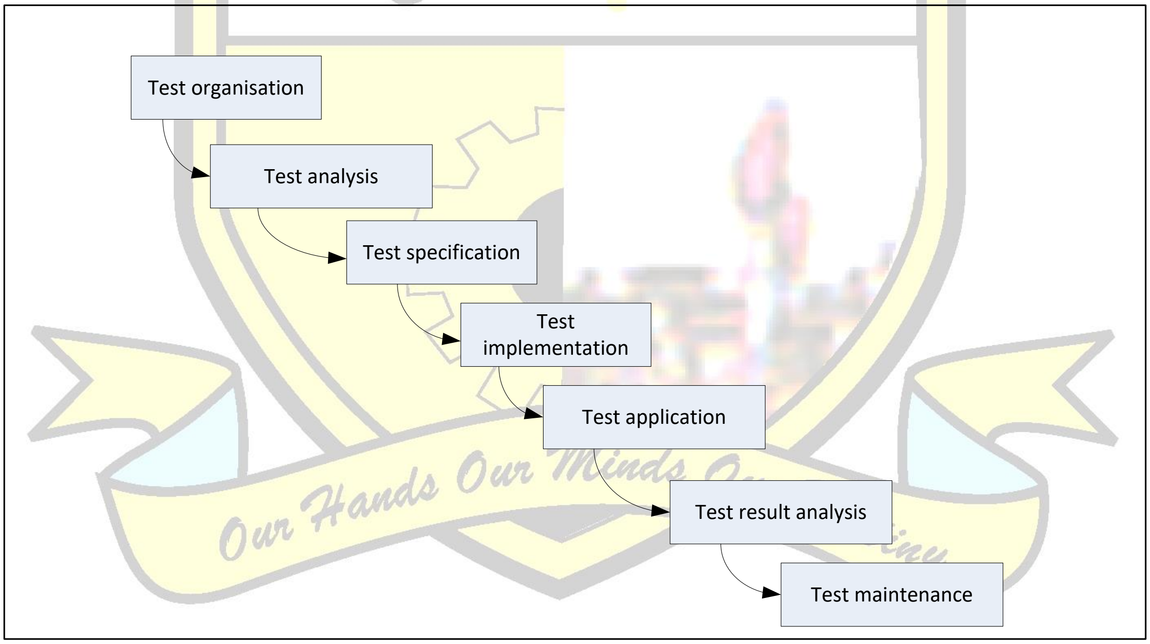


Figure 5.1 Software testing life cycle

Test organisation

Test organisation is the overall activity that observes and controls all other testing activities. It involved developing a test strategy, the break-down into different activities and tasks, planning of all activities and tasks, allocation of time and resources, control, reporting and checking of each activity and task, and keeping track of systems and delivered products.

Test analysis

Test analysis consisted of analysing what should be tested: the system under test and the requirements and specifications against which it should be tested. Moreover, the sorts of testing, the aspects of testing, who will do the testing and the level of abstraction were determined. Test analysis allowed checking whether the requirements and specifications were consistent, complete and non-ambiguous.

Test Specification

During test specification a set of test cases (a test suite) was developed and precisely specified, sequences of test events were composed, inputs were devised and the corresponding expected outputs were documented. The test suite facilitated for the test team to test the system thoroughly, completely and in every aspect, referring to the relevant requirement, specification, or design documents.

Test implementation

Test implementation concerns the preparation of the environment in which the tests will be run. This involved the choice or development of test systems, test probes, devices to stimulate or observe the system under test, implementation of the generated test specifications, etc, in such a way that the specified test suites can be efficiently and effectively run and test results can be observed and logged for later analysis.

Test application

Test application involved systematically running all the test cases in a test suite to the system under test and logging all important events that occur during the test runs.

Test result analysis

When all test events have been carefully registered they can be analysed for compliance with the expected results, so that a verdict about the system's well-functioning can be assigned. The developer and the testing team then analysed and documented the test results.

Test implementation

Test maintenance involves storing and documenting the products of the testing process (the test ware) for later reuse: the test scripts, test environments, used test tools, relations between test sets and versions of specifications and implementations. The aim is to make the testing process repeatable and reusable.

5.3.2 Testing Information flow

The testing team evaluated results from the actual test and also from the expected results and after evaluation of tests, errors were identified and sent for debugging or the strength of the system was identified and the reliability noted. Based on the testing information flow, a testing technique specifies the strategy used in testing to select input test cases and analyse test results. Different techniques reveal different quality aspects of a software system, and there are two major categories of testing techniques, white box and black box which will be discussed later on in the chapter.

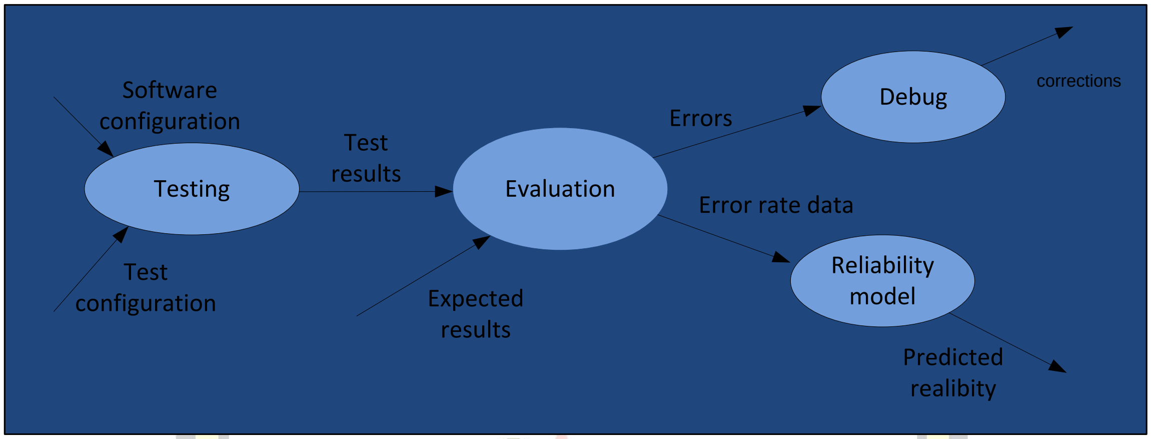


Figure 5.2 System Testing Information Flow

5.3.3 Testing methods

There are two basic classes of software testing, black box testing and white box testing.

White box testing

White box testing (also called structural testing and glass box testing) is testing that takes into account the internal mechanism of a system or component. White box testing allows the developer to have full visibility of the internal workings of the software product, specifically, the logic and the structure of the code (William, 2000). The selection of test cases is based on the implementation of the software entity. The goal of selecting such test cases is to cause the execution of specific spots in the software entity, such as specific statements, program branches or paths. The expected results are evaluated on a set of coverage criteria. Examples of coverage criteria include path coverage, branch coverage, and data-flow coverage. Three types of white box testing were used and these are unit testing, integration testing and system testing.

Unit testing

Unit testing verifies the functioning in isolation of software pieces that are separately testable (Beizer, 1990). Depending on the context, these could be the individual subprograms or a larger component made of tightly related units. Using white box testing techniques, testers (usually the developers creating the code implementation) verify that the code does what it is intended to do at a very low structural level. The objective of unit testing is to identify and eliminate the execution errors and any remaining logical errors. Individual components are independently tested without the other system components. The developer of the Insurance Administration system undertook unit testing by testing the claims module. The developer tested if the claims module gave out an error message if the applicant applies for two insurance claims for the same type of insurance within a short period of time. The claims module gave out an error message and thus the claims module did not have any errors and thus fit to be integrated with other modules.

Integration testing

Integration testing is the process of verifying the interaction between system components (Beizer, 1990). Classical integration testing strategies, such as top-down or bottom-up, are used with traditional, hierarchically structured systems. Integration testing is a continuous activity. Software components, hardware components, or both are combined and tested to evaluate the interaction between them.

In context to the Insurance Administration System, all the components were tested during the unit testing procedure discussed earlier on. The tested unit components were then integrated to evaluate whether the components interacted very well. The developer tested if the insurance policy transactions interacted well with the premiums module. The premium module feeds the clients account with the premium received that month for a specific type of insurance facet. The test was done on all other modules like the claims module and the developer concluded that the system components interacted very well.

System testing

System testing is concerned with the behavior of a whole system. The majority of functional failures should have been already identified during unit and integration testing. System testing should compare the system to the nonfunctional system requirements, such as security, speed, accuracy, and reliability (William, 2000). External interfaces to other applications, utilities, hardware devices, or the operating environment are also evaluated at this level.

The Insurance Administration System was converted into an executable file and the developer tested to see if the Insurance Administration System performed the required functions. The system was able to facilitate for online registration, online insurance policy applications, online insurance claims applications, retrieve records from the database, add records to the database, deliver reports and execute queries and also the proposed system seemed to be vey reliable.

Regression testing

Regression testing is the selective retesting of a system or component to verify that modifications have not caused unintended effects. It is the repetition of tests intended to show that the software's behavior is unchanged except insofar as required (Swanson, 1976). Regression testing was conducted at each of the test levels that were undertaken.

Black box testing

Black box testing (also called functional testing) is testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions (Beizer, 1990). Black box testing attempts to find errors in the external behavior of the code in the following categories: (1) incorrect or missing functionality; (2) interface errors; (3) errors in data structures used by interfaces; (4) behavior or performance errors; and (5) initialization and termination errors.

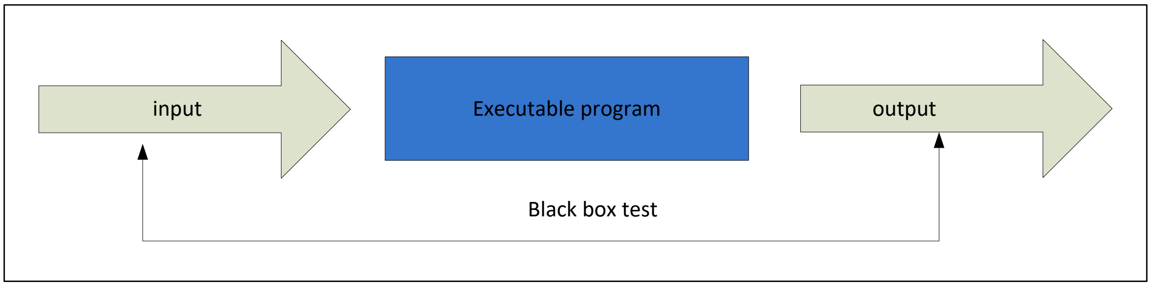


Figure 5.3 Black box Test Diagram

Acceptance testing

Acceptance testing checks the system behavior against the customer's requirements (William, 2000). The customers undertake (or specify) typical tasks to check their requirements. The proposed system was delivered to a customer and the customer ran black box acceptance tests based on their expectations of the functionality. After running the test the client was interviewed and from the results of the interview, it was concluded that the Insurance Administration System was acceptable.

Compatibility Testing

Compatibility testing is done to ensure compatibility of an application or Web site with different browsers, operating systems, and hardware platforms. The Insurance Administration System was tested on Mozilla Firefox, Google Chrome and Internet Explorer and the system worked well with all the three browsers, though the default web browser will be Internet Explorer. The proposed system was also tested on Windows XP and Windows 7 platforms and no errors were detected and thus the system can also work very well on 32-bit or 64-bit platforms.

Alpha and beta testing

Before releasing the system, sometimes it is given in use to a small representative set of potential users, in-house (alpha testing) or external (beta testing), who report potential experienced problems with use of the product. With beta testing, as the users explored the software, any exceptions/defects occurring were reported back and rectified. For the alpha testing, the users (underwriters) were approached during system development and there was noticing of every particular input or action carried out by these users

Usability testing

It evaluates the ease of using and learning the system (and system user documentation) by the end users, as well as the effectiveness of system functioning in supporting user tasks, and finally the ability of recovering from user's errors (Patton, 1994). This testing was done because the user interface of the application stands as an important consideration and needed to be specific for specific types of users, so the point was to come up with a one-size fits all approach.

5.3.4 Testing strategies

These are strategies used in ensuring the correct functionality of programs and the system as a whole. During the testing process logic and syntax errors were identified through the use of code reviews and structured walkthroughs. Verification and validation were used as testing strategies.

Verification

Verification activities include testing and reviews. Through verification, the developer makes sure that the product behaves the way it should be (Sommerville, 2004). The question that was being addressed is whether the product was being constructed in the intended manner that addresses what it is supposed to achieve. A system can be built and will be exceptionally functional but not necessarily addressing the problem at hand thus there was need to carry out systems reviews so as not to deviate from the system development focus. Sample data was used to verify whether the system was accepting the correct inputs and giving the correct results so as to ensure standard quality control.

Validation

Validation is the process of evaluating a system or component during or at the end of the development process to determine whether it satisfies specified requirements (William, 2000). The proposed system was able to warn the user if wrong data is entered in the system i.e. if a client enters text data in a number text box, the system will alert the user to input the right data. The development team vehemently performed specific tests which were performed in some of the stages which include the following.

Regression testing: The version of the software was tested with the automated test harnesses used with Risk manager to ensure that the required features of the Risk manager were still working in the new system.

Recovery testing: The system was deliberately interrupted in a number of ways, including taking its hard disk off line and turning the computer off; to ensure that all the appropriate techniques for restoring any lost data were functioning as per plan.

Security testing: In addition, unauthorised attempts to operate the software, or parts of it, were attempted. This also included attempts to obtain access to the data, or harm the software installation and the system software.

Stress testing: The test involved creating abnormal demands to the system by increasing the rate at which it was asked to accept data, and the rate at which it was asked to produce information.

Performance testing: The performance requirements were also checked. These included the size of the software when installed, the amount of main memory or secondary storage it requires and the demands made of the operating system when running within normal limits or the response time.

Defect testing

The Insurance Administration System also went through the process of defect testing. This testing was useful in identifying faults or defects in the system. A defect test aims to systematically create situations that may cause the system to perform defectively such as data overflows and type mismatches before the system is delivered to the users (Sommerville, 2004).

Test Cases- these are the input and output specifications and a statement of the module under test. The system is tested for a number of quality factors.

Test Data- this refers to the data inputs which were devised to test if the system will produce the expected outputs as specified in the requirements.

Test Results- a test is called successful if it actually reveals an anomaly in the system. A test fails if the system successfully passes the anomaly.

A test of the specified system objectives was carried and their results were displayed to show how the test procedures were conducted. Screenshots and their relative messages were used to display the testing result.

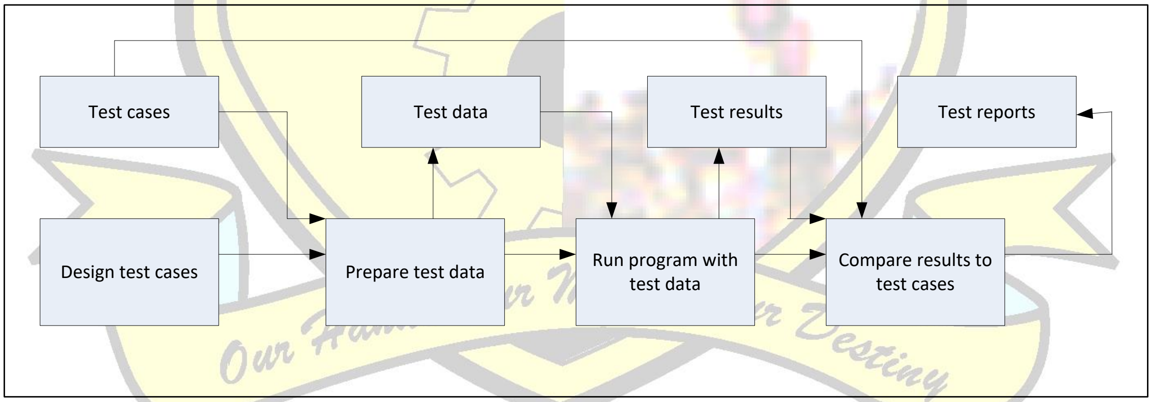


Figure 5.4 Defect testing

Bug tracking

After conducting defect testing, the testers noted all the defects of the system. This is one of the important stages as the Defect Profile Document (DPD) has to be updated for letting the developers know about the defect. A bug life cycle software test was used to monitor the bugs identified. If a bug is genuine, it was sent to the manager and he/she assigned the bug to the developer for correction. The developer resolved the bug and the manager tested the bug again. Figure 5.5 illustrates a diagrammatic representation of a bug life cycle software test.

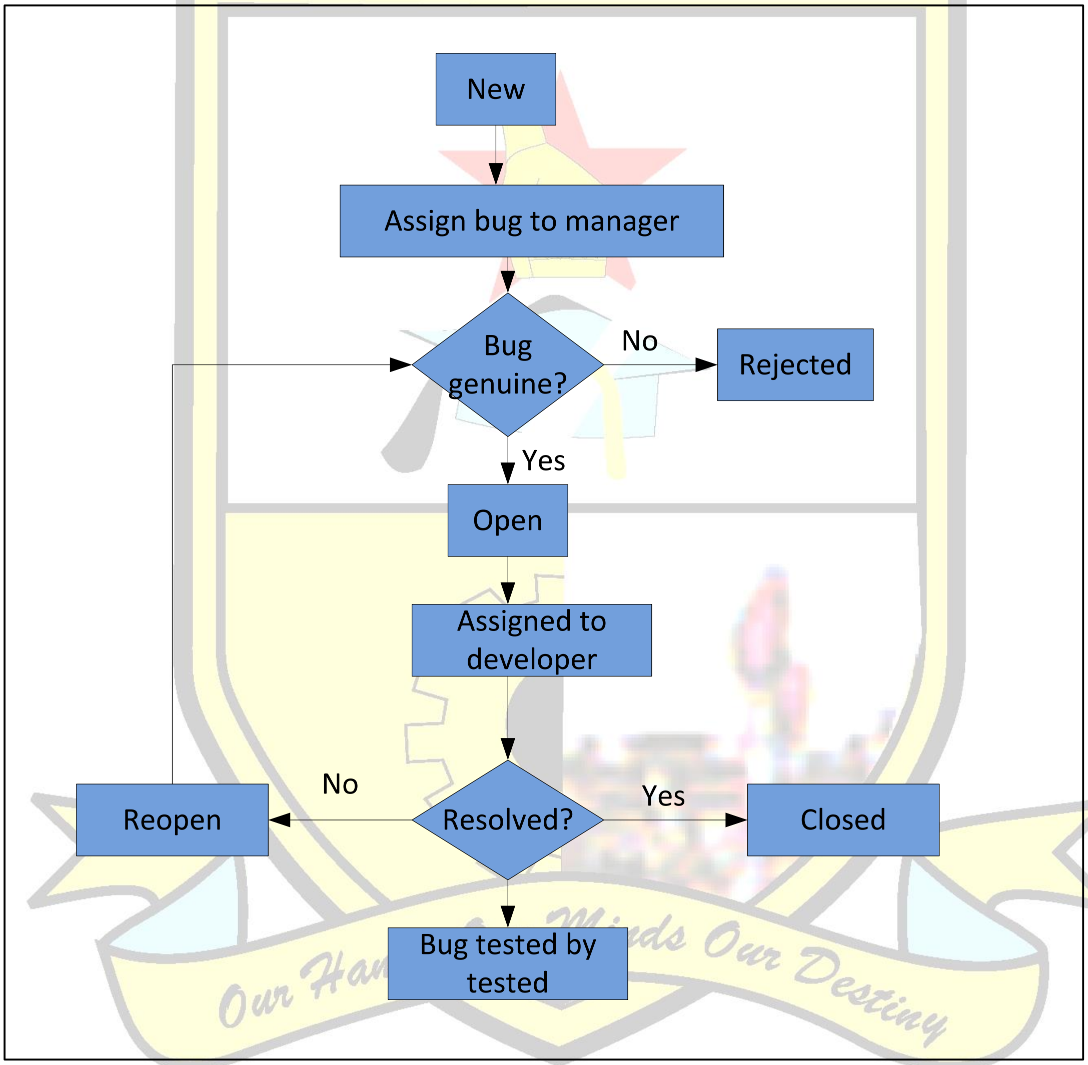


Figure 5.5 Bug Life Cycle Software test

5.3.5 System security

One of the most important things to be considered in this phase was the security of the system and within the system.

Physical Security

Physical security was achieved through adopting the following physical procedures:

Physical Locks: The server room will be strictly locked and will be monitored for twenty-four hours by the company's security department. The server room is fitted with air conditioners to prevent machines from overheating. Also, the Administration block housing the computer will be secured with burglar bars and locks will be put at the doors.

Log in Book: The Server Room will always be locked. If one enters the server room the time is recorded, as well as the time that they leave.

Fire Control Equipment: In cases of fire, smoke detectors have been put in place to send and alarm as soon as enough smoke that could be interpreted as fire is detected. And to fight the fire while still small fire extinguishes have also been put in place, and a proposal; to install overhead water sprinklers.

Software Based Security:

Should a malicious user bypass impede the physical security measures, software based measure were also implemented.

Database Access Passwords: Operating on Microsoft Windows 2010 Server, access to the Data Server is only limited to users with passwords to the Sever. Once logged on, operations that to be undertaken by the users are determined by access levels assigned to the user.

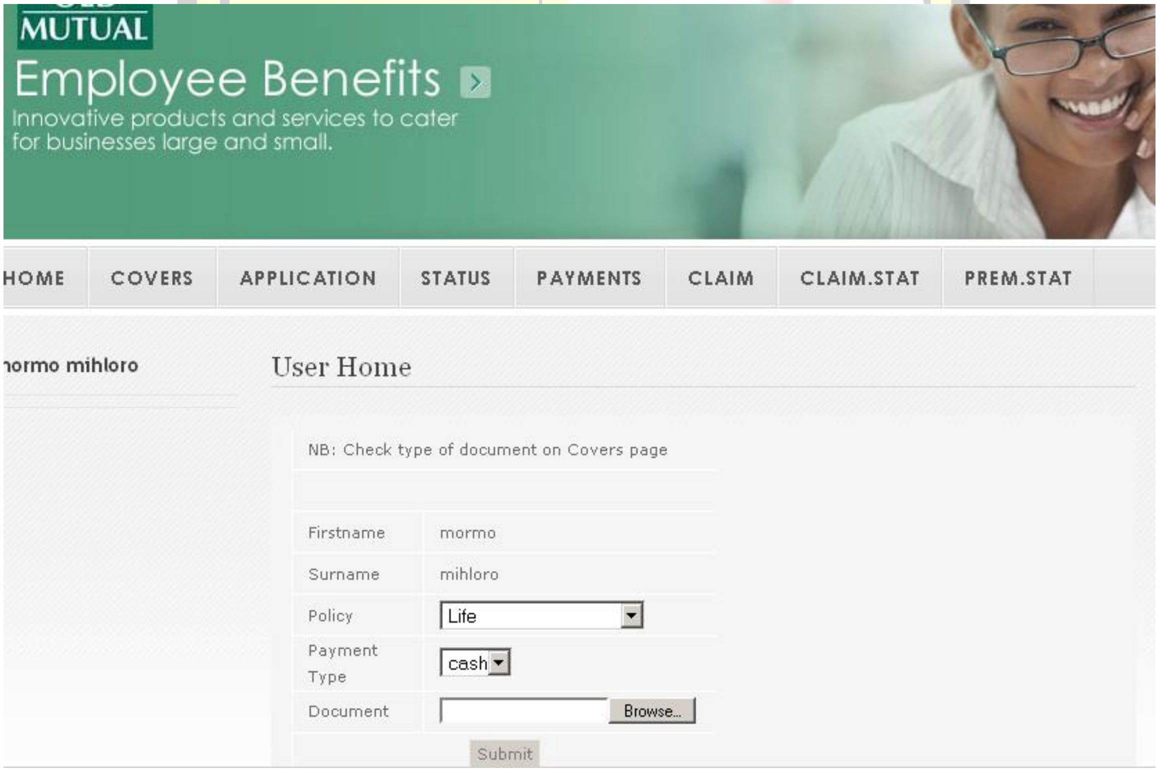
System Passwords and Usernames: All users of the Insurance Administration System are going to be assigned user profiles that assign them usernames and passwords.

Audit Trials: As a means of tracing back operations to user, audit trials are automatically created for any data manipulations operation that occurs. This is achieved through an audit stamp that is attached to each login operation. The audit stamp is a mini-record made up of date logged on and time and date and time logged out of the system. It also includes the name of the user who was logged on. In case of a violation, that activity can be traced back to the user who performed the operation showing when exactly the transaction was made.

5.3.6 Performance and objective evaluation

The IAS was also tested against the entire project objectives so as to ensure that the project was actually able to meet the initial objectives set out in the system requirements specification (SRS) document during the proposal stage. Screenshots and their relative messages were used to display the testing result.

Central repository for client communication, information and document management that will facilitate customer relationship management.



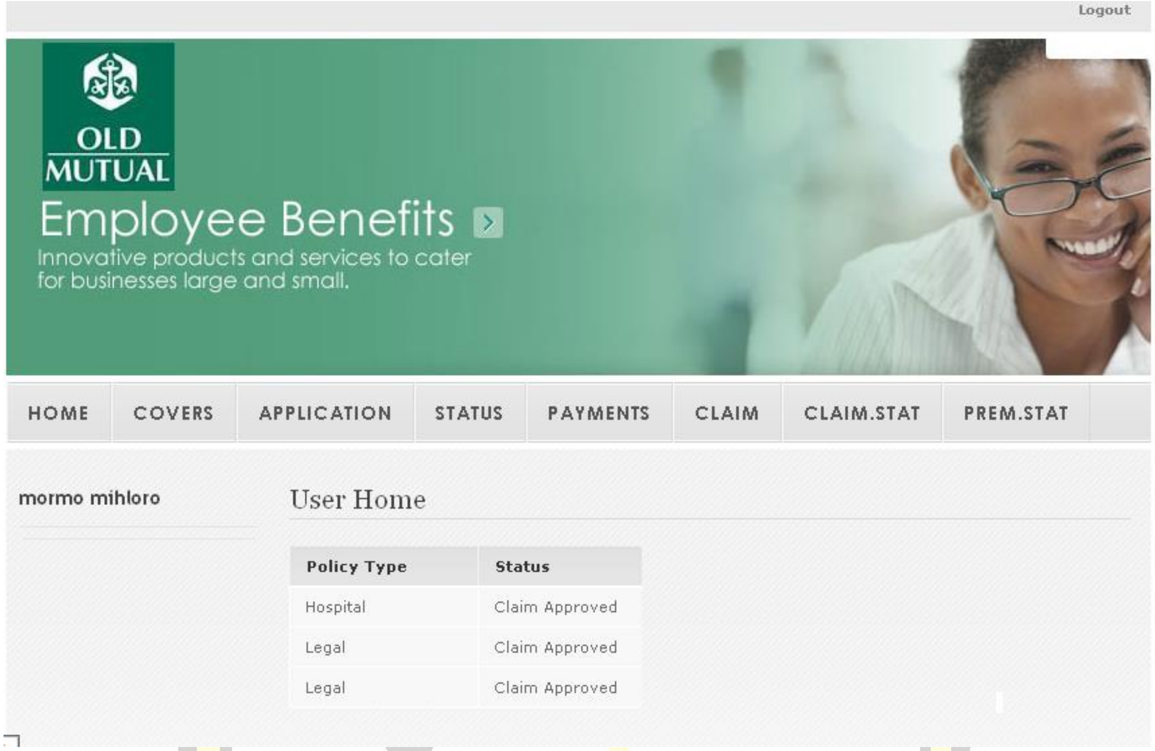
ıormo mihloro

User Home

NB: Check type of document on Covers page

Firstname	mormo
Surname	mihloro
Policy	Life
Payment Type	cash
Document	

Comprehensive process for managing the claims workflow.

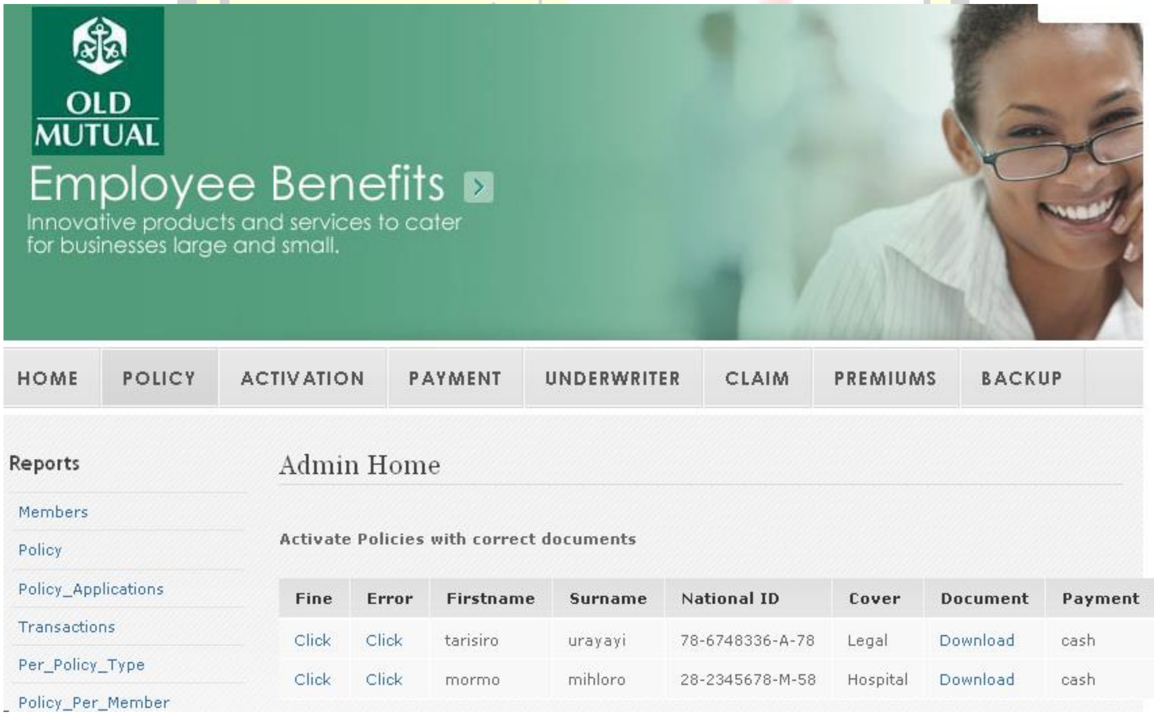


mormo mihloro

User Home

Policy Type	Status
Hospital	Claim Approved
Legal	Claim Approved
Legal	Claim Approved

Automated approval process based on user authority limits.

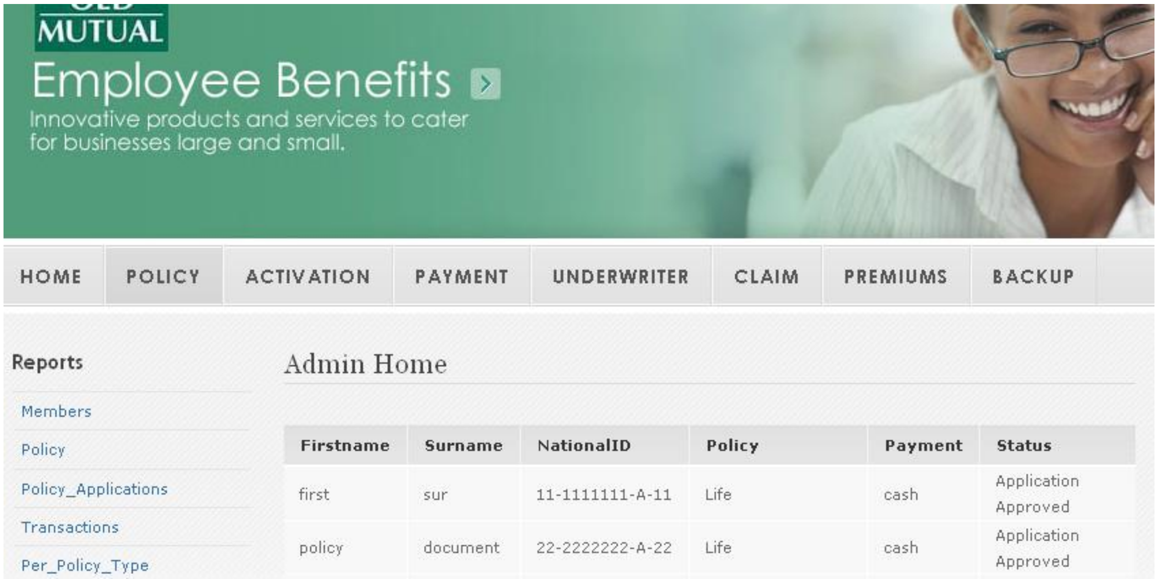


Admin Home

Activate Policies with correct documents

Fine	Error	Firstname	Surname	National ID	Cover	Document	Payment
Click	Click	tarisiro	urayayi	78-6748336-A-78	Legal	Download	cash
Click	Click	mormo	mihloro	28-2345678-M-58	Hospital	Download	cash

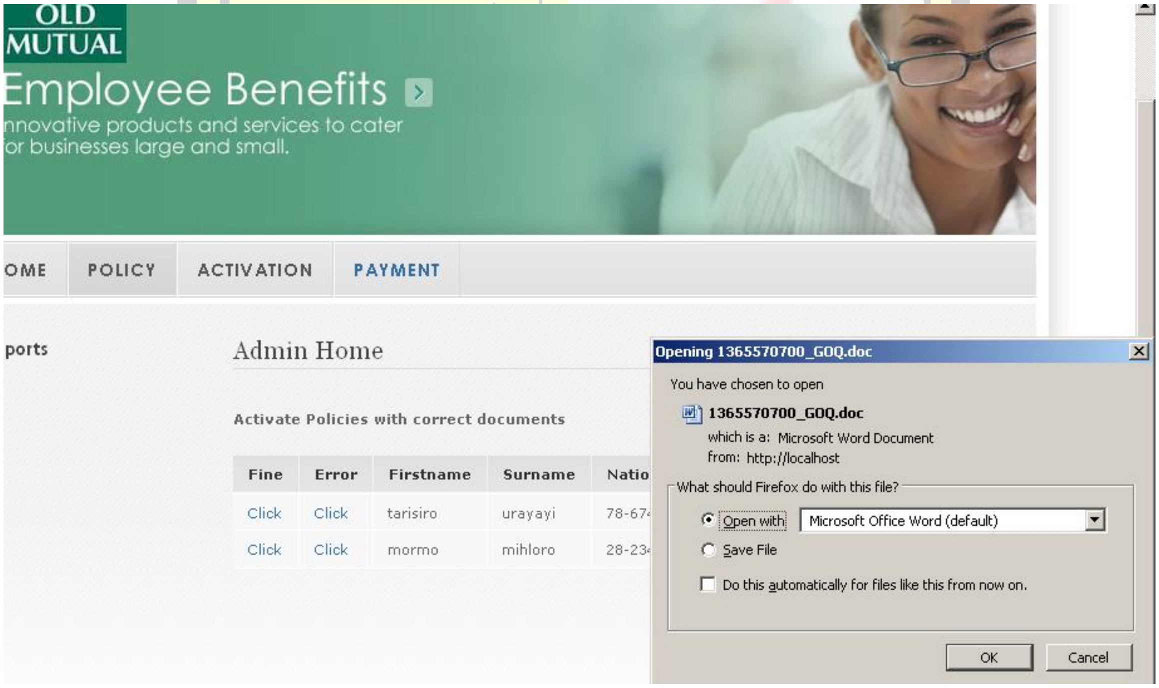
To provide a comprehensive set of standard reports, summarized and detailed, for all facets of insurance with real-time data to enable superior performance analysis and better decision making throughout the organization.



Admin Home

Firstname	Surname	NationalID	Policy	Payment	Status
first	sur	11-1111111-A-11	Life	cash	Application Approved
policy	document	22-2222222-A-22	Life	cash	Application Approved

The system will provide automatic generation of claims documents.


OLD MUTUAL
Employee Benefits
nnovative products and services to cater
or businesses large and small.
OME
POLICY
ACTIVATION
PAYMENT
ports

Admin Home

Activate Policies with correct documents

Fine	Error	Firstname	Surname	Natio
Click	Click	tarisiro	urayayi	78-67
Click	Click	mormo	mihloro	28-23

Opening 1365570700_GOQ.doc

You have chosen to open

1365570700_GOQ.doc

which is a: Microsoft Word Document

from: http://localhost

What should Firefox do with this file?

● Open with Microsoft Office Word (default)

○ Save File

□ Do this automatically for files like this from now on.

5.4 Installation

The system was installed within the Old Mutual's network as defined within the design documentation. In this phase the required software was installed on the appropriate hardware converting from the as-is system to the proposed system. A number of activities such as training, file conversion, system changeover methodologies such as direct changeover, parallel running and pilot operation were carried out during this stage. Installation included software and hardware configurations. Installation is rendered incomplete if it is not complemented with sufficient user training.

5.4.1 The process of installation

The software was installed from the CD ROM and a copy for backup was left with organisation and the other with the development team.

Steps for Application software installation

- The system is installed from the software CD
- Install system to path. (It is recommended that the system be installed to the program files folder.)
- Verify that the system is installed properly that is all folders are installed.
- Add the system data source which is "IAS" in this case
- Connect data source to the database on the main server housing the system database.

5.4.2 User training

User training defines teaching to the users on the operation of the developed system. The identified user groups were trained on how to use the system according to the areas of need. Training was conducted using seminar manuals, online training, peer support and on the job training.

IT Administrators- The training for this group mainly included how to perform system backup and how to troubleshoot and rectify some errors that may come up in the system. It was also necessary to train them on how to generate reports.

Underwriters- This is a special group that was trained on how to update clients' premiums, how to analyse client application for insurance policy and the insurance claims.

Fund manager- this group was trained on how to search for clients, view claims submitted and how to approve or disapprove claims applications.

Training Plan

The training of all user groups was in-house and was divided into three sessions. The IT support team attended all the three sessions because they need to have an overview of the whole system since they will be responsible for troubleshooting and helping the users with challenges that may be faced during the system operation as well as maintenance and further development.

Session 1:

Table 5.1: Administrators training

Venue	Old Mutual Training Room
User Group	IT administrators
Training Scope	Systems functionality overview, backup operation and Report generation
Requirements	Three computers and an overhead projector
Trainer	System Developer (Miss Mornisher Mihloro)

Session 2:

Table 5.2: Underwriters' training

Venue	Old Mutual Training Room
User Group	IT administrators , Underwriters
Training Scope	System Trouble shooting, Handling errors, Documenting changes, updating insurance premiums, analysis of claims documents, analysis of insurance policies
Requirements	Six computers and, an overhead projector.
Trainer	System Developer (Miss Mornisher Mihloro)

Session 3:

Table 5.3: Fund Managers' training

Venue	Old Mutual Training Room
User Group	IT administrators, Fund Manager
Training Scope	Claims analysis, general system functionality
Requirements	Four computers and an overhead projector
Trainers	System Developer (Miss Mornisher Mihloro)

5.4.3 File conversion and system changeover

Conversion involves changing from the old system to the new system. This is a technical process whereby the new system replaces the old system and will take effect after the establishment of the operational environment and training has been established (Mallach, 2009). The aim here is to gain user acceptance of the information system with minimum disruption.

Several methods were evaluated to come up with the best strategy to implement the Insurance Administration System and these changeover strategies are:

I. Pilot Conversion

II. Direct Conversion
III. Phased Conversion
IV. Parallel Conversion

Pilot Conversion

One department uses the new system and all others continue with the old system until a decision is made to put the new system into place across the entire system of the organisation. Pilot conversion is a combination of the direct conversion and parallel conversion (William, 2000). Pilot conversion can be applied to Old Mutual by having one type of insurance facet use the new proposed system. The short term insurance underwriters can use the new proposed system first while the legal, life funeral insurance underwriters continue using the old system.

Advantages

I. If a new system fails in pilot conversion, the old system continues with a minimal loss of data, thus risk is minimal
II. Pilot conversion also allows for the new system to be tested with real-world data as a complete system, and allows participants in the system to train others in its use.

Evaluation

In the context of the Old Mutual scenario, the distinction between a phased conversion and a pilot conversion is blurred in this situation. Pilot conversion would not apply in this current situation**.** If a system undergoing pilot conversion does fail, the data in the new system is lost. It can also lead to some confusion as to which system is being used for what, thus pilot conversion was not chosen for the implementation of the Insurance Administration System.

Direct Conversion

Direct conversion method stresses upon the introduction of a completely new system without any reference to the existing system. The old system is abandoned and the new system becomes completely operational. Direct conversion is an implementation process that involves essentially “switching off” the current system entirely and “switching on” the new system to take its place all at once (Mallach, 2009). The old system is no longer available for use from that point onward.

In the context of Old Mutual insurance administration policy, this would mean shelving the current insurance policy management system entirely and have all users and participants use the proposed Insurance Administration System (IAS) the next business day.

Advantages

I. Direct conversion can be achieved very quickly, thus reducing the downtime of the system.

II. Reduced pressure on participants as only one system will be in place

III. It is also the easiest method when compatible hardware and software exists in both systems.

Evaluation

Although direct conversion has merits that are quite favourable, Old Mutual did not employ this method of conversion, because current and the proposed systems are vastly distinct from each other. The greatest risk is the learning curve required by users and participants to interact effectively with the new system. Other major risks include the new system not performing as per the requirement report specifications, which will impact the business in a profoundly negative manner as the old system will no longer be available as a redundant failover. Participant stress is high due to training needing to be done while the old system is in use. Therefore, direct conversion was not chosen.

Phased Conversion

Phased conversion involves a gradual introduction of the new system, while at the same time replacing elements of the current system until the current system is completely replaced by the new system (Mallach, 2009). For Old Mutual to implement the phased conversion it will have to install the proposed system to one function of the manual based system. For example developer may install the proposed system to the insurance policy application process and this means that only the insurance officers will be using the proposed system. Then the other processes like claim processing and premium update processing will be left using the manual system. After the application process is successfully tested then the developer installs the proposed system to the next process and so on.

Advantages

I. Phased conversion allows for each module of a new system to be individually tested and evaluated with real-world data.

II. It also allows for staff to gradually acclimatise to the new system.

III. If a single module fails, that module can be removed and corrected, without great loss to the remainder of the new system.

Evaluation

This conversion method was arguably the least risk with respect to implementing the Old Mutual Insurance Administration System. Old Mutual did employ phased conversion because the greatest disadvantage is the time necessary for complete conversion to occur, as element by element of the current system is replaced by its electronic counterpart. Alongside this gradual replacement, training must be ongoing to ensure that users and participants are well versed with the new system during its gradual implementation. Phased conversion is also relatively expensive as the system is implemented in stages as opposed to all at once (such as in direct conversion). Phased conversion also confuses participants as they have to separate the old system and the new system especially in having to perform different tasks in each system, thus, the phased conversion approach becomes very complex. Some modules may be incompatible with the old system existing before other modules are introduced.

Parallel Conversion (Recommended)

Parallel conversion involves running both the current and the new system together for some period of time. At some pre-determined time, the current system is decommissioned entirely and all users and participants interact solely with the new system. Once there is confidence that the new one operates properly, the old one is shut down.

For Old Mutual this meant running the proposed system together with the old system. Information look-ups can be performed electronically on the proviso that the data entry is performed both electronically on the new system and via the current paper based system. This means that online transactions were also done manually.

Disadvantages

I. Participants stress high as both systems are in place and have to be operated by the same people at the same time.

II. Confusion may arise between systems if close records are not kept

III. Problems may arise from the fact that there may be big changes in the operating procedures of both systems

Justification for parallel conversion

For Old Mutual Insurance system parallel conversion is ideal. Both the new and the old systems are fully operational for a period of time, allowing comparison of the two. This allowed for the new system to be tested with a real-world set of data, which can be compared to the old system. Also, if the new system fails, the old can continue with a minimum loss of data, as both systems are kept up-to-date. There exists some element of redundancy should the new system fall short of user/participant expectation or there is some catastrophic system failure with the new solution. The switch to the new system will occur at a time when everyone is confident that the new system will perform the tasks it was originally intended for. The old system and the proposed systems can be compared and any problems in the new proposed system can be fixed to account of the differences. Testing and fixing of problems in the new proposed system is simpler as the old system is still available for use in an emergency.

Overview of system changeover

The developer analysed the four different types of system conversion. Each one of the methods had its own merits and demerits. The developer choose to use the parallel conversion since it is the only conversion method that accounts for both systems being in use at the same time and also making it possible to compare and detect any problems with the proposed system.

5.5 Maintenance

Maintenance is the modification of a software product after delivery to correct faults, improve performance, or other product attributes, or to adapt the product to a new or changing environment (Sullivan, 2002). This section area provides resources relevant to all aspects of software maintenance. Maintenance focuses on four major aspects of system evolution simultaneously which are:

I. Maintaining control the day-to-day functions
II. Maintaining control over system modification
III. Perfecting existing acceptable functions
IV. Preventing system performance from degrading to unacceptable level

5.5.1 Corrective maintenance

Corrective Maintenance is reactive modification of a software product performed after delivery to correct discovered problems (Patton, 1994). Once an error has been identified and reported, investigations will be launched to establish the root cause of such errors. The maintenance team responds to problems resulting from faults. As failures occur, they are brought to the team's attention; the team then finds the failure’s cause and makes corrections and changes to requirements, design, code, test suites and documentation, as necessary. Often, the initial repair is temporary; something to keep the system running, but not the best fix. Long range changes may be implemented later to correct more general problems with design or code. For example the IAS might develop faults in the database such that a client may end up viewing statements that are not linked to his/her transactions. The system might start to give false reports and all these errors need to be adhered to by the developer.

Advantages

I. Low cost

II. Less staff.

Disadvantages

I. Increased cost due to unplanned downtime of equipment

II. Increased labor cost, especially if overtime is needed

III. Cost involved with repair or replacement of equipment

IV. Possible secondary equipment or process damage from equipment failure

V. Inefficient use of staff resources.

5.5.2 Adaptive maintenance

Sometimes a change introduced in one part of the system requires changes to other parts. Adaptive maintenance is the implementation of these secondary changes (Sullivan, 2002).

For example if a compiler is enhanced by the addition of debugger, the developer must alter the menus icons, or function key definitions to allow users to choose the debugger option. For example, the Insurance Administration System will be changed from the current platform to a virtual machine due to the virtualisation technology that give most companies cost saving advantage as far as hardware purchasing is concerned. The system might need to be changed as well to meet changing user requirements as it exists in a constantly changing environment.

5.5.3 Perfective maintenance

As the maintenance team maintains a system, they examine documents, design, code and tests, looking for opportunities for improvement. Perfective maintenance involves making changes to improve some aspect of the system, even when the changes are not suggested by faults (Sullivan, 2002). Documentation changes to clarify items, test suite changes to improve test coverage, and code and design modifications to enhance readability are analysed during perfective maintenance. Implementation of new functional or non-functional system requirements generated by users as their organisation or business changes is also catered for in this section. For example if the Old Mutuals' clients want to make more insurance claims within a short period of time, the IAS should be able to allow such modifications without failing other functionalities of the system. This process ensures that the newly implemented system meets the system development objectives established

5.5.4 Preventive maintenance

Similar to perfective maintenance, preventive maintenance involves changing some aspect of the system to prevent failures. It may include the addition of type checking, the enhancement of fault handling, or the addition of a "catch-all" statement to a case statement, to make sure the system can handle all possibilities (Higgins, 1994). Preventive maintenance usually results when a programmer or code analyser finds an actual or potential fault that has not yet become a failure and takes action to correct the fault before damage is done. Actions performed on a time or machine-run-based schedule that detect, preclude, or mitigate degradation of a component or system with the aim of sustaining or extending its useful life through controlling degradation to an acceptable level.

Advantages

I. Flexibility allows for the adjustment of maintenance periodicity

II. Increased component life cycle

III. Energy savings

IV. Reduced equipment or process failure.

Disadvantages

I. Catastrophic failures still likely to occur

II. Labor intensive

III. Includes performance of unneeded maintenance

5.5.5 Overview of system maintenance

Having evaluated all the different types of system maintenance methods, the developer concluded that it was vital for the Insurance Administration System to be maintained by all the above mentioned methods. The responsible people will perform corrective maintenance to make sure that all the errors encountered by the system are dealt with before they ruin the whole system, preventive maintenance to make sure that errors are dealt with before they surface up or become worse off, adaptive maintenance that will facilitate the Insurance Administration System to be usable even if hardware has to be upgraded, the system will adapt to the new architectural design and also perfective maintenance that will allow for continuous assessment and improvement for the proposed system.

The implementation of the four maintenance strategies might be expensive but it is viable due to the cost that was allocated to maintenance in the cost benefit analysis in chapter 2 earlier on.

5.5.6 Disaster recovery

The IAS will be complemented with a congruent disaster recovery plan that will ensure the recovery of data (prevent data loss) in case of hardware or software failure. Backup of all the system files will be done daily at the end of the day so as to safeguard against uncertainties of data needs, and the media to be used are data tapes and external drives. IAS data files will be kept at the current offsite disaster recovery site which is the CABS head office in Borrowdale.

The system application must have its own independent backup for when there is need to reinstall the system. The system can thus be reinstalled in cases that the system is not functioning due to any reason that may cause system function not to be as expected. The system will produce reports and these will be used in the reconciliation of the company and queries from clients respectively for better planning and control of the manager.

5.5.7 Recommendations

The project was completed successfully and some recommendations were made to the various stakeholders who were involved in the development of this project. Such recommendations included:

I. For the purpose of error minimisation, users should be trained properly in the processing of data.

II. There is need to adhere to the security procedures as specified in the user manual to ensure data confidentiality and integrity.

III. The maintenance and back-up plan should be seriously taken into action as these play a major role in case of hardware or software failure thus followed as per the designed plan.

5.6 Conclusion

The project has been a success in relation to the fulfilment of the desired objectives. The project has also been well planned and budgeted for as referenced by the few problems encountered during the project development. Although it has been overwhelmingly passed as a success the developing team also looked ahead and expect to continuously work on improving the system.

REFERENCES

Beizer, B. (1990), Software Testing Techniques, 2nd Edition, Van Nostrand Reinhold, UK

Borman, K. (2011), Financial Intelligence, Revised Edition, Library of Congress, USA

Dehning, B. (2002), Returns on Investment in Information Technology: A research synthesis. Journal of Information Systems, USA

Demarco, R. (1979), Structured Analysis and system specification, Prentice Hall, USA

Denis, A. Wixom, B. (1999), System Analysis and Design, 1st Edition, Prentice Hall, London.

Elmasri, N. (2011), Fundamentals of Database Systems, 6th Edition, Pearson Education Publishing, USA

Shelly, G. Thomas J. Cashman, Harry J. Rosenblatt, (2009), Systems Analysis and Design, Cengage Learning, London

Higgins, R. Brautigam, P. (1994), Maintenance Engineering Handbook, 5th Edition, McGraw Hill Text, USA

Joseph, S. Valacich, Joey F. George, Jeffrey A. Hoffer. (2003), Essentials of Systems Analysis and Design, Prentice Hall, Canada

Karolak, D. (1996), Software Engineering risk management, Los Alamitos, IEEE Computer Society Press, USA

Keil, M. (1998), A framework for identifying software project risks, Communications of the ACM, Canada

Kendall, K. Kenneth, E. (1995), System Analysis and Design, Englewood Cliffs, Prentice Hall, London

Kerzner, H. (2011), Project Management: A Systems Approach to Planning, Scheduling and Controlling, John Wiley and sons, Canada

King, D. (2000), Project Management Made Simple: A Guide to Successful Management of Computer Systems Projects, Yourdon

Layard, R. (1999), Cost benefit analysis, 2nd Edition, Press Syndicate of the University of Cambridge, Australia

Lewis, P. (2000), The Project Manager's Desk Reference: A Comprehensive Guide to Project Planning, Scheduling, Evaluation and Control Systems, McGraw Hill, USA

Lucey, T. (1997), Management Information Systems, 8th Edition, Publication, New Delhi

Mallach, E. (2009), Information Conversion Strategies, IGI Publishing, USA

Mark, C, Richard, T. (2001), The Project Manager's Guide to Software Engineering's Best Practices, IEEE Computer Society, England

Overton, R. (2007), Feasibility Studies, Martin Books Pty Ltd, Australia

Patton, J. (1994), Maintainability and Maintenance Management, 3rd Edition, Instrument Society of America, USA

Pomberger, G. (1986), Software Engineering and Modular 2, 1st Edition, Prentice Hall, London.

Dowell, G. (2006), Beginning Database Design, John Wiley and sons Publishing, USA

Purba, S. (1995), How to Manage a Successful Software Project: Methodologies, Techniques, Tools, John Wiley, New York

Ryan, K. (2001), Database Design, Sams Publishing, USA

Shelly, G. Harry, R. (2009), System Analysis and Design, 5th Edition, Canada

Sommerville, I. (2004), Software Engineering, 7th Edition, John Wiley & sons, USA

Stickney, C. (2000), Financial Accounting: An introduction to concepts, methods and uses, Harcourt, New York

Sullivan, G. (2002). Operations and Maintenance, Best Practices, A Guide to Achieving Operational efficiency, UK

Swanson, E. (1976), The dimensions of maintenance Proceedings of the 2nd international conference on Software engineering, San Francisco

William, P. (2000), Effective Methods for Software Testing, 2nd Edition, John Wiley & Sons, USA

Whitten, J. Bentley, L. (2007), Systems analysis & design methods, 7th edition, McGraw-Hill/Irwin, USA http://searchsoftwarequality.techtarget.com/answer/Functional-requirements-and-nonfunctional-requirements-in-software-engineering-explained **Accessed 8 March 2013**

http://www.springer.com/computer/image+processing/journal/10032vv **Accessed 11 March 2013**

http://www.drcath.net/toolkit/document.html **Accessed 13 March 2013**

http://ix.cs.uoregon.edu/~michal/book/slides/pdf/Pezze-Young-CH22-System.pdf **Accessed 20 March 2013**

aches/assets/MethodsOfImplementation_AdvDisAdv.pdf **Accessed 30 March 2013**

http://www.cs.bham.ac.uk/~exc//Teaching/sTeaching/Lectures/14.System testing.pdf **Accessed 10 April 2012**

http://www.seminarprojects.com/search.php **(Accessed on 27/02/13)**

http://pewinternet/org/pdfs/PIP_Jobhunt_Memo.pdf **(Accessed on 01/03/13)**

www.uasc.edu/learning_material/systems and design/2003.html **Accessed 01/02/2013**

http://www.seminarprojects.com/search.php **Accessed 31/01/2013**

www.oldmutual.co.zw **Accessed 31/01/2013**

APPENDICES

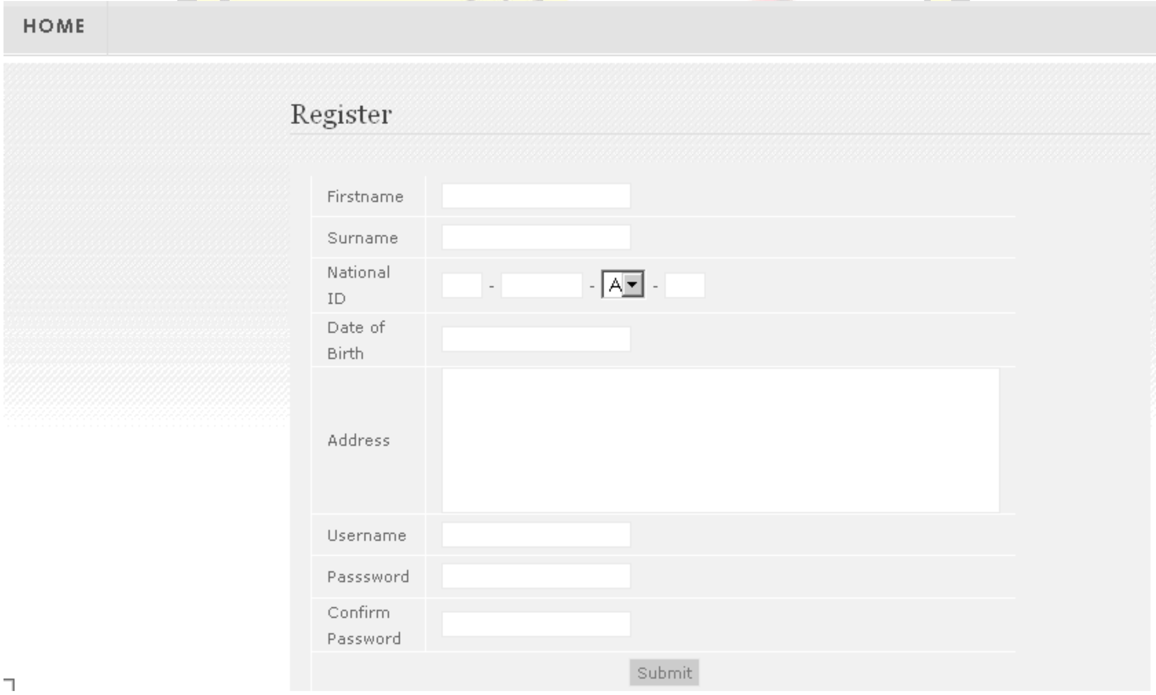
APPENDIX A: User Manual

Introduction

Below is the user-manual with detailed instructions for the use of the Old Mutual Insurance Administration System. This document is going to be used for user training and it is also going to be used as the reference in the future for further training or whenever the system needs to be maintained. The instructions are standard and the target audience is mainly the clients, underwriters, the fund manager and the system administrator.

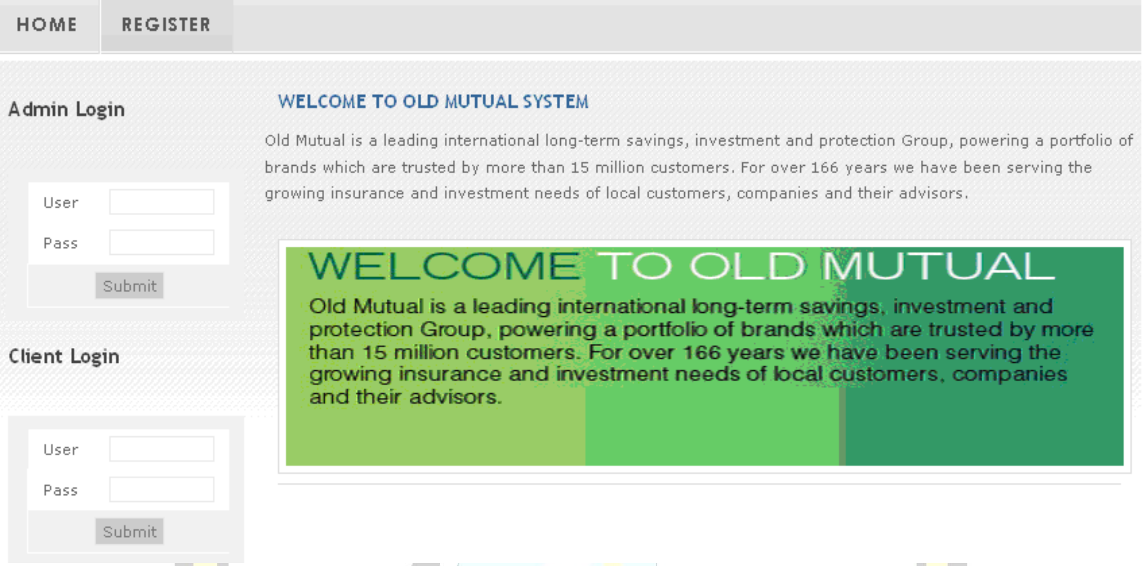
The first step is for a client to register for an online insurance account before any logins to the system. Then other tasks will be executed as highlighted only after a client has registered.

Client registration form- this form shows how the client registers for the creation of his or her online insurance account.



Log in Form

This form shows how to login as a client or administrator. One has to select where appropriate. Only registered individuals with valid credentials are able to login



WELCOME TO OLD MUTUAL SYSTEM

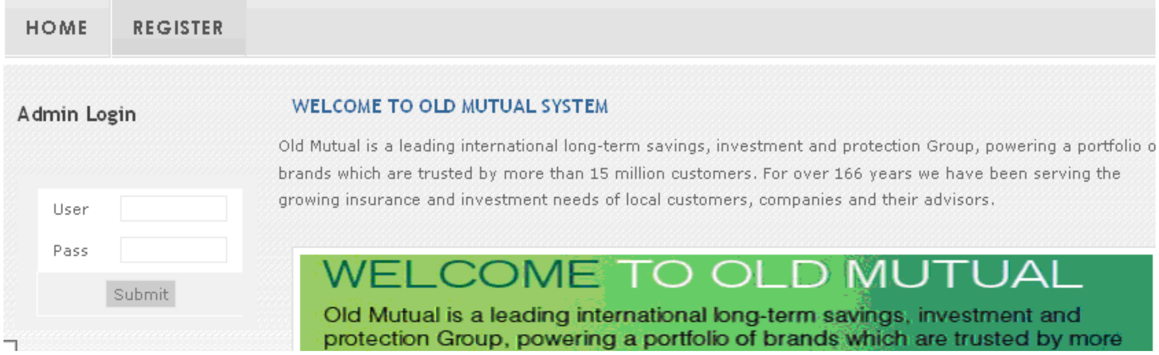
Old Mutual is a leading international long-term savings, investment and protection Group, powering a portfolio of brands which are trusted by more than 15 million customers. For over 166 years we have been serving the growing insurance and investment needs of local customers, companies and their advisors.

WELCOME TO OLD MUTUAL

Old Mutual is a leading international long-term savings, investment and protection Group, powering a portfolio of brands which are trusted by more than 15 million customers. For over 166 years we have been serving the growing insurance and investment needs of local customers, companies and their advisors.

Admin Log In form

This is the administrator's gateway to the system where he would access his or her account to carry out administrative tasks.



WELCOME TO OLD MUTUAL SYSTEM

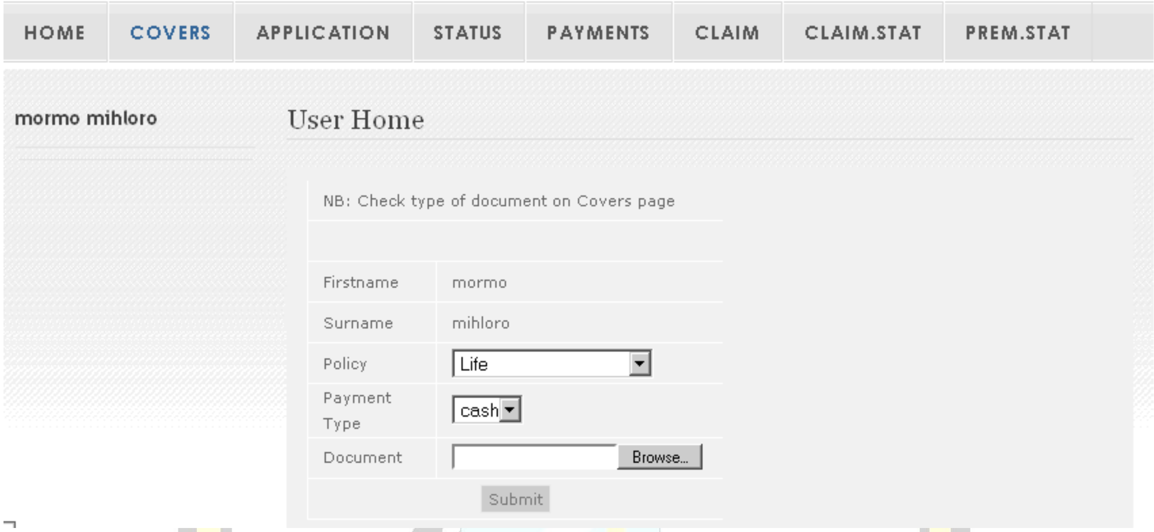
Old Mutual is a leading international long-term savings, investment and protection Group, powering a portfolio o brands which are trusted by more than 15 million customers. For over 166 years we have been serving the growing insurance and investment needs of local customers, companies and their advisors.

WELCOME TO OLD MUTUAL

Old Mutual is a leading international long-term savings, investment and protection Group, powering a portfolio of brands which are trusted by more

Applicant has logged in

After the applicant has successfully logged in he/she is provided with access to the functionalities of the Old Mutual Insurance Administration System. If the client does not have an insurance policy she/she can apply for the insurance policy.

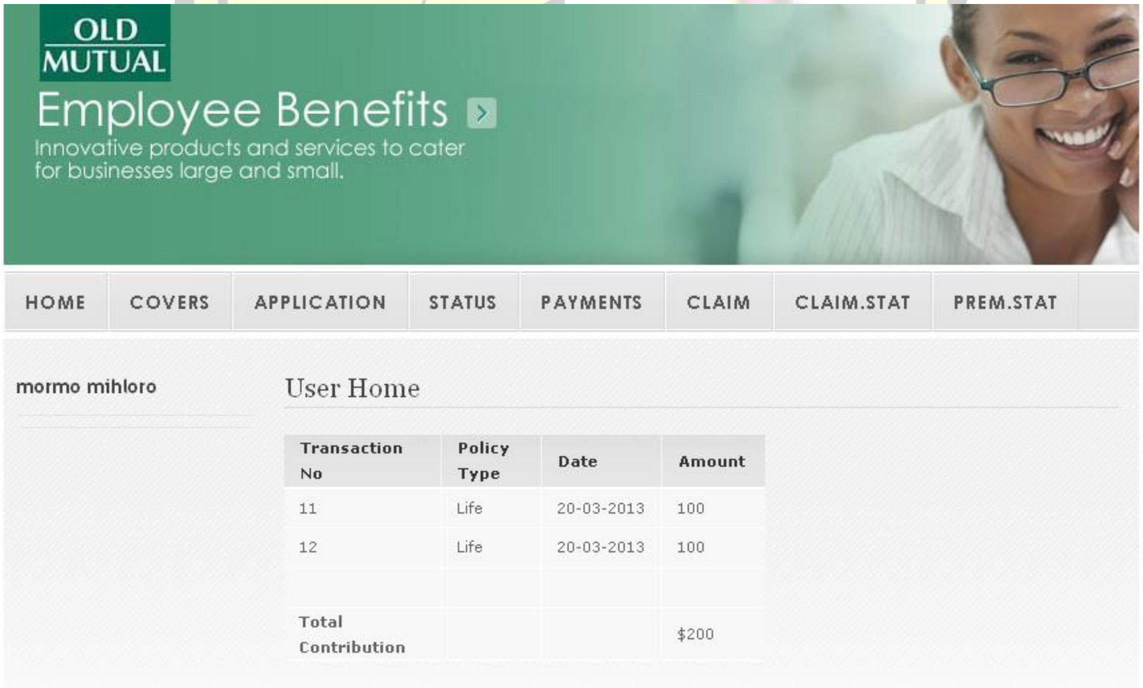


User Home

NB: Check type of document on Covers page

Firstname	mormo
Surname	mihloro
Policy	Life
Payment Type	cash
Document	

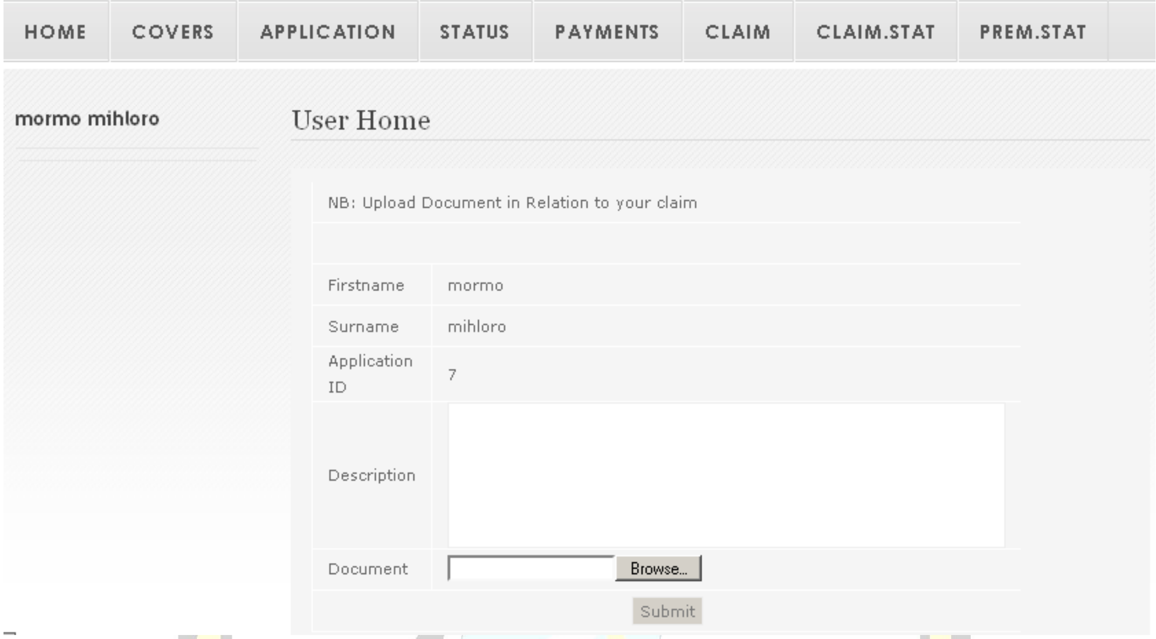
The client can also check if the premium that she or he paid for that month was updated in his or her account.



User Home

Transaction No	Policy Type	Date	Amount
11	Life	20-03-2013	100
12	Life	20-03-2013	100
Total Contribution			$200

To lodge a claim the client has to log in their accounts and provide the necessary details for that particular claim.



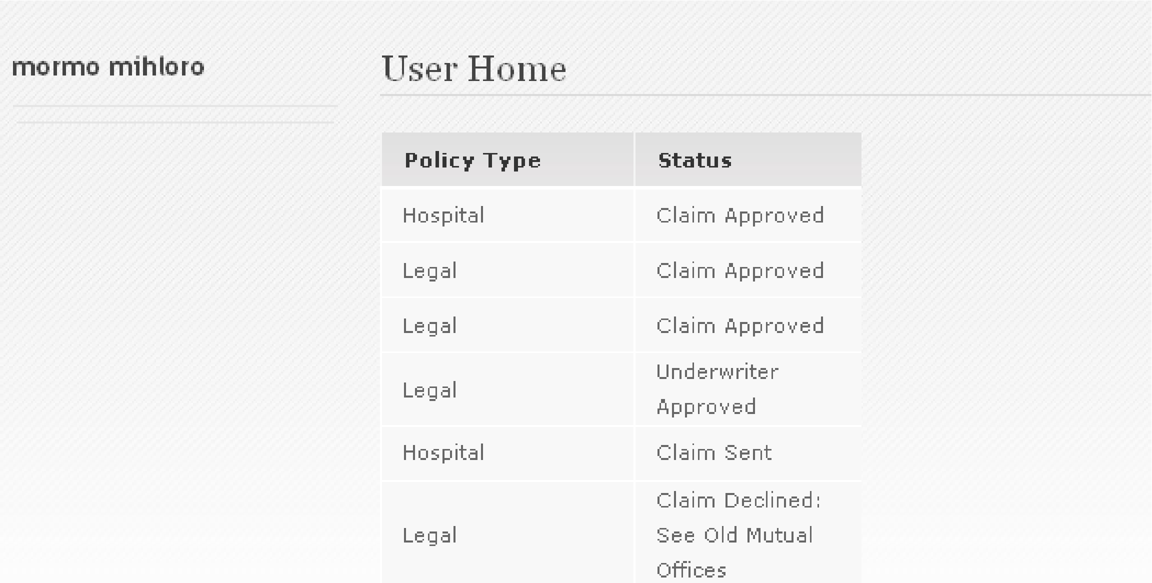
mormo mihloro

User Home

NB: Upload Document in Relation to your claim

Firstname	mormo
Surname	mihloro
Application ID	7
Description	
Document	

The client can also check if his or her claim has been approved or declined. To do this the client clicks on the claims status module.

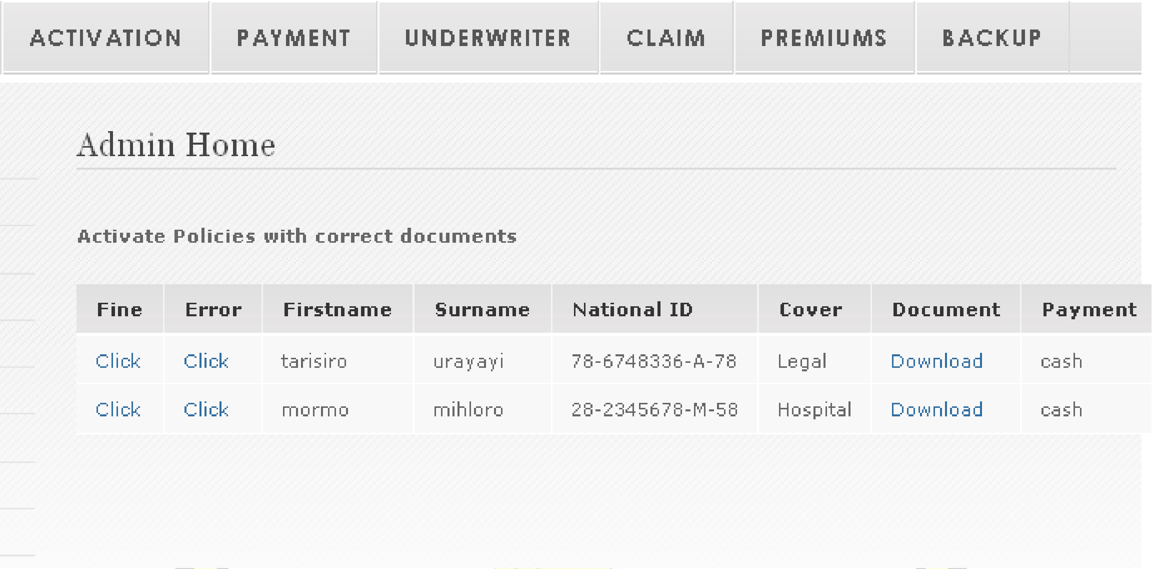


mormo mihloro

User Home

Policy Type	Status
Hospital	Claim Approved
Legal	Claim Approved
Legal	Claim Approved
Legal	Underwriter Approved
Hospital	Claim Sent
Legal	Claim Declined: See Old Mutual Offices

To activate insurance policy applications, the underwriter click on the activate module and approve or decline the application.

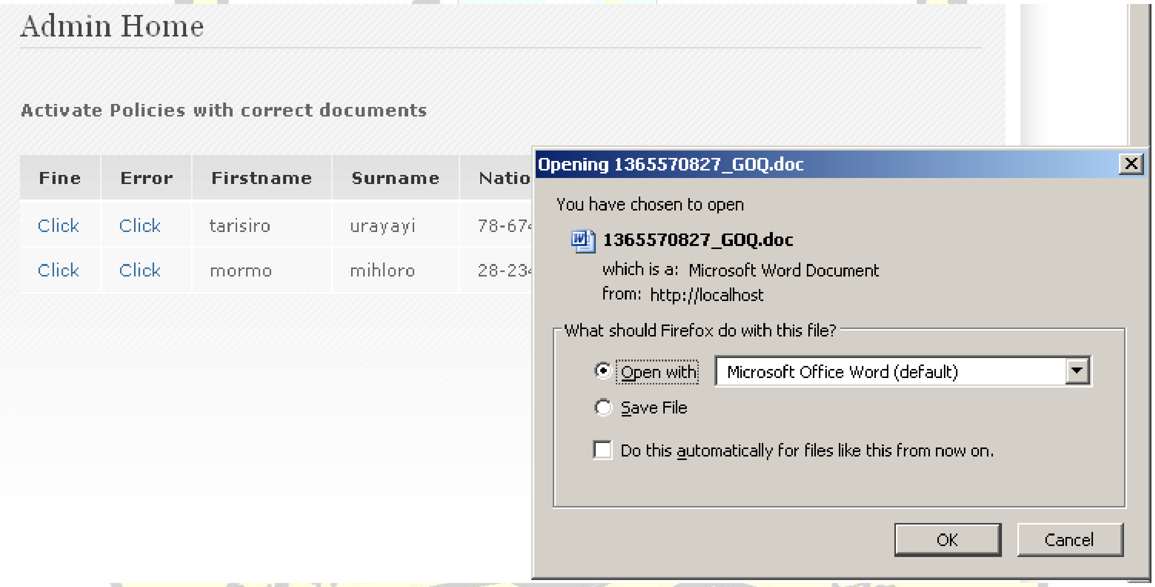


Admin Home

Activate Policies with correct documents

Fine	Error	Firstname	Surname	National ID	Cover	Document	Payment
Click	Click	tarisiro	urayayi	78-6748336-A-78	Legal	Download	cash
Click	Click	mormo	mihloro	28-2345678-M-58	Hospital	Download	cash

To view the submitted documents, the underwriter downloads the submitted documents by clicking on the download option under the document module.



Admin Home

Activate Policies with correct documents

Fine	Error	Firstname	Surname	Natio
Click	Click	tarisiro	urayayi	78-67
Click	Click	mormo	mihloro	28-23

Opening 1365570827_GOQ.doc

You have chosen to open

1365570827_GOQ.doc

which is a: Microsoft Word Document

from: http://localhost

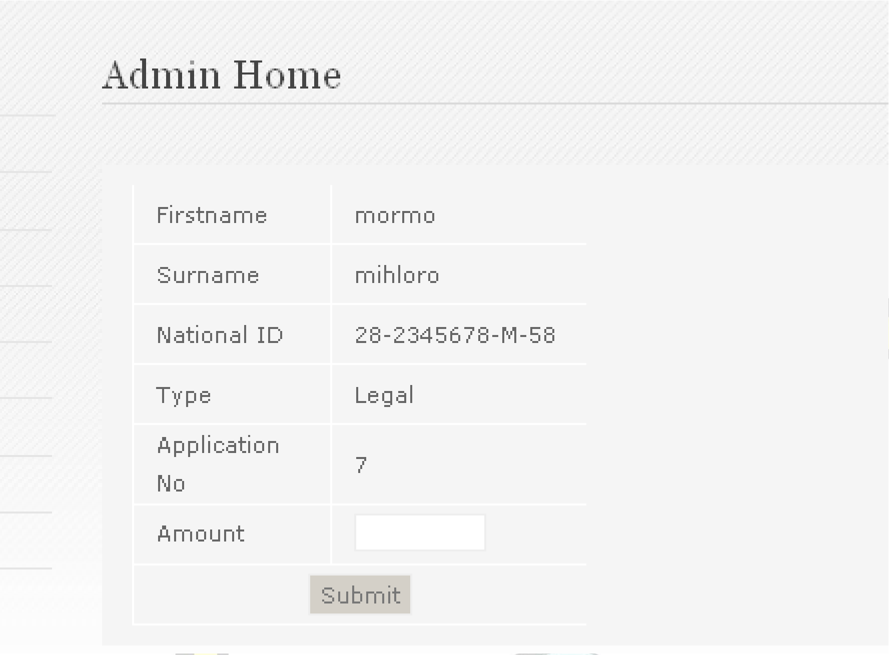
What should Firefox do with this file?

● Open with Microsoft Office Word (default)

○ Save File

□ Do this automatically for files like this from now on.

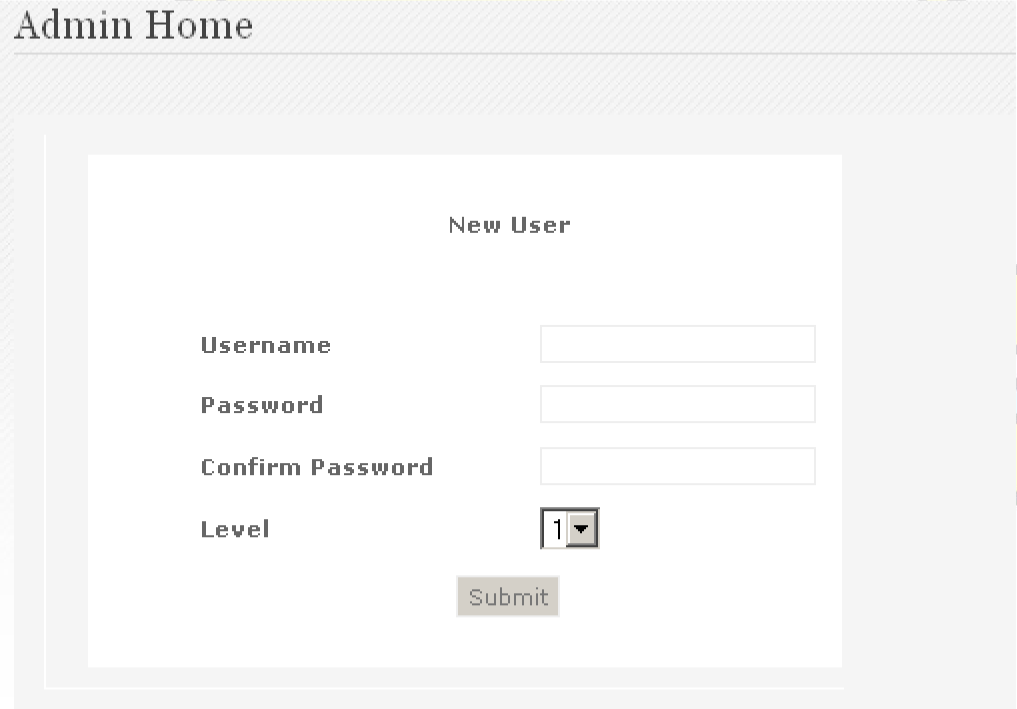
The underwriter is able to update the monthly premiums paid by the clients by clicking on the payments module and submit the national ID of the client. The following form will be displayed that allows the underwriter to update the clients' account.



Admin Home

Firstname	mormo
Surname	mihloro
National ID	28-2345678-M-58
Type	Legal
Application No	7
Amount	

The following form is for the system administrator. The system administrator adds underwriters to the system and assigns them their respective levels.



Admin Home

New User

Username

Password

Confirm Password

Level 1

APPENDIX B: Interview Guide

For the Insurance Administration System

Section 1:

Location: ………………………………………………………………

Interviewer: ………………………………………………………..

Interviewee: ……………………………………………………….

Section 2:

(1) What type of insurance policies do you give to your members?

(2) What are the conditions for one to qualify to get an insurance policy from your Institution?

(3) What is your minimum salary requirement?

(5) How is the insurance claims process conducted?

(6) Do you think a web-based collaborated database will improve effectiveness and efficiency in insurance policy management and also reduce risks?

(8) What problems are you currently facing?

(9) Are you currently connected to the internet, if no will you be able to get connected in the next 3 years?

(10) How do you cater for clients' needs at different levels and in different geographical areas in accessing information and making claims?

APPENDIX C: Questionnaire Checklist

For the Insurance Administration System

I am Mornisher Mihloro, a student at the Midlands State University in Zimbabwe pursuing a Degree in Information Systems. One of the requirements for this award is a project report with a running software system. This questionnaire is designed to find out the requirements for the above mentioned system (IAS). I therefore kindly request you to assist me with the required information in this questionnaire. I promise to keep all the given information confidential and highly guard due rights. Thank you for your cooperation.

Name of institution................................

Gender male □ female □

Job Title ……………………………………………………………………….

Qn1: What conditions would make a client qualify to get an Insurance policy?

(i)..

(ii)..

Qn2: What type of requirements is needed for the application? *(Please tick where appropriate)*

Asset □ Payslip □

Other (*Please specify***)**

………

Qn3: How long does it take to respond to a clients' application?

……………………………………………….

Qn4: How do clients pay for their monthly premiums?

....

....

Qn5: How do you update the clients' account for the premiums?

....

....

Qn6: Can a client view his/her premium statement? (*Please tick where appropriate*)

Yes □ No □

Qn7: If the answer to question 7 is yes, how does the client view his premiums statements?

……

……

Qn8: Can one account for a client hold different insurance policies?

Yes □ No □

Qn9: What are the requirements for an insurance claim?

(1)…………………………………………………………………………………………

(2)…………………………………………………………………………………………

(3)…………………………………………………………………………………………

Qn10: How long does it take for the insurance claim to be approved?

…………………………………………………

Qn11: What method are you currently using for the insurance administration (*Please tick where appropriate*)

Manual □ computerised (excel) □

Qn12: What challenges are you experiencing with the current system?

……

……

……

……

Qn13: Do you think a web-based IAS will improve the Insurance Administration and improve risk management?

Yes □ No □ Not sure □

APPENDIX D: Observation Score Sheet

For the Insurance Administration System

Name of observant: ………………………………................

Name of observer: …………………………………………

Focus of observation: …………………………………………………………

Brief description of the system

………

………

………

………

………

………

………

………

………

Area of strength

………

………

………

………

………

Areas for development:

………

………

Signed_ observant……………………….

Signed _observer………………………….

APPENDIX E: Snippet of code

Update premium

```
<?php
while($row=mysql_fetch_array($rs))
{
$date=date("d-m-Y",$row["date"]);
?>
<tr class="row-a">
<td class="first"><a href="adminhome.php?link=pactivation.php && id=<?php echo $row["id"]; ?>">Click</a></td>
<td><a href="adminhome.php?link=pdeactivation.php && id=<?php echo $row["id"]; ?>">Click</a></td>
td><?php echo $row["firstname"]; ?></td>
td><?php echo $row["surname"]; ?></td>
<td><?php echo "$".nl2br($row["amount"]); ?></td>
<td><?php echo $row["type"]; ?></td>
<td><a href="uploads/<?php echo $row["document"]; ?>">Download</a></td>
<td><?php echo $date; ?></td>
</tr>
<?php
}
?>
</table>
</body>
</html>
```

Activation of applications

```
<?php
while($row=mysql_fetch_array($rs))
{
?>
<tr class="row-a">
<td class="first"><a href="adminhome.php?link=activation.php && id=<?php echo
$row["id"]; ?>">Click</a></td>
<td><a href="adminhome.php?link=deactivation.php && id=<?php echo $row["id"];
?>">Click</a></td>
<td><?php echo $row["firstname"]; ?></td>
td><?php echo $row["surname"]; ?></td>
<td><?php echo $row["nationalid"]; ?></td>
<td><?php echo $row["type"]; ?></td>
<td><a href="uploads/<?php echo $row["document"]; ?>">Download</a></td>
<td><?php echo $row["payment_type"]; ?></td>
</tr>
<?php
}
?>
</table>
<p>&nbsp;</p>
</body>
</html>
```

Admin login details

```
<?php
session_start();
include "includes/opendb.php";
$rs= mysql_query("SELECT * FROM admin WHERE username='$_POST[username]' and
password='$_POST[password]'");
if($row=mysql_fetch_array($rs))
{
$username=$row["username"];
$_SESSION['username']=$username;
$_SESSION['logged'] = true ;
$_SESSION['id']=$row["id"];
$_SESSION['level']=$row["level"];
?>
<script language="javascript">
window.location="adminhome.php";
</script>
<?php
}
else
{
echo "<script language='javascript'> alert('wrong username or password');
window.location='index.php'; </script>";
}
?>
```

Backup

```
$query = mysql_query("select * from $table");
$num_fields = mysql_num_fields($query);
$numrow = mysql_num_rows($query);
while( $row = mysql_fetch_array($query, MYSQL_NUM))
{
$result .= "INSERT INTO ".$table." VALUES(";
for($j=0; $j<$num_fields; $j++)
{
$row[$j] = addslashes($row[$j]);
$row[$j] = str_replace("\n","\\n",$row[$j]);
$row[$j] = str_replace("\r","",$row[$j]);
if (isset($row[$j]))
$result .= "\"$row[$j]\"";
else
$result .= "\"\"";
if ($j<($num_fields-1))
$result .= ", ";
}
$result .= ");\n";
}
if ($i+1 != mysql_num_rows($tables))
$result .= "\n";
}
if((isset($_GET['action'])) && ($_GET['action'] == "save"))
{ ob_clean();
ob_start();
Header("Content-type: application/octet-stream");
Header("Content-Disposition: attachment; filename=$backupFile");
echo $result;
```

```
ob_end_flush();
echo "Backup Taken Successfully!!!";
exit;
}
?>
```

Claim application

```
<?php
while($row=mysql_fetch_array($rs))
{
$date=date("d-m-Y",$row["date"]);
?>
<tr class="row-a">
<td class="first"><a href="adminhome.php?link=claim1activation.php && id=<?php echo $row["id"]; ?>">Click</a></td>
<td><a href="adminhome.php?link=claim1deactivation.php && id=<?php echo $row["id"]; ?>">Click</a></td>
<td><?php echo $row["firstname"]; ?></td>
<td><?php echo $row["surname"]; ?></td>
<td><?php echo nl2br($row["description"]); ?></td>
<td><?php echo $row["type"]; ?></td>
<td><a href="uploads/<?php echo $row["document"]; ?>">Download</a></td>
<td><?php echo $date; ?></td>
</tr>
<?php
```

Application status

```
<?php
while($row=mysql_fetch_array($rs))
{
?>
<tr class="row-a">
<td class="first"><?php echo $row["id"]; ?></td>
<td><?php echo $row["type"]; ?></td>
<td><?php echo $row["description"]; ?></td>
 <td>
<?php
if($row["status"]==0)
echo "Application Sent";
}
if($row["status"]==1)
                                        {
```

```
echo "Application Approved";
}
if($row["status"]==-1)
{
echo "Application Declined";
}
?></td>
</tr>
<?php
```

Register

```
<tr>
<td>Firstname</td>
<td><input type="text" name="firstname" id="firstname" /></td>
</tr>
<tr>
<td>Surname</td>
<td><input type="text" name="surname" id="surname" /></td>
</tr>
<tr>
<td>National ID</td>
<td colspan="3"><input name="id1" type="text" id="id1" size="2" maxlength="2" />
<input name="id2" type="text" id="id2" size="7" maxlength="7" />
<select name="id3" id="id3">
<option value="A">A</option>
<option value="B">B</option>
<option value="C">C</option>
<option value="D">D</option>
<option value="E">E</option>
<option value="F">F</option>
<option value="G">G</option>
<option value="H">H</option>
 <option value="I">I</option>
<option value="J">J</option>
<option value="K">K</option>
<option value="L">L</option>
<option value="M">M</option>
<option value="N">N</option>
<option value="O">O</option>
<option value="P">P</option>
<option value="Q">Q</option>
<option value="R">R</option>
```

```
<option value="S">S</option>
<option value="T">T</option>
<option value="U">U</option>
<option value="V">V</option>
<option value="W">W</option>
<option value="X">X</option>
<option value="Y">Y</option>
<option value="Z">Z</option>
</select>
<input name="id4" type="text" id="id4" size="2" maxlength="2" /></td>
</tr>
<tr>
<td>Date of Birth</td>
<td><input type="text" name="asdate" id="asdate" /></td>
</tr>
<tr>
<td>Address</td>
<td><textarea name="address" id="address" cols="45" rows="5"></textarea></td>
</tr>
<tr>
<td>Username</td>
<td><label>
<input type="text" name="username" id="username" />
</label></td>
</tr>
<tr>
<td>Passsword</td>
<td><label>
<input type="password" name="password" id="password" />
</label></td>
</tr>
<tr>
<td>Confirm Password</td>
<td><label>
<input type="password" name="password1" id="password1" />
</label></td>
</tr>
<tr>
<td colspan="2" align="center"><input type="submit" name="button" id="button"
value="Submit" /></td>
</tr>
</table>
</form>
<?php
```

```
if(isset($_POST['button']))
{
if(is_numeric($_POST['firstname']))
{
?>
<script language="javascript">
 ("Enter Characters on firstname");
</script>
<?php
exit();
}
if(is_numeric($_POST['surname']))
  {
?>
<script language="javascript">
("Enter Characters on surname");
</script>
<?php
exit();
}
/* if(is_numeric($_POST['nationality']))
{
<!--
<script language="javascript">
alert("Enter Characters on nationality");
</script>
<?php
if(($_POST['password'])<>($_POST['password1']))
<script language="javascript">
alert("Password Mismatch");
</script>
<?php
exit();
if( strlen($_POST["id1"]) < 2)
{
?>
<script language="javascript">
alert("Invalid input on 1st National ID input");
</script>
<?php
exit();
}
if( strlen($_POST["id2"]) < 6)
{
```

```
?>
<script language="javascript">
alert("Invalid input on 2nd National ID input");
</script>
<?php
exit();
}
if( strlen($_POST["id4"]) < 2)
{
?>
<script language="javascript">
alert("Invalid input on 3rd National ID input");
</script>
<?php
exit();
}
$today=strtotime("now");
$important=strtotime("now - 16years");
$important1=$_POST["asdate"];
$duedate=strtotime("$important1");
if($duedate>$today)
{
?>
<script language="javascript">
alert("Invalid Date of Birth");
window.location="register.php?link=registerdetails.php";
</script>
<?php
exit();
}
$nationalid=$_POST["id1"]."-".$_POST["id2"]."-".$_POST["id3"]."-".$_POST["id4"];
$rs=mysql_query("select * from member where nationalid='$_POST[nationalid]'") or
die(mysql_error());
$count=mysql_num_rows($rs);
if($count==1)
{
?>
<script language="javascript">
alert("Member account already exist");
window.location="register.php?link=registerdetails.php";
</script>
<?php
exit();
}
```

```
mysql_query("INSERT INTO member
VALUES(NULL,'$_POST[firstname]','$_POST[surname]','$nationalid','$duedate','$_POST[
address]','$_POST[username]','$_POST[password]','$today')");
?>
<script language="javascript">
alert("Registration Updated");
window.location="register.php?link=registerdetails.php";
</script>
<?php
}
?>
```