



A WEB BASED SYSTEM FOR AIRLINE RESERVATION IN KENYA

CASE STUDY: KENYA AIRWAYS_MOMBASA

BY

KIYAYI .N. VIHENDA

BCS/7510/71/DF

A Graduation Project Report Submitted to the School of Computer Studies

In Partial Fulfillment of the Requirements for the Award of

Bachelor of Computer Science of

Kampala International University

July, 2009

Declaration

I Kiyayi N Vihenda do hereby declare to best of my knowledge that this graduation project is my original work and that it has never been submitted to any University or any other institution.

Signature 

Date... 22-07-2009

Kiyayi N Vihenda

Student

Approval

This Graduation Project report titled “A Web Based System for Airline Reservations” has been submitted with the approval of the following supervisor.

Signed  ,

Date 22-07-2009

Mr. Taremwa Danison

Supervisor

Dedication

This project research is dedicated to my beloved parents Mrs. and Mr. Kiyayi Henry Analo who have tirelessly supported me in all ways, both morally and financially together with the entire family especially my elder sister Maureen for their encouragement, prayers and moral support. May the Almighty God bless them abundantly for their good work they have done into my entire studies.

Acknowledgement

My appreciation and sincere gratitude go to my competent professional supervisor Mr. Taremwa Danson for his valuable time, encouragement, guidance and supervision during the course of the study. Without him, this book would not have been what it is. My humble thanks go to all the lecturers and entire staff of Computer Studies of Kampala International University who have equipped me with the necessary skills and knowledge to apply throughout the entire project.

Special thanks to all my friends and classmates of Kampala International University and for those whose names have not appeared here, I will always remember your contributions which are engraved deep in my heart and you will be acknowledged in my mind always. Also thanks to the general staff at the Moi International Airport (MIA) for allowing the researcher to carry out her research in their organization. Special thanks to Mr. Mwaniki who enabled me to be allowed in the organization and directed the researcher to fulfill her research project. Thank you and the Almighty God bless you all.

Finally, I will not forget the Almighty God who has given me the will and guidance throughout this time of developing the project. It is by His grace that I have come this far.

Abbreviations and Acronyms

R&T	Reservation and Ticketing System
PHP	Hypertext Preprocessor
JSP	Java Server Pages
DBMS	Database Management Systems
RAM	Random access memory
MB	Megabytes
GB	Gigabytes
OOA	Object Oriented Analysis
DAD	Development and Design
GUIs	Graphical User Interfaces
SDLC	System Development Life Cycle
CFMX	Cold Fusion Markup Language
ASP	Active Server Pages
JKIA	Jomo Kenyatta International Airport
MIA	Moi International Airport
KAA	Kenya Airport Authority
EAA	East African Airways Corporation

List of Figures and Tables

Figure 2.6: Layout Design for the Web Pages.....	page (8)
Figure 3.0: An illustration of a systems' development methodology for airline reservation.....	page (5)
Figure 3.2.3: An illustration of a block diagram of system operation	page (30)
Figure 4.4.1: Context Diagram for the Current System Process	page (37)
Figure 4.5.1.1: Use Case Diagram for the Ticket Reservation System	page (39)
Figure 4.5.1.2: System Flow Chart for the Reservation Ticket System	page (40)
Figure 4.6.1: Activity Diagram of Reservation and Issue of Tickets	page (41)
Table 4.6.1: Users-Roles-Use cases	page (43)
Figure 4.6.2.1: Activity Diagram of Use Case Inquiry to Issue of Tickets	page (44)
Figure 4.6.2.2: Activity Diagram of Use Case on Inquiry of Tickets	page (45)
Figure 4.6.2.3: Activity Diagram of Use Case on Reservation and Ticketing	page (46)
Figure 4.7: Interface for creating the tables'	page (47)
Figure 5.1.1: Site definition	page (49)
Figure 5.1.2: Template Interface	page (50)
Figure 5.1.3: Interface for creating the database connection	page (51)
Figure 5.2.1: Login Form for the Airline authorized users'	page (52)
Figure 5.3: User Page	page (53)
Table 5.3: Reservation table	page (54)
Figure 5.4.1: Testing Reservation Page	page (55)
Figure 5.4.2: Validating Reservation Page	page (56)
Figure 5.4.3: Testing Database Connectivity	page (57)

Abstract

The report provides a technical analysis and reviews of a database driven website which has been designed and constructed by the researcher to assist the MIA and other staff members of the organization in the processing and storage of the chores carried out in the airline reservations system of the Kenya Airways. In designing the database driven website, the researcher has used Macromedia Dreamweaver 8, PHP language and MySQL server. In the design of the user interfaces, Macromedia Dreamweaver 8 has been used for its support in incorporation of graphics in the process of PHP programming.

The study involved the design and implementation of a web based system for Kenya Airways in Mombasa that will provide easy, accurate and in time retrieval of all passengers' records and their details. The problem was analyzed carefully and steps to solve it were drawn. Various literatures were reviewed in chapter two of this document. This was done in order to get an overview of what has been done before concerning the problem. An overview of web based systems is given as well as various concepts concerning its implementation. The development methodology and tools has been discussed in detail in chapter three. This includes knowledge acquisition and analysis, data collection techniques, development tools and system validation. Details of the new system like system specification and data flow diagrams for the new system are discussed in chapter four. System testing, data conversion and loading and system evaluation details has been also discussed in chapter five. Finally, chapter six contains the conclusion, recommendations and areas for further research.

Table of Contents

Declaration	i
Approval	ii
Dedication	iii
Acknowledgement	iv
Abbreviations and Acronyms	v
List of Figures and Tables	vi
Abstract	vii
Table of Contents	viii
CHAPTER ONE: INTRODUCTION	1
1.1 Background of the Study	1
1.2 Problem Statement	3
1.3 Objectives of the Study	3
1.3.1 General Objective	3
1.3.2 Specific Objectives	3
1.4 Scope of the Study	4
1.5 Significance of the Study	4
CHAPTER TWO: LITERATURE REVIEW	6
2.1 Web Based Systems	6
2.1.1 Benefits of Web Based Systems	7
2.1.2 Drawbacks of Web Based Systems	8
2.2 Evaluation of Current Web Based Systems in use	9
2.2.1 ColdFusion	9
2.2.2 JavaServer Pages (JSP)	9
2.2.3 Lasso	10
2.2.4 Perl	11
2.2.5 Python	12
2.2.6 Ruby	12
2.2.7 Hypertext Preprocessor (PHP)	13
2.2.8 ASP	13
2.2.9 ASP.NET	14
2.3 Planning and Development of Web Based Systems	17
2.3.1 Development Cycle	17
2.3.2 XAMPP	19
2.3.3 Macromedia Dreamweaver 8	19
2.4 Components of Web Based Systems	21
2.4.1 MySQL	21
2.4.2 Microsoft Access	22
2.4.3 SQL Server	22
2.4.4 Database	23
CHAPTER THREE: METHODOLOGY	25

3.1	Data Collection.....	26
3.1.1	Documentary Review.....	26
3.1.2	Observation.....	26
3.1.3	Questionnaire.....	27
3.1.4	Interview.....	28
3.2	System Design.....	28
3.2.1	User Interface Design.....	28
3.2.2	Database Design.....	29
3.2.3	The Data Integration Framework.....	29
3.3	System Implementation.....	30
3.4	Coding.....	30
3.5	System Testing and Validation.....	31
CHAPTER FOUR: SYSTEM ANALYSIS AND DESIGN.....		32
4.1	Investigation of the Current System.....	32
4.2	Description of the Proposed System.....	33
4.3	Requirements of the Proposed System.....	34
4.3.1	System Requirement.....	34
4.3.2	User Requirements.....	34
4.3.3	Non Functional Requirements.....	35
4.3.4	Specific Requirements.....	35
4.3.5	Benefits of the Proposed System.....	36
4.4	Current System Processes.....	37
4.4.1	Context Diagram for the Current system's Processes.....	37
4.5	System Design.....	37
4.5.1	Logical Design.....	38
4.6	Program Specification.....	40
4.6.1	Activity Diagram of Reservation and Issue of Tickets.....	41
4.6.2	Activity Diagram of Use Case Inquiry to Issue of Tickets.....	43
4.7	Physical Design.....	46
CHAPTER FIVE: SYSTEM IMPLEMENTATION.....		48
5.1	Website Design.....	48
5.2	User Interface Design.....	51
5.2.1	Typical Functions of the User Interface.....	51
5.3	Interactions Between the User Interface and the Database.....	52
5.4	System Testing and Validation.....	54
5.5	File Creation and Conversion.....	58
5.6	Parallel Change Over.....	58
CHAPTER SIX: DISCUSSION, CONCLUSION AND RECOMMENDATION.....		60
6.1	Discussion.....	60
6.2	Conclusion.....	61
6.3	Recommendations.....	62
6.3.1	Areas for Further Research.....	63
BIBLIOGRAPHY.....		64
APPENDICES.....		66

Appendix A: Questionnaire Guide.....	66
Appendix B: Sample Codes.....	69

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

Kenya Airways traces its history back to 1946 with the formation of the East African Airways Corporation ("EAA"). Initially, EAA had a good reputation for service and reliability. With the formation of the East African Community, EAA passed into the joint ownership of the governments of Kenya, Tanzania, and Uganda. Shortly after the collapse of the East African Community in 1976, EAA was placed in liquidation. Kenya Airways was incorporated in January 1977 as a company wholly owned by the Kenyan government. It was established as the national flag carrier of Kenya and acquired certain of the assets and staff of EAA. Kenya Airways is the leading operator on domestic routes. Kenya Airways operates 67 flights a week to four domestic destinations: Mombassa, Malindi, Kisumu, and Nairobi. The most important of its domestic routes is Nairobi-Mombasa, on which Kenya Airways operates 56 flights a week (Â©2008 Kenya Airways [1]).

Kenya Airways operates scheduled passenger service and cargo services to 24 international destinations with 45 flights a week. Kenya Airways serves 7 destinations in Europe; 11 in sub-Saharan Africa; and 6 in North Africa, Asia, and the Middle East. Kenya Airways, as the national airline of Kenya, has rights under existing bi-lateral agreements to operate flights to a total of 58 countries. Privatization and the formation of cross-border alliances are trends in the airline industry and in many other industries. This case study examines the privatization of Kenya Airways as a means for fostering discussion and critical thinking about privatization, liberalization, and enterprise restructuring. Kenya straddles the equator on the east coast of Africa. It has an area of 583,000 square kilometers (slightly larger than France) and a population of about 35 million persons. Nairobi, with a population of 1.3 million persons, is the capital of Kenya. Kenya has been an independent republic since 1963.

English is the official language and is used in business throughout the country. Kiswahili is the Kenyan national language. The legal system is based on English common law.

There is considerable on-line material available about Kenya. Two large sites are the Kenya section of Africa Online and Kenya web. The air passenger transport market in Kenya currently amounts to about two million air journeys (domestic plus international) from the airports in Nairobi and Mombasa. The air freight market is currently about 90,000 tons of cargo and mail. Most international visitors are European tourists, 50% of which arrive on chartered flights. Traffic on the Middle East/Asian routes consist mainly of foreign expatriates based in the Middle East and traveling to Kenya for vacation, Kenyans who work in the Gulf region, and family visitors and business travelers between Kenya and India. The majority of travelers on sub-Saharan routes consist of business travelers.

Tourists travel to Kenya to see its abundant and diverse wild life, experience the natural beauty of the land, and to enjoy Kenya's colorful multifaceted culture. In 1993, Kenya hosted over 830, 00 visitors, of which about 680,000 were on holiday. Most of these visitors arrived to Kenya by air. Among all African countries in 1993, Kenya ranked in the top 5 for both international tourist arrivals and tourism receipts. Over the past decade, Africa recorded growth in tourist visits of 8.2%, the second highest growth rate among world regions and significantly higher than the world average of 5.5%. The World Tourism Organization projects that tourism to Kenya will grow at an annual rate of about 6% through the year 2010.

Kenya has two international airports: Jomo Kenyatta International Airport (JKIA) in Nairobi, and Moi International Airport (MIA) in Mombasa. JKIA, which handled 2.6 million passengers in 1993, is the primary airport for all international scheduled air traffic. The terminal complex, taxiways, aprons, holding and maintenance facilities were built in 1978. MIA is a smaller international airport, handling about 900,000 passengers in 1993. MIA has recently been renovated and expanded. Foreign charter companies are allowed to operate only from MIA. There are plans for constructing a new international airport at Eldoret, a major horticultural center in the western part of Kenya.

There are also domestic airports in Nairobi, Kisumu, and Malindi. Kenya Airport Authority (KAA) operates all airports and runways in Kenya. KAA authorizes airlines to use airport facilities and compiles traffic data (Roberts,2001) [2]).

1.2 Problem Statement

According to the watchdog group Transparency International 2005, Kenya is regularly ranked among the ten most attractive flight countries in the world. Lack of a web based system to keep track of customers' airline reservations has led to delay of customers to get serviced. This has costed customers a lot of time and hence makes the whole process very costly. The implemented system has provided people at the grassroots the opportunity to make request decisions that maximize their welfare according to their level of earnings but with the lack of a system to sensitize this, it is difficult for the citizens and the tourists to do this. Lastly views and complaints by users are sent by mail, the chances of these mail getting lost are high and also the amount of effort and time used to sort and go through them is much.

1.3 Objectives of the Study

1.3.1 General Objective

The general objective of this study was to develop a web based information system that may be accessed by customers' for airline reservations for Kenya Airways in Mombasa.

1.3.2 Specific Objectives

- i. To investigate the existing literature on what is required in designing and implementation of a web based database driven website for Kenya Airways.
- ii. To investigate approaches on development of web based system for air reservations.
- iii. To design a web based database driven website for Kenya Airways system basing on the requirements collected from the research.
- iv. To implement the designed website for Kenya Airways.
- v. To test and validate the proposed system for Kenya Airways.

1.4 Scope of the Study

Kenya has two international airports: Jomo Kenyatta International Airport (JKIA) in Nairobi, and Moi International Airport (MIA) in Mombasa. The researcher's main study was about Moi International Airport (MIA) in Mombasa which is a smaller international airport, handling about 900,000 passengers in 1993 that are from foreign charter companies. MIA has recently been renovated and expanded. The system developed is limited to the customers' requests for the flight, booking process online and orders that are offered by the Kenya Airways for its customers. Any services to the customer that is not a request online for a flight detail which is not updated in the database is not included in the study.

1.5 Significance of the Study

In today's competitive world, web based database driven website provide competitive advantage to businesses due to increase in the amount of information processed and precautions taken on how this information should be stored. There has been an increase on the amount of data created, stored, and retrieved by most organizations (Kjetil, 2002; Beynon Davies, 2003[3]). Since it is every manager's dream to get new customers as well as maintaining the existing customers, the implemented dynamic website enables easy tracking of the company customers as well as their flight booking requests.

The designed system has helped to overcome the problem of unauthorized data access by ensuring security and data integrity with the clients' data. This was achieved by creating authenticated usernames and passwords giving access to only those authorized parties to avoid unauthorized data alteration in the system.

The implementation of this project has ensured the efficiency and effectiveness in updating its users' requests through their email accounts in terms of requests. This eliminates the problems of tourists and customers losing confidence in the Kenya Airways services and operations.

The developed system has helped in to boosting Kenya's image and economy in general. This is where most tourists and customers take flight through booking online in different parts of Kenya and even outside East Africa like United Kingdom thus raising the Kenya revenue to the government through the foreign revenue.

The development of this system has enabled the Kenya Airways operation and management to achieve its core value like transparency, accountability, equality in resource distribution and efficient use of its services through the tracking of its services and operation.

The degeneration of Kenya's economy due to mismanagement of funds has been solved through the introduction of the new system which captures the country's economy through the tracking of the customers' views and the requests and its operation.

Academically, the study has helped the researcher to gain practical skills giving the researcher a clear overview of how to design and implement what is studied theoretically in class. The success of this project also contributes to the fulfillment of the requirement of a Bachelors Degree in Computer Science through the application of all the general knowledge taught theoretically in class to real life situation.

CHAPTER TWO

LITERATURE REVIEW

This chapter gives the researcher different ideas which were generated by different scholars about web based driven websites. These ideas gave the researcher a foundation basing on analysis of the new system design. The researcher reviewed on existing work done by different researchers based on online reservations, usage, merits and demerits of web based systems and how various requirement stage undergone to develop the design.

2.1 Web Based Systems

A web based application is a software package that can be accessed over the internet through web browser. The software and database reside on a central server rather than being installed on the desktop system and is accessed over a network. A Website is a collection of Web pages, images, videos or other digital assets that is hosted on one or several Web server(s), usually accessible via the Internet, cell phone or a LAN. There are basically two types of websites pages that is static/non-dynamic websites pages which is a website that has web pages stored on the server in the same form as the user will view them and dynamic websites which is a website is that has frequently changing information or collates information on the hop each time a page is requested or information is fed or needed from a database (Lowery, 2006) [7].

A website (alternatively, website or Website) is a collection of Web pages, images, videos or other digital assets that is hosted on one or several Web server(s), usually accessible via the Internet, cell phone or a LAN (Cailliau, 2007) [6]. A Web page is a document which is always accessible via HTTP, a protocol that transfers information from the Web server to display in the user's Web browser. Websites are written in, or dynamically converted to, HTML (Hyper Text Markup Language) and are accessed using a software interface classified as a user agent. Web pages can be viewed or otherwise accessed from a range of computer-based and Internet-enabled devices of various sizes, including desktop computers, laptop computers and cell phones.

After each Web page is created, they are typically linked together using a navigation menu composed of hyperlinks (Lowery, 2006) [7].

Once a web based system is completed, it must be published or uploaded in order to be viewable to the public over the internet. This is done using an FTP client. All publicly accessible websites are seen collectively as constituting the "World Wide Web (WWW)". The pages of web based systems can usually be accessed from a common root URL called the homepage, and usually reside on the same physical server. The URLs of the pages organize them into a hierarchy, although the hyperlinks between them control how the reader perceives the overall structure and how the traffic flows between the different parts of the sites. Some websites require a subscription to access some or all of their content. A website should focus on a specific subject and provide a variety of information that relates to the subject (Lowery, 2006) [7]. Websites are constrained by architectural limits e.g., the computing power dedicated to the website. The first on-line website appeared in 1991 where CERN announced that the World Wide Web would be free to anyone (Cailliau, 2007) [6]. There are basically two types of websites that is static/non-dynamic websites which is a website that has web pages stored on the server in the same form as the user will view them and dynamic websites which is a website is that has frequently changing information or collates information on the hop each time a page is requested (Ramyalg, 2005) [8].

2.1.1 Benefits of Web Based Systems

Gianni 2002 [10] states that querying a database that sits behind a search engine is merely the tip of the iceberg when it comes to using a database to provide dynamic content for your website visitors. The following are some of the benefits that can be achieved by employing a database to drive a website. Much like a search engine, you can provide a directory of contacts or events on the Web that can be searched in a number of ways. The data is stored in a database and can be searched through a Web interface.

If you have a large number of documents that you want to make available to the public on the Web, you might consider developing a database-driven library of these documents that would allow people to search for certain documents or browse lists of categorized documents (Gianni, 2002) [10].

Aggregating data from surveys can be a valuable way for you to evaluate programs or collect information from your members or people who care about your work. Sending out paper surveys and doing the data entry to analyze your results can be frustrating and time consuming, so using the Web to collect that information is a great way to simplify the process (Gianni, 2002) [10].

If your Website contains a large number of pages or you have a number of different people involved in updating and maintaining your site, you may consider using a content management system to maintain a consistent look and feel for all your pages. Rather than developing a number of static Web pages, you develop the content and plug it into a template for your Website. The information is stored in a database (Stefan, 2000) [9].

2.1.2 Drawbacks of Web Based Systems

According to (Lowery, 2006) [7], the drawbacks of web based systems are as follows:

- i. Inconsistency of data which causes inflexibility whereby it is very difficult to change data and programs once they have been implemented.
- ii. Limited data updates which causes poor enforcement of standards, poor programmer productivity and excessive program maintenance.
- iii. The biggest problem with the ASP based model web based systems is its speed and reliability of the Internet. If your internet goes down, then you can't access your POS software. In addition, the speed of your POS will be determined by the performance of your internet connection. If your internet is slow, it will take a little longer to switch screens and run the software.

2.2 Evaluation of Current Web Based Systems in use

2.2.1 ColdFusion

ColdFusion is a program based on the ColdFusion Markup Language (CFMX) programming language that builds and deploys powerful web applications and services. With proven tag-based methodology, the benefits of ColdFusion's features is that it allows web developers to easily harness the power of the Java Platform to deliver powerful Internet applications to millions of people worldwide (Swanson, 2004) [16].

ColdFusion is a server side language, which is executed by a ColdFusion application server. It works by executing CFM templates (which is a mixture of HTML and ColdFusion instructions) to return normal HTML to the requesting browser. The ColdFusion MX Enterprise software suite provides all the necessary software to create large-scale websites, applications, and hosting environments, and is available for either a stand-alone installation for web developers or for industry-level deployment on J2EE application servers (Downes, 2003) [15]. ColdFusion MX supports most major databases including Microsoft SQL, Microsoft SQL Server, Oracle, IBM DB2, Sybase and IBM Informix. In addition, CFMX can run on most major web servers including IIS, Apache and Sun One. With ColdFusion, you can build and deploy powerful web applications and services with far less training time and fewer lines of code than ASP, PHP, and JavaServer Pages (JSP). Macromedia ColdFusion MX supports most operating systems including Windows, UNIX, Linux, IBM AIX and HP-UX (Swanson, 2004) [16]. The drawbacks of ColdFusion is that it requires experts to use it.

2.2.2 JavaServer Pages (JSP)

JavaServer Pages (JSP) is a Java technology that allows software developers to dynamically generate HTML, XML or other types of documents in response to a Web client request. The technology allows Java code and certain pre-defined actions to be embedded into static content (Hans, 2003) [18].

The JSP syntax adds additional XML-like tags, called JSP actions, to be used to invoke built-in functionality. Additionally, the technology allows for the creation of JSP tag libraries that act as extensions to the standard HTML or XML tags.

Tag libraries provide a platform independent way of extending the capabilities of a Web server. JSPs are compiled into Java Servlets by a JSP compiler. A JSP compiler may generate a servlet in Java code that is then compiled by the Java compiler, or it may generate byte code for the servlet directly. JSPs can also be interpreted on-the-fly reducing the time taken to reload changes (Hans, 2003) [18].

JSP is a fairly flexible server technology too, as it allows you to extend and enhance its functionality in a similar way to CFML, providing support for third-party extensions, or "tag libraries". Its syntax and usage is also not dissimilar to PHP and JavaScript, as they all have a common inspiration root.

However, whilst both PHP and JSP share a common background of sorts, JSP is much more difficult to learn and less user-friendly than PHP (Downes, 2003) [15], hence the need for other types like Lasso as explained below.

2.2.3 Lasso

Lasso is a tag-based programming/scripting language used to send instructions to the Lasso Web Data Engine (WDE). The Lasso WDE runs on a web server as a plug-in or CGI, where it is typically used as middleware to enable the web server software to talk to a back-end database (Porter, 2008) [19].

While the Lasso language itself is fairly high-level, it is nevertheless a reasonably complex language with hundreds of tags, and the fact that Lasso code gets scattered around in the HTML code of web pages makes writing it and reading it somewhat difficult. Until recently, Lasso developers had no choice but to rely entirely upon text-editors like BBEdit to do their coding. But with the release of the Lasso Studio for Dreamweaver (LS/DW) in December 1999, Lasso developers finally got a graphical interface to work with. And with the version 1.5 update, released in early 2000. The LS/DW is ready for prime time. Lasso hasn't gotten any easier and doing advanced work with Lasso still requires that you get your hands dirty (Porter, 2008) [19]. Lasso allows you to create a custom Web page which displays data from your database to site visitors. You can allow site visitors to search the database or to add, update, and delete records from your database.

Your Web pages will be written in a combination of standard HTML and Lasso's programming language LassoScript. Lasso has a built-in security model which allows you to determine what database action each visitor can perform. Lasso can connect to many other databases such as: FileMaker, MySQL, Oracle, Microsoft SQL Server, PostgreSQL, OpenBase, and SQLite (Porter, 2008) [19].

2.2.4 Perl

Perl is either an abbreviation of "Practical Extraction and Report Language" or "Pathologically Eclectic Rubbish Lister". In 1987, Larry Wall started to develop Perl (Borchert, 2002) [20].

Advantages of Perl

- a. Perl and a huge collection of Perl modules are free software (either General Public License (GNU) or Artistic License).
- b. Perl runs on all platforms and is far more portable than other environments. Perl/Tk applications or CGI scripts in Perl can be ported between UNIX and Microsoft Windows without any modifications at all. Perl provides those features that are required for large projects that is;
 - i. Object-oriented techniques.
 - ii. Arbitrary data structures.
 - iii. A large number of built-in checks by the compiler and at runtime.
 - iv. Modularization.
 - v. A large number of built-in checks by the compiler and at runtime.

Disadvantages of Perl

- a. It is possible to write completely unreadable and impossible to maintain code.
- b. The language is pretty much overloaded as the designers of Perl never hesitated to add new features. The long historic development of Perl contributes to this.
- c. The syntax of Perl is non-trivial.

2.2.5 Python

Python is powerful object oriented interpreted scripting languages. Python lets you write programs that access, display and update the information in the database with minimal effort (Wesley, 1995) [21]. The various individuals who wrote the Python database modules invented their own interfaces, and the resulting proliferation of different Python modules caused problems: no two of them were exactly alike, so if you decided to switch to a new database product, you had to rewrite all the code that retrieved and inserted data. To solve the problem, a Special Interest Group (SIG) for databases was formed. After some discussion, the Database SIG produced a specification for a consistent interface to relational databases (DB-API). Thanks to this specification, there's only one interface to learn.

Porting code to use a different database product is much simpler, often requiring the change of only a few lines (Wesley, 1995) [21]. Some of the disadvantages of using Python are: More work to install on our server, no smarty, can not reuse old code.

2.2.6 Ruby

Ruby is a Dynamically Typed, Object Oriented Scripting Language, which is very popular in Japan. Until recently, it was only moderately popular in the West, but this seems to be changing that (Matz, 2003) [22]. Ruby is a language of careful balance. Its creator, Yukihiro Matsumoto, blended parts of his favorite languages (Perl, Smalltalk, Eiffel, Ada, and Lisp) to form a new language that balanced functional programming with imperative programming. Ruby is seen as a flexible language, since it allows its users to freely alter its parts. Essential parts of Ruby can be removed or redefined, at will. Existing parts can be added upon. Ruby tries not to restrict the coder. In Ruby, everything is an object. Every bit of information and code can be given their own properties and actions. While Ruby often uses very limited punctuation and usually prefers English keywords, some punctuation is used to decorate Ruby. Ruby needs no variable declarations. It uses simple naming conventions to denote the scope of variables (Matz, 2003) [22].

2.2.7 Hypertext Preprocessor (PHP)

Hypertext Preprocessor (PHP) is one of the most popular server side scripting languages running today. It is used for creating dynamic web pages that interact with the user offering customized information (Downes, 2003) [15]. PHP code is inserted directly into the HTML that makes up a website. When a visitor comes to the website, the code is executed. Because PHP is a server side technology, the user does not need any special browser or plug-ins to see the PHP in action (Swanson, 2004) [16].

PHP uses a modular system of extensions to interface with a variety of libraries such as graphics, Extensible Markup Language (XML), encryption, among others. In addition, programmers can extend PHP by writing their own extensions and compiling them into the executable or they can create their own executable and load it using PHP dynamic loading mechanism (Swanson, 2004) [16]. PHP has tons of server interfaces, database interfaces and other modules available. Of the server interfaces, PHP can load into Apache, Internet Information Server (IIS), Roxen, and AOLserver. It can also be run as a CGI module. Database interfaces are available for MySQL, MS SQL, Informix, Oracle and plenty of others. If a database is not supported, Open Database Connectivity (ODBC) is an option (Swanson, 2004) [16]. I chose to use PHP because of the following advantages provided by it; it is fast, stable, secure, easy to use, open source (free) and smart

2.2.8 ASP

Active Server Pages (ASP) is a proprietary technology, based around Microsoft's Internet Information Server. It allows you to combine HTML, Visual Basic Script (VBScript), and JavaScript with reusable ActiveX Components to create dynamic web pages. ASP requires that you purchase a version of Windows that comes with IIS. Windows NT Server, Windows 2000, and Windows XP Professional all come with a version of IIS, but it can prove to be an expensive option when installing a production server of your own. There are alternatives, however, so that ASP pages can be used on other web servers, such as Apache. Amongst these are ChilliSoft ASP, and iASP, both of which are fairly popular (Downes, 2003) [15]. Though still commercial in nature they do provide a cheaper alternative and do not commit you to the use of a Windows-based web server.

ASP has proven to be a popular choice amongst developers, as there are many relatively cheap hosting companies that provide ASP hosting via a shared server, where multiple users and websites exist on the same machine. In such an instance, you are not required to buy any form of Windows, and ASP really does become a viable option (Downes, 2003) [15].

2.2.9 ASP.NET

ASP.NET is a web application framework marketed by Microsoft that programmers can use to build dynamic websites, web applications and web services. It is part of Microsoft's .NET platform and is the successor to Microsoft's Active Server Pages (ASP) technology. ASP.NET is built on the Common Language Runtime, allowing programmers to write ASP.NET code using any Microsoft .NET language. ASP.NET is similar in nature to ASP, in that you must have an appropriate Windows-based operating system and web server to process ASP.NET pages. However, Halcyon Software, the developers of iASP, are currently working on a version of the .NET framework called iNET, for use on alternative web servers. This will ultimately lead to another commercial program, but does open up more possibilities for development. As the popularity of ASP.NET grows, it will undoubtedly become just as viable an option as ASP. As more hosting companies provide ASP.NET support via a shared server, the price of ASP.NET hosting will reduce dramatically (Downes, 2003) [15]. The ASP.NET directory structure can be determined by the developer's preferences.

Apart from a few reserved directory names, the site can span any number of directories. The structure is typically reflected directly in the URLs. Although ASP.NET provides means for intercepting the request at any point during processing, the developer is not forced to funnel requests through a central application or front controller. ASP.NET aims for performance benefits over other script-based technologies (including Classic ASP) by compiling the server-side code to one or more DLL files on the web server.

This compilation happens automatically the first time a page is requested (which means the developer need not perform a separate compilation step for pages).

This feature provides the ease of development offered by scripting languages with the performance benefits of a compiled binary. However, the compilation might cause a noticeable delay to the web user when the newly-edited page is first requested from the web server (Walther, 2007) [17]. The ASPX (an html file format used to create Web form pages) and other resource files are placed in a virtual host on an Internet Information Services server (or other compatible ASP.NET servers; see Other Implementations, below).

The first time a client requests a page, the .NET framework parses and compiles the file(s) into a .NET assembly and sends the response; subsequent requests are served from the DLL files. By default ASP.NET will compile the entire site in batches of 1000 files upon first request. If the compilation delay is causing problems, the batch size or the compilation strategy may be tweaked. Developers can also choose to pre-compile their code before deployment, eliminating the need for just-in-time compilation in a production environment (Walther, 2007) [17].

ASP.NET uses a visited composites rendering technique. During compilation the template (.aspx) file is compiled into initialization code which will build a control tree (the composite) representing the original (static) template. Literal text goes into instances of the Literal control class, server controls are represented by instances of a specific control class. The initialization code is combined with user-written code (usually by the assembly of multiple partial classes) and results in a class specific for the page. The page doubles as the root of the control tree (Walther, 2007) [17]. Actual requests for the page are processed through a number of steps. First, during the initialization steps, an instance of the page class is created and the initialization code is executed. This produces the initial control tree which is now typically manipulated by the methods of the page in the following steps. As each node in the tree is a control represented as an instance of a class, the code may change the tree structure as well as manipulate the properties/methods of the individual nodes. Finally, during the rendering step a visitor is used to visit every node in the tree, asking each node to render itself using the methods of the visitor. The resulting HTML code is sent to the client.

After the request has been processed, the instance of the page class is discarded and with it the entire control tree (Walther, 2007) [17]. Database driven website have used in various way on the internet to obtain information a, to do business and shop. (Dalton, 1997) [11], with (Seddon, 1998) [12] suggested approaches for the airline and construction industries towards solving typical computer problems.

Their suggestions can be generalized as awareness whereby senior management should be aware of the urgency of the problem and the associated financial and technical impacts like the absolute deadline.

Turban et al. (2000, p.20) [13] information technology (IT) in its narrow definition refers to the technological side of an information system (IS). It can be viewed as a sub system of information systems. Examples of information system application areas include:

- i. Flight reservation systems in air where a new system was designed so that schedulers can use “what if” scenarios to see if a new flight would be more profitable with a smaller or bigger airplane.
- ii. Computerized parts inventory systems like Seattle mariners using technology for profitability and inventory control.
- iii. In Internet applications like electronic mail, browsing and participating in newsgroups and chat rooms as noted by (James A, O’Brien, 2003, p.129) [14].
- iv. Online banking sites; where you can log in (by entering your user name and password) and check out your bank account balance. Your bank account information is stored in a database and has been connected to the web page with programming thus enabling you to see your banking information (Stefan, 2000) [9].
- v. E-Commerce; Database driven website is the key technology behind e-commerce. It has enabled businesses to be conduct through the internet and has made a variety of products to be easily available (just a click a way) to the customer.
- vi. Online publishing; many newsgroup centers such as nation media group and standard are making use of dynamic website to publish and archive their papers and magazines.

- vii. Computerized library system like squeezing out more efficiency for Dell Computer Corporation which is one of the world's largest manufacturers of personal computers.
- viii. Automated teller machines.

2.3 Planning and Development of Web Based Systems

2.3.1 Development Cycle

The research project follows the fundamental phases of the system development life cycle (SDLC) namely investigation and planning, analysis, design and implementation (Dalton, 1997) [11].

- i. In the investigation and planning phase, the researcher will identify the value of order processing system, conduct a feasibility analysis, and plan the project.
- ii. The analysis phase will involve the development of an analysis strategy, gather information, and building a set of analysis models.
- iii. In the design phase, the researcher will develop the physical, interface, program designs, and database (DB) and file specifications.
- iv. Finally, in the implementation phase, a prototype of the system will be constructed and tested to ensure it performs as designed.

One of the most common types of dynamic web pages is the database driven type. This means that you have a web page that grabs information from a database (the web page is connected to the database by programming,) and inserts that information into the web page each time it is loaded. If the information stored in the database changes, the web page connected to the database will also change accordingly and automatically that is without human intervention (Stefan, 2000) [9]. Using a database in conjunction with your web pages allows you to deliver text, images and other web content based on specific requests for information from people viewing your site. So rather than static pages which always deliver the same information, you can offer dynamically generated pages which could better serve your site's visitors (Gianni, 2002) [10]. In the database driven website all information is stored in table format of databases on the server.

Usually database driven websites have admin panel with user-friendly interface where you can manage the site content, i.e. add, edit, or delete information by yourself, create new pages or delete old ones. You can update your site easily each time you need it (Ramyalg, 2005) [8]. According to Gianni (2002) [10] three languages that are commonly used for interacting with databases on the Web are ColdFusion (a proprietary solution developed by Macromedia), Active Server Pages (also called ASP - a proprietary language developed by Microsoft) and PHP ((Hypertext Preprocessor) an open source solution not "owned" by any company). In project I have chosen to use PHP. PHP has a lot of advantages such as, it is fast, stable, secure (PHP offers many levels of security to prevent malicious attacks), easy to use, open source and it has connective abilities that it uses a modular system of extensions to interface with a variety of libraries such as graphics, XML, encryption. In addition, programmers can extend PHP by writing their own extensions and compiling them into the executable or they can create their own executable and load it using PHP dynamic loading mechanism plus it integrates well across many platforms and with various software programs.

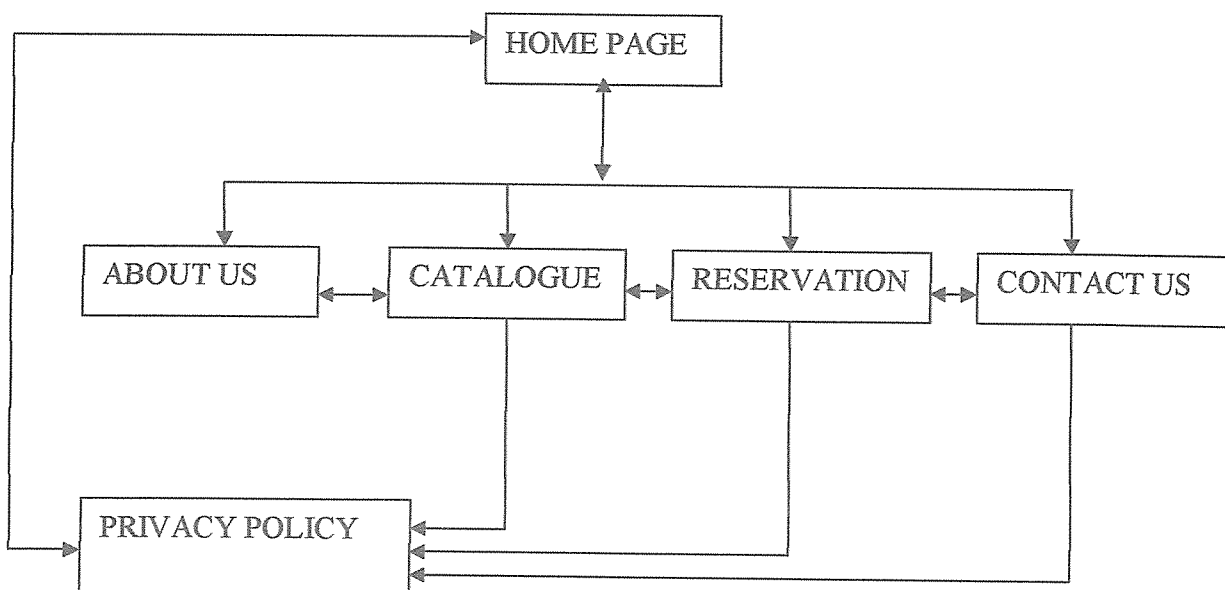


Figure 2.6: Layout Design for the Web Pages

In the development of the system database, the researcher used PHP and MySQL which is a database manipulation language for it supports relational databases, data storage in multiple tables, its efficient and available. In addition, MySQL can run on different platforms like Windows, UNIX among others. It is also embedded with SQL which enables querying of databases hence real time report generation.

2.3.2 XAMPP

XAMPP is a free, cross-platform web server, consisting mainly of the Apache HTTP Server, MySQL database, and interpreters for scripts written in the PHP and Perl programming languages. It also comes with a number of other modules, including OpenSSL and phpMyAdmin. XAMPP is released under the GNU General Public License and it is mainly used for web development projects (to develop dynamic websites).

XAMPP can be installed on computers with the following operating systems; Microsoft Windows, Linux, Sun Solaris and Mac OS X. XAMPP only requires one zip, tar or exe file (zip, tar and exe are types of file formats) to be downloaded and run, and very little configuration of the various components that make up the web server is required. Installing XAMPP takes less time than installing every installation contained in the package as a separate install. XAMPP provides support for creating and manipulating databases in MySQL and SQLite. XAMPP is a development tool that allows website designers and programmers to test their work on their own computers without any access to the Internet. It is sometimes used to actually serve web pages on the World Wide Web. A special tool is provided to password-protect the most important parts of the XAMPP package. For the project, the researcher used XAMPP to create the database for the Kenya Airways site and to test the site without uploading it on the Internet.

2.3.3 Macromedia Dreamweaver 8

Macromedia Dreamweaver 8 is software that enables rapid and easy development of web applications. Dreamweaver gives you the tools to fashion a wide variety of Web applications without ever looking at, much less enhancing the code.

Dreamweaver supports different technologies that enable the creation of a database-enabled web site. These are; Active Server Pages (ASP), Java Server Pages (JSP), Cold Fusion and Hypertext Preprocessor (PHP). The researcher used Macromedia Dreamweaver Version 8 together with PHP to develop the website which is database enabled. Macromedia Dreamweaver 8 is a full-featured web page design software application that enables you to create, edit, and publish web pages.

With Dreamweaver 8, managing multiple websites is a breeze, and improved interface features like collapsible code, page zoom, and a ruler make the program super versatile. A new feature in Dreamweaver 8 lets you zoom in on a page, up to 6,000+ percent. This is helpful when you need a much closer look at something, or want to make sure your design elements are matched up perfectly. The Dreamweaver 8 design also features rulers which will help you manage elements and place them in the perfect position, without error. The webpage rendering engine is also improved, and will show you a more realistic version of how the page will actually be displayed in web browsers. This helps with design differences between Internet Explore (IE) and Mozilla Firefox, as each browser displays some elements a little differently.

Dreamweaver System Requirements are as follows; in order to run Dreamweaver 8 on the Windows platform, you need the following minimum system: 800 MHz Intel Pentium processor or equivalent, Windows 2000 or Windows XP, 256 MB RAM (1GB recommended to run Studio products), 1024 x 768, 16-bit display (Macromedia recommends 32-bit), 650 MB available disk space (More if you plan on storing your Web pages on the hard drive). Macintosh Dreamweaver System Requirements are as follows; in order to run Dreamweaver 8 on a Macintosh, you need the following minimum system: 600 MHz Power PC G3 processor or later, Mac OS X 10.3 and later, 256 MB computer RAM (1GB recommended to run Studio products), 1024 x 768, thousands of colors display (Macromedia recommends millions of colors), 300 MB available disk space (More if you plan on storing your Web pages on the hard drive). Downes, 2003 [15] states that there are two major technologies that are used in database driven websites which are:

a. Client-side scripting

Here the pages rely on the ability of the website visitor's own computer to manifest their effects. The web pages are sent via the web server to the visitor's web browser, and once loaded the scripts begin to work.

Because the code that generates these interactions is sent directly to the browser it is open to public scrutiny, and is therefore not the best choice in all situations. A user authentication system, for example, becomes ineffective because the code that handles the login process is completely visible, and therefore insecure. An example is a dynamic page created with JavaScript.

b. Server-side scripting

Here the code is "pre-processed" by an interpreter, and the results of this processing are passed back to the web server and on to the visitor's web browser. As only the results of the processing are sent to the browser, the code that generated them remains hidden, and is therefore much more secure.

2.4 Components of Web Based Systems

2.4.1 MySQL

MySQL is an SQL (Standard Query Language) Database Server. A server is a piece of software that runs in the background on a computer. It is built to keep running and executing specific tasks (Downes, 2003) [15]. The advantages of MySQL include; MySQL is very fast that is efficient fairly well featured, MySQL is can handle large amounts of data, MySQL is very inexpensive to free, and it's easy to learn and quick to implement. The Disadvantages of MySQL include; MySQL cannot handle complex relational issues such as subselects (subqueries), transactions and commit/rollback not directly supported, commercial sites with online ordering capabilities will find this lack of transaction support disappointing, MySQL has a weak support for foreign keys and referential integrity and lastly MySQL has no stored procedures and triggers.

2.4.2 Microsoft Access

Microsoft Access is a relational database management system from Microsoft that combines the relational Microsoft Jet Database Engine with a graphical user interface and software development tools. Access can use data stored in Access/Jet, Microsoft SQL Server, Oracle, or any ODBC-compliant data container (including MySQL and PostgreSQL). Skilled software developers and data architects use it to develop application software. Relatively unskilled programmers and non-programmer "power users" can use it to build simple applications. It supports some object-oriented techniques but falls short of being a fully object-oriented development tool. The advantages of Microsoft Access include; Microsoft Access being very inexpensive to free, Microsoft

Access is easy to learn and it's also quick to implement while the disadvantages include it being limited in what it can do with respect to real database manipulations that is those contained in SQL Microsoft Access does not handle large amounts of data well and it is less reliable.

2.4.3 SQL Server

Microsoft SQL Server is a relational database management system (RDBMS) produced by Microsoft. Its primary query language is Transact-SQL, an implementation of the ANSI/ISO standard Structured Query Language (SQL) used by both Microsoft and Sybase. The architecture of Microsoft SQL Server is broadly divided into three components: SQLOS which implements the basic services required by SQL Server, including thread scheduling, memory management and I/O management; the Relational Engine, which implements the relational database components including support for databases, tables, queries and stored procedures as well as implementing the type system; and the Protocol Layer which exposes the SQL Server functionality (Delaney, 2007) [23].

SQL Server also includes an assortment of add-on services. While these are not essential for the operation for the database system, these provide value added services on top of the core database management system.

These services either run as a part of some SQL Server component or out-of-process as Windows Service and presents their own API to control and interact with them (Delaney, 2007) [23]. The advantages of SQL Server is that it supports very large databases and its support for triggers, stored procedures, and other database analysis tools while the disadvantages of SQL Server is that it is expensive and relatively difficult to learn.

2.4.4 Database

A database is a single organized collection of structured data, stored with minimum duplication of data items so as to provide a consistent and controlled pool of data which is common to all users of the system and is independent of programs that use that data. Independency of the database and programs using it means that one can be changed without changing the other (French, 1992) [4].

A database relies upon software to organize the storage of the data and to enable a person or program to extract desired information. The term "database" refers to the collection of related records and the software is database management system (DBMS); this is sometimes shortened to database manager or database system. According to Jewett (2008) [5] the earliest known use of the term data base was in November 1963, when the System Development Corporation sponsored a symposium under the title Development and Management of a Computer-centered Data Base. Database as a single word became common in Europe in the early 1970s and by the end of the decade it was being used in major American newspapers. The abbreviation DB, however, survived.

The first database management systems were developed in the 1960s. A pioneer in the field was Charles Bachman. Bachman's early papers show that his aim was to make more effective use of the new direct access storage devices becoming available: until then, data processing had been based on punched cards and magnetic tape, so that serial processing was the dominant activity. Two key data models arose at this time: CODASYL developed the network model based on Bachman's ideas, and independently the hierarchical model was used in a system developed by North American Rockwell later adopted by IBM as the cornerstone of their IMS product.

While IMS along with the CODASYL IDMS were the big, high visibility databases developed in the 1960s, several others were also born in that decade, some of which have a significant installed base today (Jewett, 2008) [5]. To satisfy the requirements of functionality, flexibility, and usability in a large sites which changes frequently one should create a central repository for the creation, retrieval, and searching of test cases (database). Therefore very large sites and sites which change frequently are best managed if they run off a database (Chen, P. P., Zvieli, 1986[24]).

CHAPTER THREE

METHODOLOGY

This chapter gives a description of how the information for airline reservation was developed. The methodology used was adopted from several existing methodologies for different applications especially in the field of knowledge engineering, computer science and software engineering (Mohd, 2005) [25]. This web based system for airline reservation is a result of integration of these techniques.

The methodology used was according to the following subsections knowledge acquisition and representation, design, implementation and validation. However, it was noted that the development process was iterative, though it was depicted as comparing of phases. The figure below shows the system's development methodology.

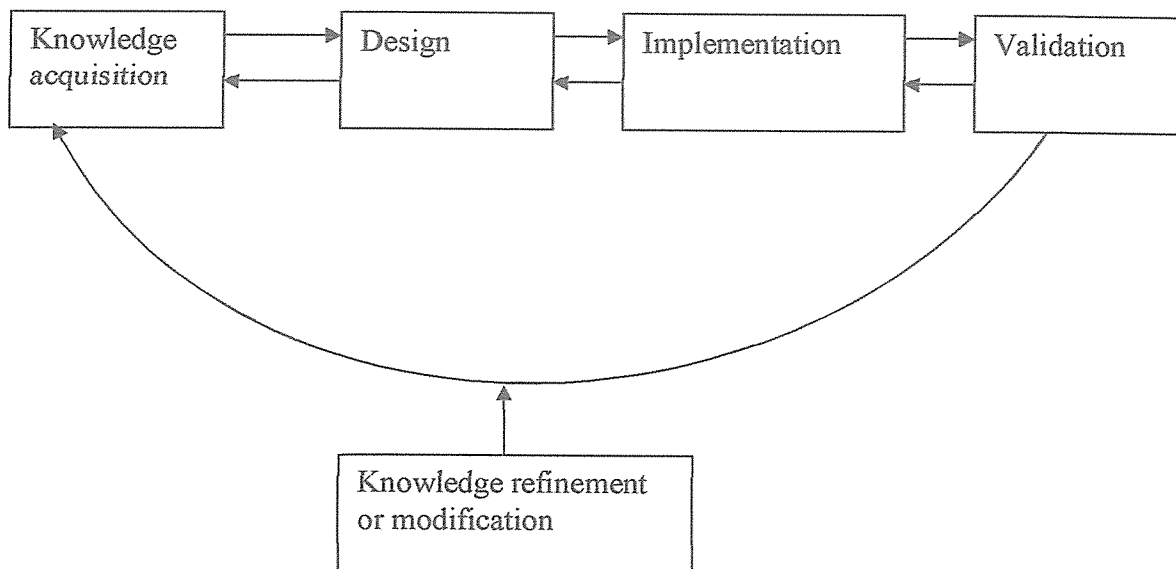


Figure 3.0: An illustration of a systems' development methodology for airline reservation

3.1 Data Collection

The information relevant to the project objectives was collected using personal communication that is interviews and observation, questionnaire and secondary documents like manuals and Internet for reading references by academic authorities as discussed below.

3.1.1 Documentary Review

This is the review of previous study and existing literature and documentations. This has helped the researcher to get a background of the problem which provided a good starting point and helped in obtaining the requirements for the system. By reviewing some of the documents, the researcher was familiar with the operations that are performed and the formed relations within the organization. The main advantage of this approach is that it takes less time and it is less expensive as compared to other forms of fact finding because it is reliable to the existing literature. Background reading was used for the following reasons:

- i. It is an inexpensive way of acquiring information
- ii. It is easily available depending on the document one needs.
- iii. It provided opportunity for study of trends of events over time.
- iv. Data collected was highly qualified, valid, available and reliable.
- v. It was used to find out the historical trends of events that seem impossible with other data collection methods.
- vi. It is a strong base to build a question such as a reference.

3.1.2 Observation

Observation involves recording the behavioral patterns of people, objects and events in a systematic manner. Observational methods may be: structured or unstructured, disguised or undisguised, natural or contrived, personal, mechanical, non-participant, participant. The researcher used the natural and participant method because as a Kenyan citizen living in Kenya was able to survey what goes around and to naturally observe the day to day activities in the country concerning the airline passengers.

This helped to better understand what the problem really is, its scope and to obtain the best solution to the problem. There are 2 types of observation techniques the researcher used:

- i. Work sampling: This method involves a large number of observations taken at random intervals. The observation period is not continuous the researcher needs to predefined the operations of the job to be observed.
- ii. Living the system: In this type of observation the analyst actively performs the role of the user for a short time.

This was one of the most effective ways to learn the problems and the requirements that the new system should cater for. The researcher actively booked a ticket and observed the time that it took for the reservation to be made.

Observation was used due to the following reasons:

- i. The information gathered was reliable. This is because most of the time the method was used to check the validity of data obtained from others.
- ii. Observation is relatively cheaper compared to other techniques.
- iii. The analyst was able to observe how the system they were using operated.

3.1.3 Questionnaire

The researcher also carried out questionnaires, this is a document containing standard question that are to be answered by a large group of people in order to get information from them. The researcher printed out the questionnaires and distributed them randomly to the beneficiaries because MIA covers a large geographic area in Eastern Kenya to answer and also the staff members who were in the field. The sample questions are in appendix A. The reasons for choosing questionnaire method are that:

- i. The question were easy to answer because they were short & simple
- ii. It saved time since the group was large, especially in Nairobi where the population is large.
- iii. It ensured privacy & confidentiality.
- iv. Data which was gathered was easy to analyze.

The researchers using this information gathered embarked on developing the new system to the expectation of the staff of MIA at the same time the passengers because they were to benefit from it. The system was designed according to the requirement and specification gathered.

3.1.4 Interview

Interviewing is a technique that is primarily used to gain an understanding of the underlying reasons and motivations for people's attitudes, preferences or behavior. It is a technique where the interviewer interacts face to face with the interviewee. Interviews can be undertaken on a personal one-to-one basis or in a group.

The researcher conducted both one-on-one and group interviews in Kisumu, Eldoret, Mombasa and Nairobi where the Kenya Airways has its airports in common. Although the approach is expensive and time consuming, the researcher was able to get a one-on-one chance with the passengers thus was able to better understand the problem, its scope, and to elicit the requirements of the passengers and the staff in general. Through the interviews, the researcher was also able to obtain views from the people on how they believed the problem could be solved which helped the researcher to come up with this project.

3.2 System Design

It defines the structure and contents of the new system and the Specification of how the system was implemented.

3.2.1 User Interface Design

User interface design was the specification of a dialogue or conversation between the system user and the computer. It resulted in data input and information output. The input design permits the users to display or make a data entry into the designed forms by the researcher passengers' details file, reservation file among others as shown in the created software. The output designs shows the reports produced after data entry. The output is what is produced by the system after data entry has been done. This output is in form of reports and can be printed as hardcopy.

The softcopy is displayed on the screen as required by the user. Examples of reports include passengers' details report and reservation update report.

3.2.2 Database Design

This is the process of translating logical data models into physical database schemas. A database is a collection of tables with logical pointers that relate records in one table to records in a different table. Database architecture is build around a database management system that provides the technology to define the database structure and then to crease, read, update and delete records in the tables that make up the structure. A DBMS provides a data language to accomplish this which provides at least two components.

For example the data definition language to create and maintain the database structure and rules and the data manipulation language to create, read, use, update and delete records in the database.

3.2.3 The Data Integration Framework

The fundamental challenge was to enable data to be extracted from multiple sources, to be integrated dynamically into a single application. To answer this seemingly simple request turned out to be very complex. There was need to incorporate data about airline reservation for the customers' from various PHP flat files with login data and general information about information about the customers' stored in the database all depending on the request for the customer's reservation details. The data from the database was integrated in HTML format with the data from the flat files and presented to the user who uses a web browser to access the system as shown in the figure below.

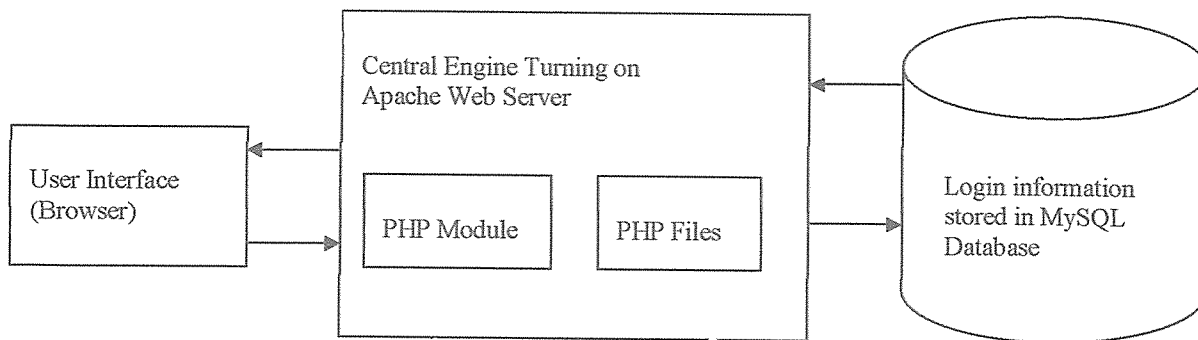


Figure 3.2.3: An illustration of a block diagram of system operation

3.3 System Implementation

This is a stage of software development which involves turning the requirements into technological terms. Implementation includes conducting of system tests, conversion from the old system to the new system, installation of databases and training of users. The interfaces were developed by the use of Macromedia Dreamweaver 8 while the database was created by use of XAMPP software. In order to perfect the functionality of the system, the researcher ensured every process flow of reservations has its own entity and attributes corresponding to the user friendly interfaces.

3.4 Coding

This was the actual programming by the researcher which involved writing codes by use of PHP programming language.

3.5 System Testing and Validation

Testing is the process of checking whether the system is working as per required and it provides the desired results. The researcher tested both the database and the website.

In the database the researcher tested each table individually by validating the field size, checking the constraints and testing the stored procedures. In the website the researcher tested to see if users can perform the required task using the website by performing some of the necessary tasks using the site. The researcher tested each and every page individually before testing the entire website. After testing the database and the website individually, the researcher tested the connectivity between the website and the database by trying to retrieve information from the database and by entering data through the forms on the website to see if the information can be stored on the database. This enabled the researcher to achieve the objective which was to test and validate the designed website.

System validation is an attempt to build into the computer program the power to detect whether entries made are correct. This was done using data from the real world environment.

Audit testing was also done to ensure the system is free of errors and ready to be placed into operation. The incorrect data items are detected and reported. The validation checks used in this project include:

- i. Range Check, the fields are checked to insure that they contain the correct number of characters.
- ii. Format Check, this check ensured that fields are enabled using the correct data types that is a nonnumeric field should not have numeric data and vice versa.

CHAPTER FOUR

SYSTEM ANALYSIS AND DESIGN

Descriptions of both the current system are given in this chapter together with the tools used to design the proposed system. System analysis is a problem-solving technique that decomposes a system into its component pieces for the purpose of studying how well those component parts work and interact to accomplish their purpose. System analysis was driven by the business concerns of system owners and system users. It addresses the data, processes and interface building blocks from system owners and systems user's perspectives. It was part of preliminary investigation of the previous system. A detailed study of the previous system was done in order to find out:

- i. The difficulties and problems of current system.
- ii. The user's requirements.
- iii. The inputs to the system
- iv. The outputs generated.

4.1 Investigation of the Current System

An investigation was carried out to establish how the existing system functions and what its problems were. This led to a definition of a set of options from which the users could choose their required system. In carrying out an investigation, information about the current system was collected and by recording the problems and requirements described by the users of the current system, a picture of the required system was built.

Several major problems were identified with the current system as stated below:

It was difficult for the counter clerk to keep track of the ticket reservation transactions due to the large amount of paperwork involved. This was proved by the long period of time it took to come up with the reports that showed the different types of passengers who were waiting for their reservations for the tickets. Because of the vast amount of paperwork, the reports produced were mostly inconsistent and inaccurate due to human error.

Response times had doubled. It therefore took a lot of time to carry out a single transaction which led to long queues of reservations waiting to be serviced. This led to customer relations problems.

The working environment was not conducive for the employees in the front office department due to the many file cabinets that occupied a lot of space causing congestion. Paper documents could be easily destroyed or misplaced and in most cases there was normally no backup for the documents which meant that there was risk for loss of important information. It took time to come up with urgently needed information due to the type of filing system that was being used. This led to unnecessary delays.

4.2 Description of the Proposed System

After analyzing the current system and in order to improve the current situations caused by the current system, considerations should be made to implement a new automated ticket booking system. This will increase the company's profits by improving the quality of service, utilization of resources including people, faster access to management information and reduction of expenditure and much time consumed for identifying the different types of transactions manually.

The researcher assumes that the domain analysis is complete and development and design (DAD) is ready for the reference. The analyst also has a fair knowledge of the system and the system environment. For the sake of convenience and to reduce complexity at this stage in object oriented analysis (OOA), the researcher is ignoring certain conditions, constraints and features that the real system may have. For example, the researcher is not considering cancellation of tickets as a requirement. The researcher assumes that all passengers irrespective of their type (senior citizen, military personnel, and special category passengers) are the same, and therefore are charged with the same fare. The researcher now begins the requirement of the proposed system as stated below.

4.3 Requirements of the Proposed System

After data collection and analyzing, the researcher came up with the requirements of the new system. The system, after completion was expected to:

- i. Expedite the processing of orders through improved data capture technology.
- ii. Reduce response time by attending to orders as soon as they come in.
- iii. Secure and ensure that only authorized people gain access to the system.
- iv. Keep all records on all clients both presently and in future.
- v. Update, edit or amend records of the passengers tickets already reserved.

4.3.1 System Requirement

- i. The passenger has a prior knowledge of the reservation and ticketing system. The passenger arrives at the airline ticket counter and interacts with the counter clerk first through an enquiry and then follows the process of form filling, tendering, payment and collecting the tickets.
- ii. Passenger accepts the ticket or leaves the counter.
- iii. Passenger seeks information on fare, airplane timings and availability of tickets.
- iv. Passenger can have multiple tickets or a single ticket.
- v. Airplanes are identified by name and number.
- vi. No receipt is issued for money transacted.
- vii. Output of the system is ticket(s) with details.
- viii. The process is triggered through a form filled by the passenger detailing the requirements of tickets, airplane and date.

4.3.2 User Requirements

- i. The system shall be easy to use; hence the graphical user interface needs to be suitable.
- ii. The system should be interactive, in case user interacts with the system, it should be able to provide real time feedback hence generating timely reports.

- iii. The system should be able to provide information to the different users depending on their levels and authority or responsibility in the organization.

4.3.3 Non Functional Requirements

These are the requirements which define system properties and constraints that enable efficiency in the system.

- i. Availability: The system should be able to provide necessary information to the authorized users.
- ii. Reliability: Access needs to be easy but secure and therefore it should be a reliable backbone of the company information.
- iii. Functionality: The system should be able to perform its functions as expected that is to say that it should be secure.
- iv. Usability: In case of use, the system should be simple, interactive and easy.
- v. Scalability: The system should be flexible and can easily adapt to change incase need arises.
- vi. Performance: The system should be able to respond to user requirements within the expected time and also create necessary reports.

4.3.4 Specific Requirements

a. Server Machine Software

- i. Windows operating system.
- ii. Web server with hypertext markup language.
- iii. Antivirus software and graphical browser installed.
- iv. MySQL server.

b. Server Hardware

- i. Pentium 3 and above.
- ii. Random access memory (RAM) of 512 megabytes (MB) and above.
- iii. Hard disk space of 3 gigabytes (GB).
- iv. Internet connection.

c. Client Machine

- i. Web browser software.
- ii. Internet connection.

d. Security

- i. Password usage on the system.
- ii. System audits should be carried out.

4.3.5 Benefits of the Proposed System

The proposed system led to the following benefits to the counter clerk officer and Kenya Airways in general:

- i. Save on the amount of storage space due to the introduced electronic filing.
- ii. Easy backups for cases where records are damaged.
- iii. Better methods of records retrieval through special search program that it provides.
- iv. Easy report generation.
- v. Utilization of human resources and time.
- vi. Proper maintenance of records for future reference.
- vii. The system is not prone to manipulations and alterations.

4.4 Current System Processes

4.4.1 Context Diagram for the Current system's Processes

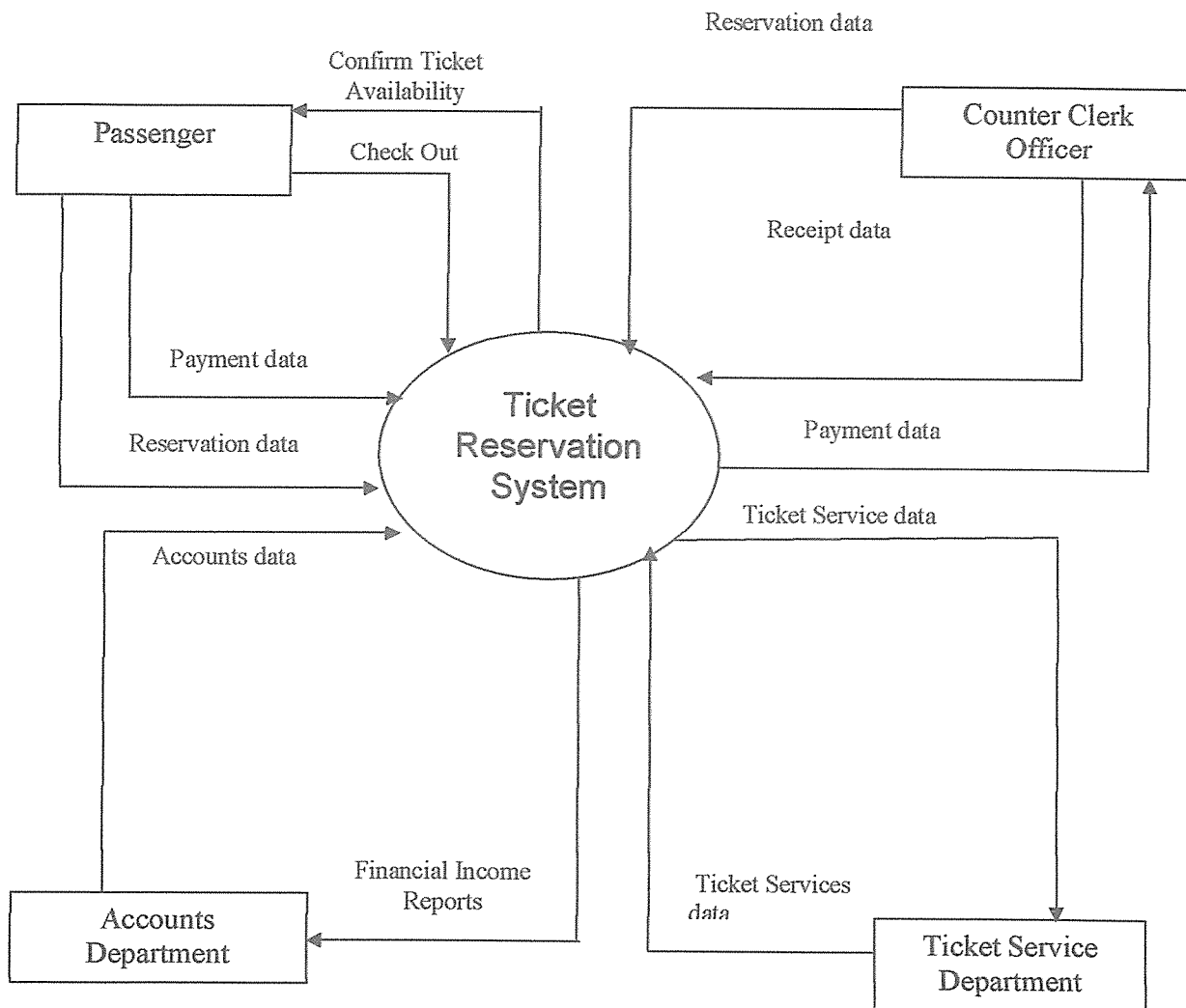


Figure 4.4.1: Context Diagram for the Current System Process

4.5 System Design

Having obtained the new system requirements, the next task was to design a system that meets the specified user requirements. System design involved the specification and design of a technical, computer based solution for the business requirements identified during system analysis.

It was driven by the technical concerns of the system designer. System design was looked at from three perspectives:

- i. Logical Design
- ii. Physical Design
- iii. Database Design

4.5.1 Logical Design

Logical design was used to model what the system must do but not how the system will be implemented. It shows the inputs, outputs and processes to be carried out along with the controls over the activities. It involved defining:

- i. How the system will appear to the users.
- ii. *The data and the file structure for the entire new system designed.*
- iii. The development of the output format.
- iv. The part of information system in terms of their relationship to each other.

4.5.1.1 Use Case Diagram for Reservation Ticket Booking System

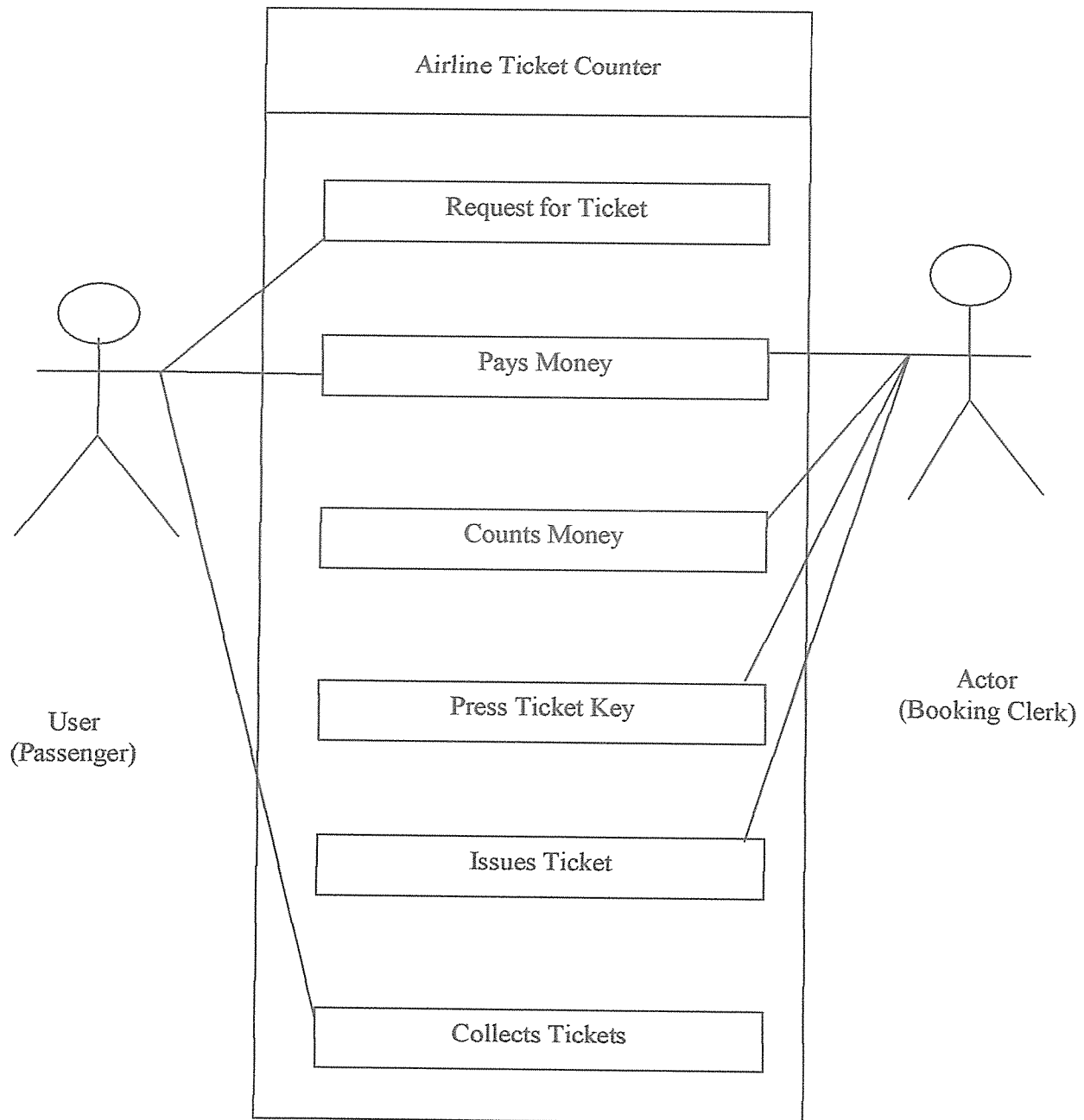


Figure 4.5.1.1: Use Case Diagram for the Ticket Reservation System

4.5.1.2 Flow Chart for the Ticket Reservation System

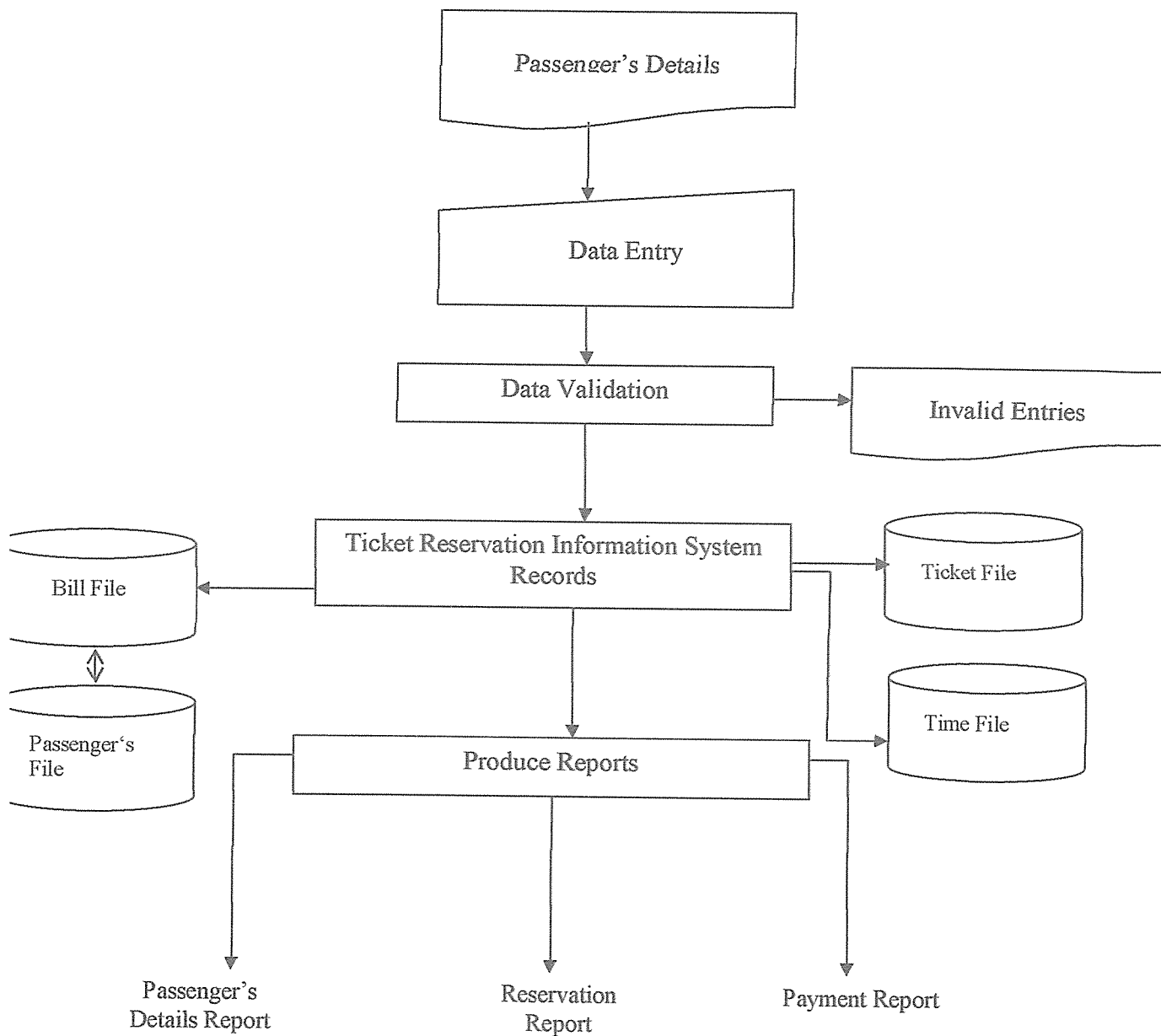


Figure 4.5.1.2: System Flow Chart for the Reservation Ticket System

4.6 Program Specification

Based on the system observation by the analyst, a high level activity diagram is drawn modeling the process of reservation and issue of tickets to the passenger.

The activity diagram brings everybody concerned with the system on the ground to a common understanding of the system as it functions. The researcher used this activity diagram of the process to develop use cases, which together achieves the goal of issuing the ticket.

4.6.1 Activity Diagram of Reservation and Issue of Tickets.

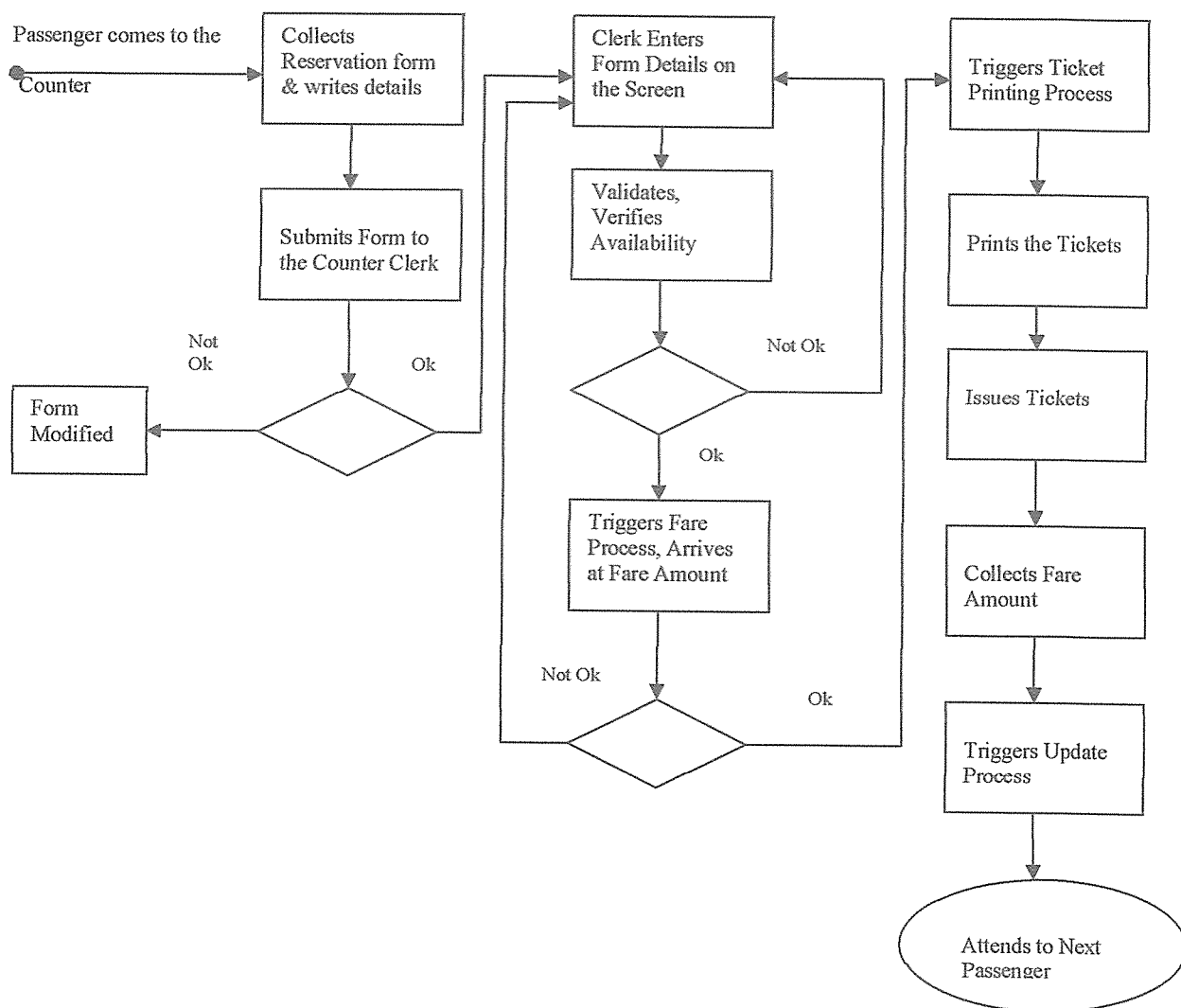


Figure 4.6.1: Activity Diagram of Reservation and Issue of Tickets

In the ticket reservation system, users are the passenger, counter clerk and the Reservation and the Ticketing System (R&T System). Take each user and identify the roles played, which leads the researcher to identify the use case. The table below shows the result of the process of identifying the use cases.

The system has three users, eight roles and eleven use cases. To illustrate the process of identifying the use case, let the passenger (a user of the system). A passenger as a user may play one or more of three roles. The roles are:

- i. Enquiring about the availability of tickets on particular dates to a destination and the fare per ticket.

The role is enquiring

- ii. Reserving the ticket(s) on a particular airline on particular date for a destination by requisitioning through a reservation form.

The role is reservation.

- iii. Cancellation the tickets after issuing and payment.

The role is cancellation.

As explained in the case of passenger, the roles are use cases. Similarly, one can probe into the roles and use cases for counter clerk and reservation and ticketing.

User	Role	Use Case
1. Passenger	<ul style="list-style-type: none"> — Enquiry. — Reservation and ticketing. — Cancellation. 	<ul style="list-style-type: none"> — Enquiry ticket availability and other details. — Reserve seat and berths, tickets. — Cancel tickets.
2. Counter Clerk	<ul style="list-style-type: none"> — Form data entry. — Requisition processor. — Ticket processor. — Data manager. 	<ul style="list-style-type: none"> — Enter Reservation Requisition Form. — Process requisition for booking. — Process ticket to print. — Submits ticket data for updating.
3. Reservation and Ticketing System	<ul style="list-style-type: none"> — System server. 	<ul style="list-style-type: none"> — Process reservation data, process ticketing process cancellation. — Update the status by date and airline.

Table 4.6.1: Users-Roles-Use cases

4.6.2 Activity Diagram of Use Case Inquiry to Issue of Tickets

Interaction diagrams are used to show the interactions between user or actor and the system. Use case is a scenario that develops through interaction. The following are the different scenarios through the interacting on diagrams:

- Use Case: Enquiring on ticket availability
 - i. Passenger submits information about the date and airline, and requirement of tickets to the counter clerk.
 - ii. Clerk checks the availability of tickets on the date and airline.

- iii. Communicates the availability status to the passenger.
- iv. If not OK, the passenger proceeds to book the ticket through a requisition form.
- v. If not OK, the passenger changes the date or airline and requests availability.
- vi. Steps (ii) to (iv) are repeated.

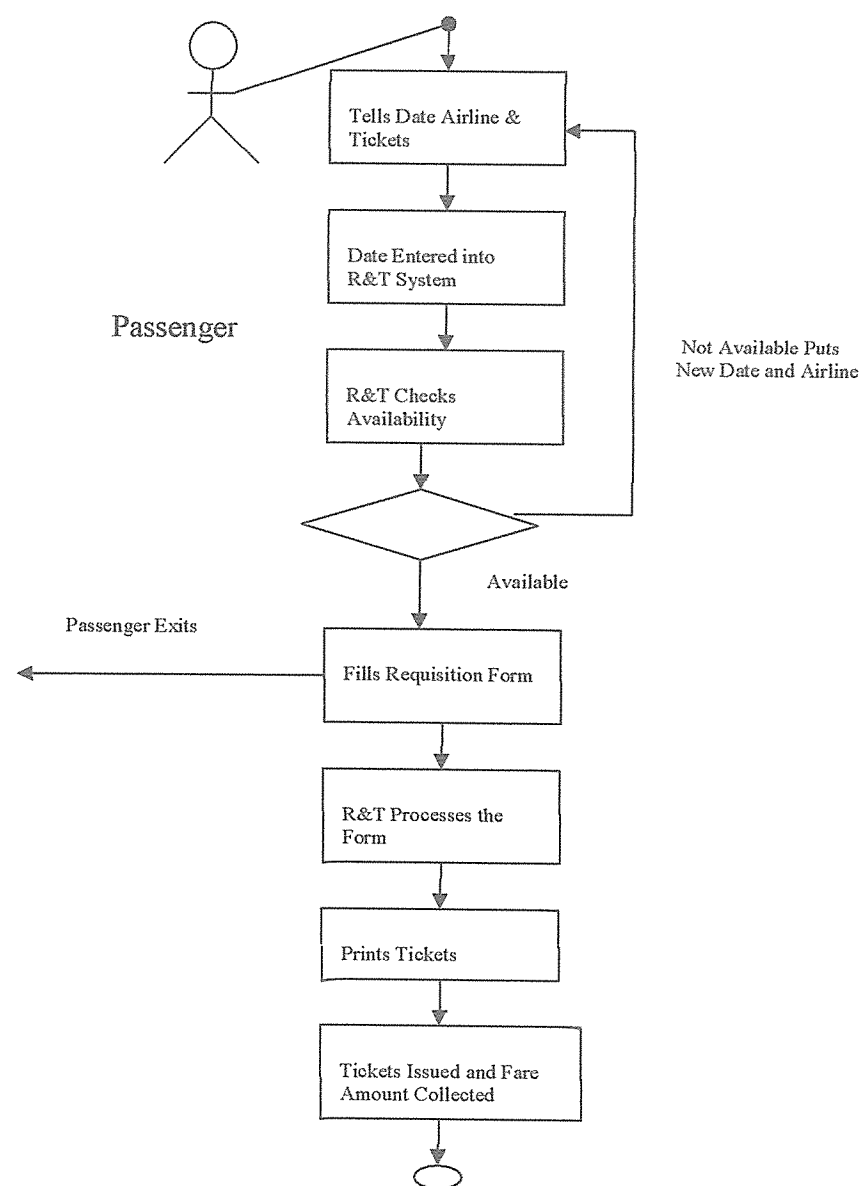


Figure 4.6.2.1: Activity Diagram of Use Case Inquiry to Issue of Tickets

The process of database design generally consists of a number of steps which are not all necessary in all cases. To create the database, the researcher started by determining the data to be stored in the database. Then the researcher identified the relationships between the different data elements by looking out for the dependencies in the data that is where one piece of information is dependent upon another (when one piece of information changes, the other will also change). Using xampp-win32-1.5.0 (Apache Friends), the researcher created a database called airline together with its tables. Below shows the interface that enabled me to create the tables;

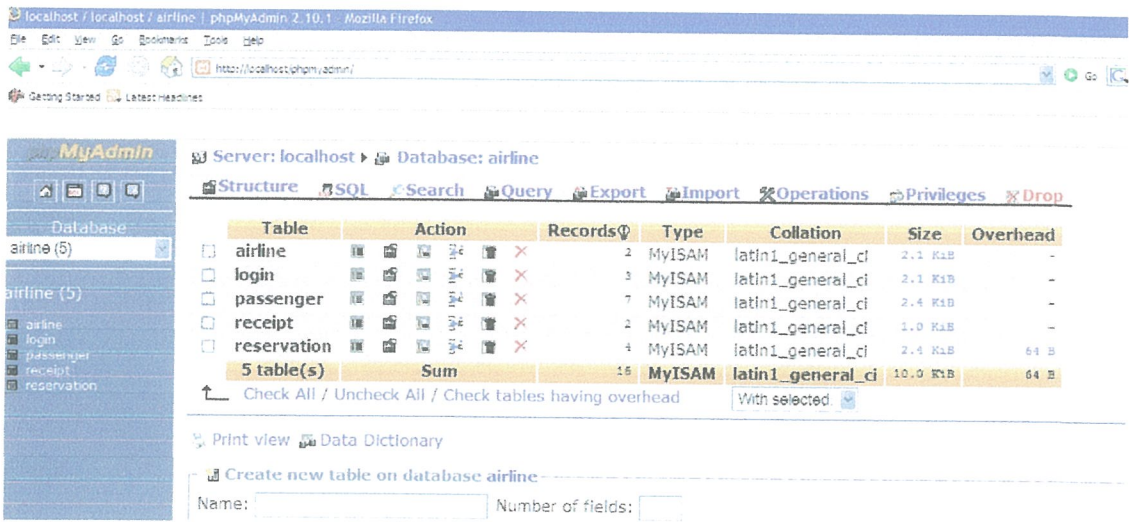


Figure 4.7: Interface for creating the tables'

The database contains five tables which can only be modified by the counter clerk. This enabled the researcher to achieve her objective which was to carry out system study and to design a database for efficient data storage. Hence, the researcher has dealt with preliminary design then a detailed design of the system. This is where the preliminary design has used the Computer Aided Software Engineering (CASE) tools and detailed design which defines the requirements for input, processing, storage and output as well as system control and backup. This included the Conceptual, Logical and Physical modeling of the database.

CHAPTER FIVE

SYSTEM IMPLEMENTATION

This chapter explains the results achieved after the implementation of the system analysis and design as discussed in chapter four. The researcher has discussed the tools that were used in developing the system, the test data used for testing and how the system works. System development involved converting design specifications from the design phase into executable programs. It involved actual programming which involved writing codes using Hypertext Preprocessor (PHP) programming language.

5.1 Website Design

Web design is a process of conceptualization, planning, modeling, and execution of electronic media content delivered via Internet in the form of technologies. Examples of such technologies are markup languages suitable for interpretation and display by a web browser or other web-based graphical user interfaces (GUIs). The intent of web design is to create a web site (a collection of electronic files residing on one or more web servers) that presents content (including interactive features or interfaces) to the end user in the form of web pages once requested. Designing a web site is defined as the arrangement and creation of web pages that in turn make up a web site. A web page consists of information for which the web site is developed.

The basic aspects of design are: The content, that is, the substance and information on the site should be relevant to the site and should target the area of the public that the website is concerned with, usability that is the site should be user-friendly, with the interface and navigation simple and reliable, appearance that is the graphics and text should include a single style that flows throughout, to show consistency and visibility that is the site must also be easy to find via most, if not all, major search engines and advertisement media.

Using Macromedia Dreamweaver version 8 to develop the website, the researcher started by defining a site for the project. A site definition is the essential starting point to the development of any website. With the information that is provided in the site definition, Dreamweaver will be able to accurately track the directory structure of the site, keep a watchful eye over all the internal and external hyperlinks, maintain a list of all images, movies, colors, and external files and to provide one with a preview of what the page will probably look like without even uploading it to the server. Without the information provided in the site definition, Dreamweaver will have no idea where to begin looking. Dreamweaver employs two methods for defining sites that is the Basic method, which employs a step-by-step Wizard that guides you through the Site Definition process, and there is an Advanced Definition process, which gives you complete control over every aspect of your site's definition. To define the projects site, the researcher used the Advanced Definition process as shown below:

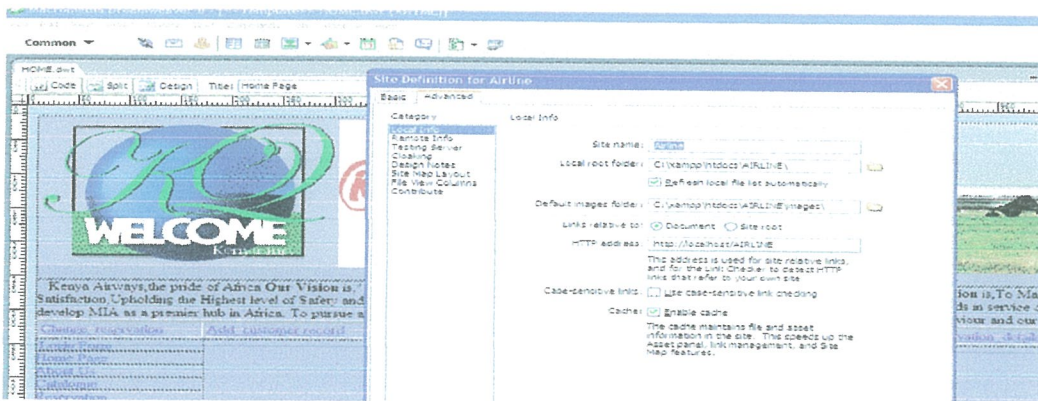


Figure 5.1.1: Site definition

After the site definition the researcher started working on the pages. For flexibility purposes the researcher created a template using Macromedia's Fireworks MX because it is designed to integrate with Dreamweaver, and it allows one to easily break an image up into its component parts (slicing). This allows one to easily create a table-based layout, as Fireworks exports all the necessary HTML for use within Dreamweaver.

The researcher decided to use a template to achieve flexibility because, if one created a number of web pages based on a template within Dreamweaver and the client decided to change the size and color of the logo, one would only have to change one file, the template. Once it has been changed, Dreamweaver automatically adjusts all the pages that have been created using that template. The template is shown below:

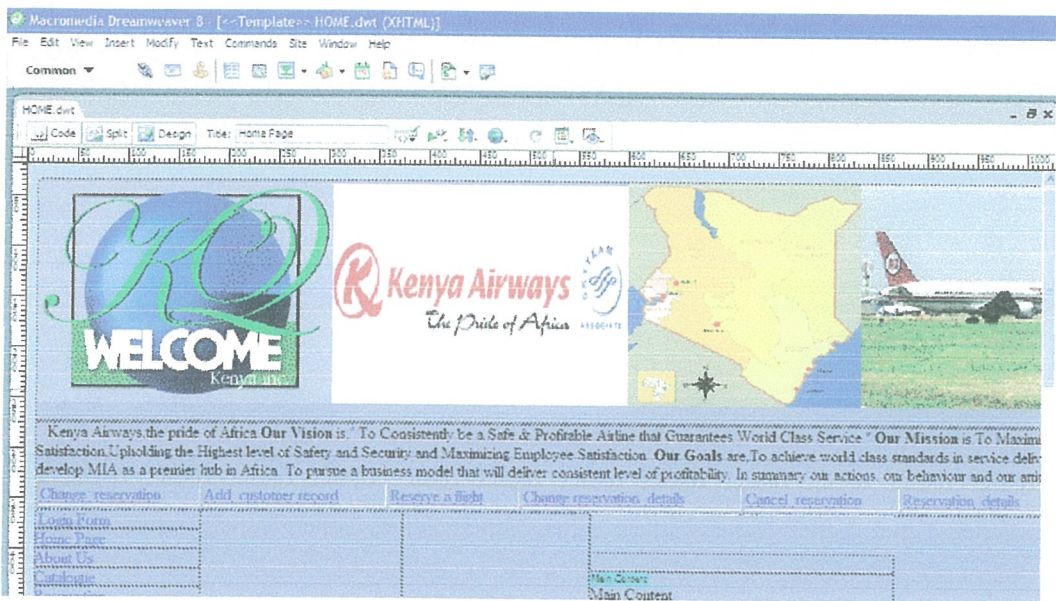


Figure 5.1.2: Template Interface

The above interface enables the users of the system to interact with it through the defined links by the researcher. The links enable the users of the system to navigate through the entire system for more information concerning the particular users. Like the login form which enables the authorized users to enter into the system to access information.

After the site definition, the researcher connected the site to the database (airline database). The researcher created the connection using MySQL connection interface as shown below;

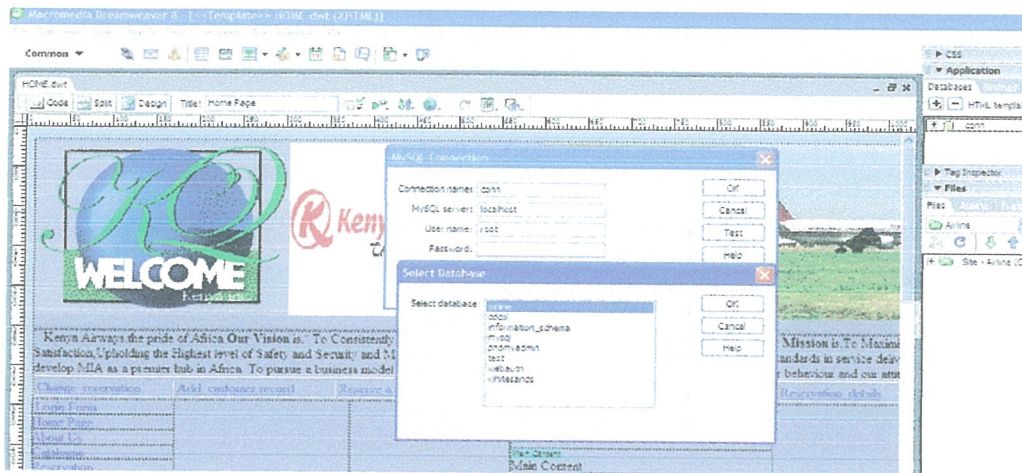


Figure 5.1.3: Interface for creating the database connection

The above interface enables the researcher to link the information from Dream weaver Macromedia 8 to the airline database created using XAMPP software to enable information transfer from the customers' into the system to enable the counter clerk to update their data and send them feedback. This was done by defining the connection name, MySQL server and the user name.

Lastly, the researcher created the pages and applied the template to all the pages as shown in the software created for approval. This enabled the researcher to achieve the objective which was to design a database driven website for MIA.

5.2 User Interface Design

The user interfaces are made up of forms and hyperlinks which enable an authorized user to link to other pages, they are created using Hypertext Markup Language (HTML), Cascading Style (CSS) and PHP.

5.2.1 Typical Functions of the User Interface

- i. To enable an authorized user to log into the system.
- ii. Enable the users to view the available data and use it making strategic decisions.

- iii. Give the counter clerk the ability to view reports to easily reserve tickets and cancel the basing on the passengers details status.
- iv. Enable storage and safety of the sensitive data entered into the system.
- v. Provide the authorized users with the ability of querying forms.
- vi. To enhance the ability to retrieve data from the database for easy interpretation.

Figure 5.2.1: Login Form for the Airline authorized users'

This interface solves the problem of security on the system. For anyone to access any information in the system they have to use the authorized detail which is the use name and password while logging in. this is the interface which enables the counter clerk to enter into the system and update the customers' details and send them feedback.

5.3 Interactions Between the User Interface and the Database

The system allows the use to interact with the database through the use of MySQL/PHP connections and other scripting languages like Java script and PHP programming language. The user can navigate to the other pages from the home web page by use of links provided on the page and to other pages by use of links provided on each page.

The information about the reservation details from each customer is stored in the database. The database has five (5) tables which are; the passenger table which store necessary information about the passenger, the reservation table which contains all the reservations that have been undertaken by the customer, receipt table which contains the details about the passenger and the airline, the airline table which contains all the information about the type of flight and finally the login form for the authorized users to access the database information for manipulation. The passenger and reservation table can only be manipulated by a member from the MIA Kenya Airways, while the information stored on the passenger table is strictly from the customer. The interaction between the interface and the database was achieved through the web browser. The example below shows how the user interacts with the system through the web browser.

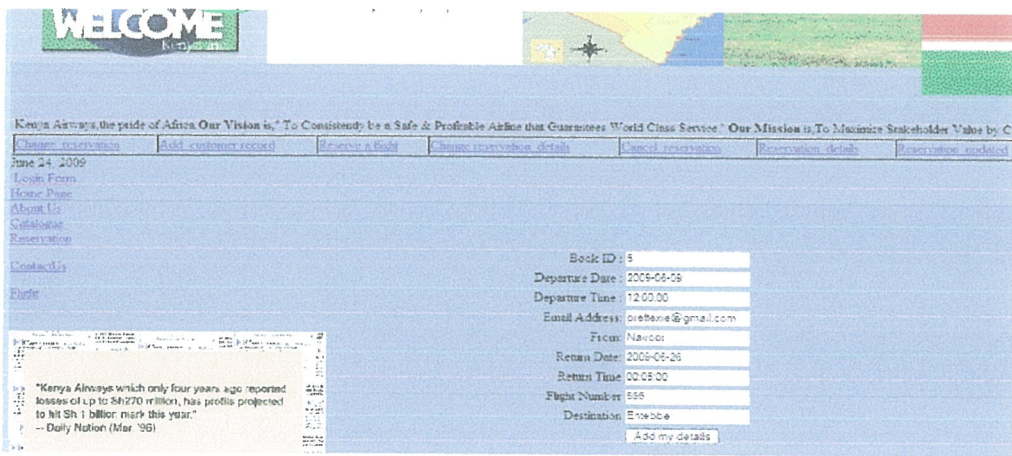


Figure 5.3: User Page

This page allows the user to enter their reservation details, which will be inserted into the database table, reservation. As a reminder the database structure for the reservation table after the above data entry is shown below.

Query results operations

Print view Print view (with full texts) Export

Show: 30 row(s) starting from record # 0

In horizontal mode and repeat headers after 100 calls

Sort by key: None Go

	from	bookId	to	departureDate	departureTime	returnDate	returnTime	email
<input type="checkbox"/>	Kenya	4	Dar	2006-04-05	20:00:00	2007-04-07	10:00:00	noelvihi@g
<input type="checkbox"/>	Mombasa	1	Kisumu	2009-05-02	13:00:00	2009-06-04	19:00:00	noelvihi@g
<input type="checkbox"/>	Eldoret	3	Mombasa	2009-06-28	16:00:00	2009-06-30	11:00:00	etamalice@
<input type="checkbox"/>	Nairobi	5	Entebbe	2009-06-09	13:00:00	2009-06-26	13:05:00	prettexie@

Check All / Uncheck All With selected:

Show: 30 row(s) starting from record # 0

In horizontal mode and repeat headers after 100 cells

Bookmark this SQL query

Label: ☐ Let every user access this bookmark

Table 5.3: Reservation table

This is the interface table which displays the passenger details in the database after direct entry through the form in the web browser. This is where the counter clerk selects the details for the passenger's flight reservation.

5.4 System Testing and Validation

Individual component testing and composite application testing was done to ensure that the proposed system conforms to the organization system requirements before deployment. Using the individual component testing, individual modules were frequently verified so that they are functioning as required. On ensuring their smooth functionality, composite application testing was applied to make sure that the modules can easily work together as a system as well as on the proposed system in line with management policies. To achieve the objective of testing and validation for the proposed system, the researcher used sample data for testing at all phases of the system implementation. The proposed system was validated to ensure integrity, availability and interoperability. The researcher used data constraints to several data inputs to ensure data consistence as well as security control measures, for example the passwords and user names, email address, among others hence optimal assurance of the system integrity.

To validate the system, the researcher used the following test data which tested the results against the desired system results.

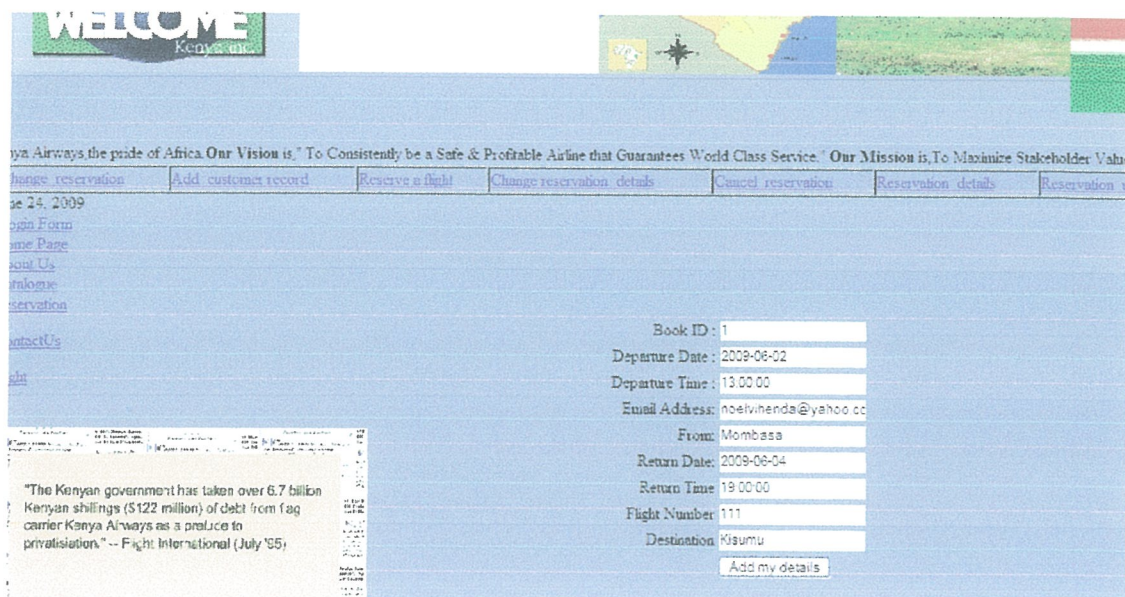


Figure 5.4.2: Validating Reservation Page

The researcher got the following results after testing the reservation page for submission. At this stage, the researchers uploaded this page to the server, call it up in the browser, and entered a test record. The researcher used a database administration tool, to check the data she typed if it has been entered correctly, and into the correct fields. If any data is incorrect, you can double-click on the Insert Record behavior, and the dialog box will appear again, allowing you to correct any problems. The completed page when viewed in a browser looks similar to the screenshot overleaf above.

To test the connection to the database, the researcher tried to retrieve data from the database as shown below, which produced the desired results below:

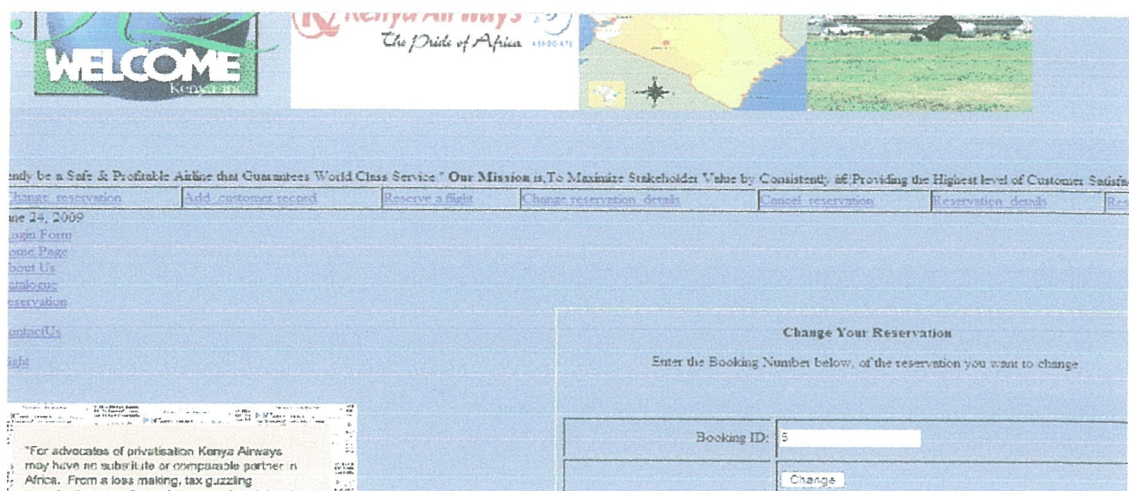


Figure 5.4.3: Testing Database Connectivity

This interface shows the details of what the passenger should enter to be able to change the reservation information in the database. When the passenger clicks on "Change" after entering the desired "Booking ID" the form sends this data to the `change_reservation_details.php` page.

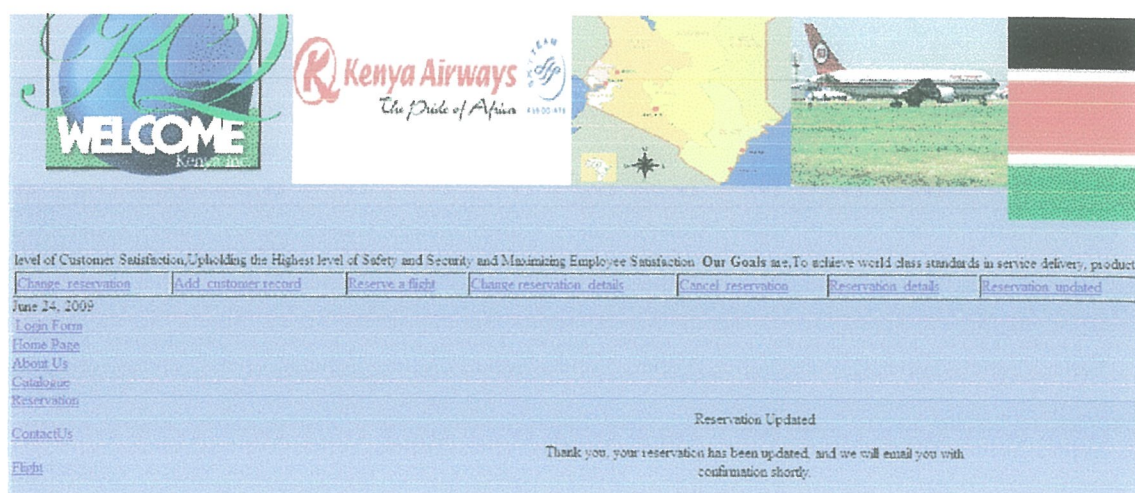


Figure 5.4.4: Results Produced after Testing Database Connectivity

This informs the passenger that the new reservation details have been saved and confirmed by the organization counter clerk.

5.5 File Creation and Conversion

File creation and conversions involves producing a file on a specified medium which is capable of being read or written by the computer software involved and free from errors. Until master file conversion has taken place, a computer system cannot become operational. The file conversion procedures are as follows:

- i. Record manually the existing data of the old file then transfer the file to a special form required by the new system.
- ii. Insert any new data into the file.
- iii. Transcribe the completed file into a new medium
- iv. Validate the file to ensure they are free of errors.

During the system testing, personnel from various department were involved in the testing, this apart from assisting to improve the confidence of the employee it also ensure error free system is implemented .each group has a special contribution namely.

- i. System manager checks if the system conforms to policy of organization.
- ii. System analysts to check whether objectives are met and the system is error free.
- iii. Programmers to debug the program in case of error.
- iv. The auditor to check whether controls in the system.
- v. The operation manager to check on the operational weakness of the system.
- vi. The user department to ensure the system is tested to their satisfaction.

5.6 Parallel Change Over

During the system development the researcher recommend that parallel change over method be used in interacting with the system. This means running the old system and new system simultaneously until new system is proved free of bugs and it is working correctly as required.

In conclusion, if the files are currently held on a computer system, then it should be possible to move the data from the present implementation to the target hardware and software. However, a thorough investigation should be done on cost and compatibility. Some routines may need to be written to modify the data after conversion.

The old and the new system are run, simultaneously for agreed period of time and results from the two systems are compared. Once the user has complete confidence in the new system, the old system is abandoned. All the steps undertaken during this project has been so useful to the researcher because she was able to realize the way how things work in the outside world that is out of what is theoretically studied in the classroom. The researcher has gained the skills in various fields like research and design among others. It has been so educative, and inspirational that all the students ought not to miss for doing the research and the necessary investigations.

CHAPTER SIX

DISCUSION, CONCLUSION AND RECOMMENDATION

This chapter examines how the objectives of the study were achieved using the web based system software developed for airline reservation, incorporated with module as well as validation using previous studies. The outcomes and contribution of the study to evaluation of changes in the software developed are discussed and conclusions given as given below.

6.1 Discussion

This section of the research provides a summarized overview of how the objectives and tasks of the project have been achieved, lesson learnt, recommendations as well as the conclusion and any other area for further research which should be carried out. The system achieves a lot more than the old system, it provide an environment that is both and convenient and efficient for the retrieval and storage of information.

The system provides an easy way of updating information, so that information changes in one file is automatically updated in all files. The system ensures that an effortless and speedy means of generating reports is in place. It ensures that Kenya Airways maximizes on its benefits and reduces on the operational costs. The passengers' can customize and extract the data they want to view. The system has a login form that provides a secure security that is hard for unauthorized users to access information in the database. Further more, software engineering principles like robustness, usability, interoperability, efficiency and platform were strictly followed making the software highly user friendly yet minimizing any undesirable trade offs.

The home page which was used as a template by the researcher consists of links to information for airline reservation pages that provides information for the user to make a reservation plus the necessary information required by the system user. The system enables the counter clerk to manage system login information, which is creating, modifying, deleting and viewing users' profiles.

The entire system was implemented by the researcher using web clips with the user interface and other components in HTML, MySQL, JavaScript and PHP together with XAMPP. It was the researcher's wish to automate the update of the knowledgebase and reasoning steps when new information is gained from experience obtained after using expert system module for a given number of times. However, this was not possible, because the programming tool (web clips) did not provide support for this.

6.2 Conclusion

The strength of a company to manage and make use of its stored sensitive data promises its excellence in today's competitive business world. This triggers the need to use all the information technology techniques available in order to secure and make good use of the available company data. Simplicity is the target everyone would wish to achieve in everyday life hence coming up with an automated reservation system to make passengers' life easy and to save the on costs. Transactions in an organization can be secured and updated when the information about the company and the customers' is managed and monitored effectively. The main purpose for developing this project was to provide the Kenya Airways staff with a way to monitor and contribute to the welfare of the passengers' reservations and better organization of the airline facilities. The research was able to achieve through the system produced as output of the project.

The researcher also wanted to provide users with more insight to the airline facilities which was achieved through the developed system. Lastly, the researcher wanted to make the information concerning the reservation available to the majority of users from Kenya and abroad. With the produced system being a website, this was achieved. All the steps undertaken during this project report have been useful to the researcher because she has come to realize the way how activities are done in the outside world that is out of university life. The researcher has gained skills in various fields like the creation of a dynamic website, research and design among others. It has been so educative, inspirational and interesting that any student ought not to miss the research project report.

The study provided solution of providing easy and faster way of accessing information by the users of the system. Nevertheless, findings of the study were limited to a specific target (Moi International Airport in Mombasa). The desired aim of providing quick access to information on airline reservations was more applicable to other citizens staying outside the country plus any other person who has the desire to tour the world was achieved.

6.3 Recommendations

Business systems can be so interactive once they cater for everything that relates to their organizations like the reservations online activities. This project can be improved by embedding in more dynamic utilities that were not possible due to the project scope but can improve the Kenya Airways business transactions. For the users to obtain accurate and up-to-date information which they can user to make suggestion and any concrete conclusions, the system need to be updated frequently, if possible on a daily basis. The database should be managed by a skilled person with the necessary expertise on database design and management. The database should be protected from intrusion because the information stored in it is very delicate.

The researcher's findings of the study recommended adoption and usage of a web based system developed for airline reservation, so as to improve on knowledge of manual record keeping of the customers', reduce on time accessibility of records as well as minimizing on costs of keeping the records by avoiding redundancy. Furthermore, a mechanism enabling updates to the system would also be available so that when there is need the system can be updated according to the passenger's details. This is because of the faster growing of technology which changes each and every time whereby new ideas are invented by other researchers. The web based system for the airline reservation developed can still be improved upon as discussed in the section below.

6.3.1 Areas for Further Research

To make the system more comprehensive the researcher suggests addition of the following: further research to be carried out on how to enter data in the database in XAMMP. For one to enter data in XAMMP, one should use forms created in the web pages or use the INSERT INTO command in SQL.

Using forms to enter data into the database is relatively easy but not all data can be entered through forms, for example in the Project table, there is no form created to enable one to enter the data into the database. On the other hand using SQL commands to enter data is a little bit difficult, cumbersome and requires skills. The researcher's suggestion is that research should be carried out on this to find easier ways of doing this like when creating the database or a table. The system can have the capability to be used in different languages. Further work is recommended so that it can have the capability of being usable in local languages like Swahili language for the East African region.

BIBLIOGRAPHY

1. *Business Information Systems 11th International Conference*, BIS 2008, Innsbruck, Austria, May 5-7, 2008. Proceedings
2. CS French (1992), *Computer Science* Fourth Edition, DP publications L.T.D Guernsey
3. Chen, P. P. and Zvieli, A., "Entity-Relationship Modeling of Fuzzy Data," *Proceedings of 2nd International Conference on Data Engineering*, Los Angeles, February 1986, pp. 320-327
4. DATE, C.J (2000) *An Introduction To Database Systems*_Seventh Edition Thomson Press, New Delhi
5. Delaney, Kalen, et al (2007) *Inside SQL Server 2005: Query Tuning and Optimization*. Microsoft Press
6. Dittman et. al, 2001. *Systems Analysis and Design Methods*, Irwin /McGraw-Hill; 5th Edition, New York
7. Downes, Green, Mairlot (2003) *Dreamweaver MX-PHP Web Development*, New York
8. E. H. Chi, "The Scent of a Site," *Proceedings of the Conference on Human Factors in Computing*, Los Angel
9. Hutchison et. al, (1999) *Using Information Technology*, Irwin /McGraw-Hill, Third Edition
10. James Martin, (1976). *Principals of database management*, Prentice Hall of India Private LTD, Delhi
11. Jefrey, Lonnie and Kelvin, (2000). *System Analysis And Design Methods*._Fifth Edition, Irwin Mc Graw_Hill , New York
12. J. Carroll, Wiley, (1995) *Scenario-Based Design*. New York, NY
13. Sarah .E. Hutchinson and Stacey.C.Sawyer (2000), *Computers, Communications And Information* Seventh Edition, Irwin Me _Graw, New York
14. *School of Business and Science*, Middlesesex University, Burroughs Road, NW4 4BT London, UK
15. Shelley Lowery <http://www.cyberindian.com/web-designing/essential-website-components.php>
16. *Systems, CHI 2000*, pp. 161–168, The Hague, The Computing Netherlands, 2000.

17. P. Sawyer (2000) *Designing for Scent*, technical report, User Interface Engineering-New York.
18. Williams, sawyer and Hutchinson (1999).*Using Information Technology*_Third Edition Irwin McGraw_Hill, New York.

APPENDICES

Appendix A: Questionnaire Guide

Questionnaire on the System Analysis for the MIA

This questionnaire is prepared in preparation of the Kenya airline information storage and retrieval management system. A system analysis research which will facilitate the design intended to improve on the current system used at MIA which will be customized to the present operations carried out.

An efficient, dependable, technologically advanced, sophisticated, comprehensive, information system for Kenya Airways plays a vital role in the smooth or round functioning of the organization.

You have been considered a key informative in this research and your views are very important .The information provided will be treated confidentially.

Background of Information

Time interview started.....
Time interview ended.....
Location.....
Name of interviewee.....
Title/Position.....
Name of interviewer.....

Processes used in the basic operations of the system

1. What are the basic processes carried out in this organization of the Kenya Airways?.....
.....
.....

2. How do you obtain the data from the different departments within this organization of Kenya Airways?

.....
.....

3. What is the average time needed to process a transaction file (receipt) before it is issued to the passenger?

.....
.....

4. What are the sources of data input into this system?

.....
.....

5. What are the reports generated at the end of each process?

.....
.....

6. In which format is data presented after processing?

☐ Statistical ☐ Text

7. How often is the data entered into the system?

☐ Daily ☐ Weekly ☐ Monthly

Others specify with a reason

.....
.....
.....

9. What are the procedures of work being done manually or electronically?

.....
.....

Connectivity and data flow within the system

10. How many computers are connected to the system in place?

☐ 0-5 ☐ 5-10 ☐ 10-15 ☐ More

11. Which parties are interested in this data after it has been processed?

.....
.....

12. What data is shared with other organization?

.....
.....

Controls used to ensure system security

13. What performance controls are in place to ensure data security?

.....
.....

14. How do you ensure the confidentiality of the data in files during transmission?

.....
.....

Challenges faced while using the existing system

15. What are the problems faced while using the current system?

.....
.....

16. What do you think should be done to address the problems mentioned above?

.....
.....

17. What are the possible measures put in place to cater for the unpredictable disasters that may damage the data in storage?

.....
.....

18. How do you rate the performance of the existing system? Why?

☐ Very good ☐ Good ☐ Fair ☐ Poor

☐ Very poor

.....

Appendix B: Sample Codes

Home Page

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<!-- DW6 -->
<head>
<!-- Copyright 2005 Macromedia, Inc. All rights reserved. -->
<!-- TemplateBeginEditable name="doctitle" -->
<title>Home Page</title>
<!-- TemplateEndEditable -->
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<link rel="stylesheet" href="../../mm_travel2.css" type="text/css" />
<script language="JavaScript" type="text/javascript">
//----- LOCALIZEABLE GLOBALS -----
var d=new Date();
var
monthname=new
Array("January","February","March","April","May","June","July","August","September",
,"October","November","December");
//Ensure correct for language. English is "January 1, 2004"
var TODAY = monthname[d.getMonth()] + " " + d.getDate() + ", " + d.getFullYear();
//----- END LOCALIZEABLE -----
</script>
<!-- TemplateBeginEditable name="head" --><!-- TemplateEndEditable -->
<style type="text/css">
<!--
.style1 {color: #BDD3FB}
.style2 {font-family: Geneva, Arial, Helvetica, sans-serif}
-->
</style>
</head>
```


Consistently â€¦Providing the Highest level of Customer Satisfaction,Upholding the Highest level of Safety and Security and Maximizing Employee Satisfaction. Our Goals are,To achieve world class standards in service delivery, product quality and operational performance.To be the Airline of choice in Africa .To develop MIA as a premier hub in Africa. To pursue a business model that will deliver consistent level of profitability. In summary our actions, our behaviour and our attitude at work will be driven by safety, customer satisfaction and quality considerations.

```

</marquee>
<table width="100%" border="1">
<tr>
<td><a href=" ../Reservations/Change
Reservation.php">Change_reservation</a></td>
<td><a href=" ../Reservations/passenger details.php">Add_customer record
</a></td>
<td><a href=" ../Reservations/change.php">Reserve a flight </a></td>
<td><a href=" ../Reservations/Change Reservation details.php">Change
reservation_details </a></td>
<td><a href=" ../Reservations/Cancel
Reservation.php">Cancel_reservation</a></td>
<td><a href=" ../Reservations/passenger details.php">Reservation_details</a></td>
<td><a href=" ../Reservations/Reservation
updated.php">Reservation_updated</a></td>
<td><a href=" ../Reservations/Reservation
cancelled.php">Reservation_cancelled</a></td>
<td><a href=" ../Reservations/Confirm
reservation.php">Confirm_reservation</a></td>
</tr>
</table>
<div align="left">
<script language="JavaScript" type="text/javascript">

```

```

        document.write(TODAY);        </script>
    </div> </td>
</tr>
<tr>
    <td colspan="7" bgcolor="#003366"></td>
</tr>
<tr>
    <td width="366" valign="top" bgcolor="#9999CC">
        <table border="0" cellpadding="0" cellspacing="0" width="165"
id="navigation">
            <tr>
                <td width="165">&nbsp;<a href="../LoginForm.php">Login Form</a>
            </td>
            </tr>
            <tr>
                <td width="165"><a href="/AIRLINE/home_page.php" class="navText">Home
Page </a></td>
            </tr>
            <tr>
                <td width="165"><a href="../about us.php" class="navText">About Us </a></td>
            </tr>
            <tr>
                <td width="165"><a href="../catalogue.php" class="navText">Catalogue</a></td>
            </tr>
            <tr>
                <td width="165"><a href="../Reservation.php"
class="navText">Reservation</a></td>
            </tr>
            <tr>

```

```

        <td width="165"><p><a href="../contact us.php" class="navText">
ContactUs</a></p>
        </td>
</tr>
<tr>
        <td><a href="../Flight.php">Flight</a></td>
</tr>
</table>
        <br /> <span class="pageName"></span><br />
        <br /><br /></td>
        <td width="187" bgcolor="#9999CC"><br />
        <td colspan="2" valign="top" bgcolor="#9999CC"><br />
        <br />
        <br />
        <table width="305" border="0" cellpadding="0" cellspacing="0"
bgcolor="#9999CC">
        <tr>
        <td align="left" valign="top" class="pageName"><br />
</td>
        </tr>
        <tr>
        <td valign="top" class="bodyText"><!-- TemplateBeginEditable name="Main
Content" -->Main Content<!-- TemplateEndEditable --></td>
        </tr>
</table>
<br /></td>
        <td width="187" bgcolor="#9999CC"><br />
        <td width="299" valign="top" bgcolor="#9999CC"><br />
        <br />
        <table border="0" cellspacing="0" cellpadding="0" width="190">

```

```

        <tr>
        <td colspan="3" class="subHeader" align="center">NEW
DESTINATIONS</td>
        </tr>
        <tr>
        <td width="40" bgcolor="#9999CC">&nbsp;</td>
        <td width="110" valign="top" bgcolor="#9999CC"
class="smallText" id="sidebar"><br />
        <p><br />
        <br />
        <a href="javascript:;"></a></p>
        <br />
        &nbsp;<br />
        &nbsp;<br />
        </td>
        <td width="40" bgcolor="#9999CC">&nbsp;</td>
        </tr>
</table>
        <td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
<tr>
        <td width="366"></td>
        <td width="187"></td>
        <td width="502"></td>
        <td width="5">&nbsp;</td>
        <td width="187"><p align="center" class="style2">KIYAYI .N. VIHENDA</p>
<p align="center" class="style2">BCS/7510/71/DF</p>
<p align="center" class="style2">Copyright (C) 2008/2009 </p></td>

```

```
<td width="299"></td>
<td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
</table>
</body>
</html>
```


Login Form

```
<?php require_once('Connections/conn.php'); ?>

<?php
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
$theNotDefinedValue = "")
{
    $theValue = (!get_magic_quotes_gpc()) ? addslashes($theValue) : $theValue;
    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "long":
        case "int":
            $theValue = ($theValue != "") ? intval($theValue) : "NULL";
            break;
        case "double":
            $theValue = ($theValue != "") ? "'" . doubleval($theValue) . "'" : "NULL";
            break;
        case "date":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "defined":
            $theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
            break;
    }
    return $theValue;
}

$editFormAction = $_SERVER['PHP_SELF'];
if (isset($_SERVER['QUERY_STRING'])) {
    $editFormAction .= "?" . htmlentities($_SERVER['QUERY_STRING']);
}
```

```

if ((isset($_POST["MM_insert"])) && ($_POST["MM_insert"] == "form1")) {
    $insertSQL = sprintf("INSERT INTO login (username, password) VALUES (%s, %s)",
        GetSQLValueString($_POST['username'], "text"),
        GetSQLValueString($_POST['password'], "text"));

    mysql_select_db($database_conn, $conn);
    $Result1 = mysql_query($insertSQL, $conn) or die(mysql_error());
    $insertGoTo = "Reservations/passenger details.php";
    if (isset($_SERVER['QUERY_STRING'])) {
        $insertGoTo .= (strpos($insertGoTo, '?')) ? "&" : "?";
        $insertGoTo .= $_SERVER['QUERY_STRING'];
    }
    header(sprintf("Location: %s", $insertGoTo));
}

?><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<!-- DW6 -->
<head>
<!-- Copyright 2005 Macromedia, Inc. All rights reserved. -->
<title>Home Page</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<link rel="stylesheet" href="mm_travel2.css" type="text/css" />
<script language="JavaScript" type="text/javascript">
//----- LOCALIZEABLE GLOBALS -----
var d=new Date();
var
monthname=new
Array("January","February","March","April","May","June","July","August","September",
"October","November","December");
//Ensure correct for language. English is "January 1, 2004"
var TODAY = monthname[d.getMonth()] + " " + d.getDate() + ", " + d.getFullYear();
//----- END LOCALIZEABLE -----

```

```

</script>
<style type="text/css">
<!--
.style1 {color: #BDD3FB}
.style2 {font-family: Geneva, Arial, Helvetica, sans-serif}
-->
</style>
</head>
<body bgcolor="#C0DFFD">
<table width="1552" height="1351" border="0" align="center" cellpadding="0"
cellspacing="0">
  <tr bgcolor="#3366CC">
    <td colspan="3" rowspan="2" valign="top" bgcolor="#9999CC" class="style1"></td>
    <td height="19" colspan="3" align="center" valign="bottom" nowrap="nowrap"
bgcolor="#9999CC" id="logo">WELCOME TO MOMBASA INTERNATIONAL
AIRPORT </td>
    <td width="6" bgcolor="#9999CC">&nbsp;</td>
  </tr>
  <tr bgcolor="#3366CC">
    <td height="113" colspan="3" align="center" valign="top" bordercolor="#ECE9D8"
bgcolor="#9999CC" id="tagline"><p></p> </td>
    <td width="6" bgcolor="#9999CC">&nbsp;</td>
  </tr>
  <tr>
    <td colspan="3" bgcolor="#003366"></td>
  </tr>

```



```

<td colspan="2" valign="top" bgcolor="#9999CC"><br />
    &nbsp;<br />
    &nbsp;<br />
    <table width="305" border="0" cellpadding="0" cellspacing="0"
bgcolor="#9999CC">
    <tr>
        <td align="left" valign="top" class="pageName">&nbsp;</td>
    </tr>
    <tr>
        <td valign="top" class="bodyText">
<br /><form action="" method="POST" name="form1" id="form1">
    <table width="100%" border="0" cellspacing="0" cellpadding="1" class="TitleColor">
    <tr style="vertical-align: top">
        <td>
            <table width="100%" border="0" cellspacing="0" cellpadding="4">
            <tr class="HeaderColor">
                <td width="20%" style="vertical-align: top">
</td>
                <td style="vertical-align: top">
                    <h3>Sign In</h3>
</td>
            </tr>
            <tr>
                <td width="20%" style="vertical-align: top">
</td>
                <td style="vertical-align: top">
                    <h3>Sign In</h3>
</td>
            </tr>
            <tr style="vertical-align: top">
                <td width="20%" class="TitleColor">
                    <label for="username"><strong>User Name</strong></label>&nbsp;<br />
                    <input id="username" name="username" type="text" size="25" />
                    <p>&nbsp;</p>
                    <label for="password"><strong>Password</strong></label>&nbsp;<br />
                    <input id="password" name="password" type="password" size="25" />
                    <p>
                    <input type="submit" name="ButtonName" value="Sign In" />

```

```

        <a href="home page.php"></a></p>        </td>
<td width="80%" class="StoryContentColor">
    <h4 class="TitleColor">Instructions:</h4>
    Only Authorised Members are supposed to Login the </td>
</tr>
</table>    </td>
</tr>
</table>
<input type="hidden" name="MM_insert" value="form1">
</form></td>
    </tr>
</table>
<br /></td>
<td width="187" bgcolor="#9999CC">&nbsp;</td>
    <td width="299" valign="top" bgcolor="#9999CC"><br />
        &nbsp;<br />
        <table border="0" cellspacing="0" cellpadding="0" width="190">
            <tr>
                <td colspan="3" class="subHeader" align="center">NEW
DESTINATIONS</td>
            </tr>
            <tr>
                <td width="40" bgcolor="#9999CC">&nbsp;</td>
                <td width="110" valign="top" bgcolor="#9999CC"
class="smallText" id="sidebar"><br />
                <p><br />
                <br />
                <a href="javascript:;"></a></p>
                <br />

```

```

        &nbsp;<br />
        &nbsp;<br />                                </td>
        <td width="40" bgcolor="#9999CC">&nbsp;</td>
        </tr>
</table>    </td>
<td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
<tr>
    <td width="366"></td>
    <td width="187"></td>
    <td width="502"></td>
    <td width="5">&nbsp;</td>
    <td width="187"><p align="center" class="style2">KIYAYI .N. VIHENDA</p>
<p align="center" class="style2">BCS/7510/71/DF</p>
<p align="center" class="style2">Copyright (C) 2008/2009 </p></td>
    <td width="299"></td>
    <td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
</table>
</body>
</html>

```

Passenger Details Page

```
<?php require_once('../Connections/conn.php'); ?>

<?php
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
$theNotDefinedValue = "")
{
    $theValue = (!get_magic_quotes_gpc()) ? addslashes($theValue) : $theValue;

    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "long":
        case "int":
            $theValue = ($theValue != "") ? intval($theValue) : "NULL";
            break;
        case "double":
            $theValue = ($theValue != "") ? "'" . doubleval($theValue) . "'" : "NULL";
            break;
        case "date":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "defined":
            $theValue = ($theValue != "") ? $theDefinedValue : $theNotDefinedValue;
            break;
    }
    return $theValue;
}

$editFormAction = $_SERVER['PHP_SELF'];
if (isset($_SERVER['QUERY_STRING'])) {
```



```

SeditFormAction .= "?" . htmlentities($_SERVER['QUERY_STRING']);
}

if ((isset($_POST["MM_insert"])) && ($_POST["MM_insert"] == "form1")) {
    $insertSQL = sprintf("INSERT INTO passenger (fName, lName, age, Gender,
emailAddress, idNo, bookId) VALUES (%s, %s, %s, %s, %s, %s, %s)",
        GetSQLValueString($_POST['fname'], "text"),
        GetSQLValueString($_POST['lname'], "text"),
        GetSQLValueString($_POST['age'], "int"),
        GetSQLValueString($_POST['gender'], "text"),
        GetSQLValueString($_POST['email'], "text"),
        GetSQLValueString($_POST['idno'], "text"),
        GetSQLValueString($_POST['bookid'], "int"));

    mysql_select_db($database_conn, $conn);
    $Result1 = mysql_query($insertSQL, $conn) or die(mysql_error());

    $insertGoTo = "../home page.php";
    if (isset($_SERVER['QUERY_STRING'])) {
        $insertGoTo .= (strpos($insertGoTo, '?')) ? "&" : "?";
        $insertGoTo .= $_SERVER['QUERY_STRING'];
    }
    header(sprintf("Location: %s", $insertGoTo));
}

?><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"><!-- InstanceBegin
template="/Templates/HOME.dwt" codeOutsideHTMLOIsLocked="false" -->
<!-- DW6 -->
<head>
<!-- Copyright 2005 Macromedia, Inc. All rights reserved. -->

```

```

<!-- InstanceBeginEditable name="doctitle" -->
<title>Home Page</title>
<!-- InstanceEndEditable -->
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<link rel="stylesheet" href="../mm_travel2.css" type="text/css" />
<script language="JavaScript" type="text/javascript">
//----- LOCALIZEABLE GLOBALS -----
var d=new Date();
var monthname=new
Array("January","February","March","April","May","June","July","August","September",
,"October","November","December");
//Ensure correct for language. English is "January 1, 2004"
var TODAY = monthname[d.getMonth()] + " " + d.getDate() + ", " + d.getFullYear();
//----- END LOCALIZEABLE -----
</script>
<!-- InstanceBeginEditable name="head" --><!-- InstanceEndEditable -->
<style type="text/css">
<!--
.style1 {color: #BDD3FB}
.style2 {font-family: Geneva, Arial, Helvetica, sans-serif}
-->
</style>
</head>
<body bgcolor="#C0DFFD">
<table width="1552" height="1351" border="0" align="center" cellpadding="0"
cellspacing="0">
<tr bgcolor="#3366CC">
<td colspan="3" rowspan="2" valign="top" bgcolor="#9999CC" class="style1"><img

```


Airline of choice in Africa .To develop MIA as a premier hub in Africa. To pursue a business model that will deliver consistent level of profitability. In summary our actions, our behaviour and our attitude at work will be driven by safety, customer satisfaction and quality considerations.

```

</marquee>
<table width="100%" border="1">
<tr>
<td><a href="Change%20Reservation.php">Change_reservation</a></td>
<td><a href="passenger%20details.php">Add_customer record </a></td>
<td><a href="change.php">Reserve a flight </a></td>
<td><a href="Change%20Reservation%20details.php">Change reservation_details
</a></td>
<td><a href="Cancel%20Reservation.php">Cancel_reservation</a></td>
<td><a href="passenger%20details.php">Reservation_details</a></td>
<td><a href="Reservation%20updated.php">Reservation_updated</a></td>
<td><a href="Reservation%20cancelled.php">Reservation_cancelled</a></td>
<td><a href="Confirm%20reservation.php">Confirm_reservation</a></td>
</tr>
</table>
<div align="left">
<script language="JavaScript" type="text/javascript">
document.write(TODAY); </script>
</div> </td>
</tr>
<tr>
<td colspan="7" bgcolor="#003366"></td>
</tr>
<tr>
<td width="366" valign="top" bgcolor="#9999CC">

```

```

        <table border="0" cellspacing="0" cellpadding="0" width="165"
id="navigation">
    <tr>
        <td width="165">&nbsp;<a href=" ../LoginForm.php">Login Form</a>
    </td>
</tr>
<tr>
    <td width="165"><a href="/AIRLINE/home page.php" class="navText">Home
Page </a></td>
</tr>
<tr>
    <td width="165"><a href=" ../about%20us.php" class="navText">About Us
</a></td>
</tr>
<tr>
    <td width="165"><a href=" ../catalogue.php" class="navText">Catalogue</a></td>
</tr>
<tr>
    <td width="165"><a href=" ../Reservation.php"
class="navText">Reservation</a></td>
</tr>
<tr>
    <td width="165"><p><a href=" ../contact%20us.php" class="navText">
ContactUs</a></p>
</td>
</tr>
<tr>
    <td><a href=" ../Flight.php">Flight</a></td>
</tr>
</table>

```

```

        <br /> &nbsp;<span class="pageName"></span><br /> &nbsp;<br />
&nbsp;<br /></td>
    <td width="187" bgcolor="#9999CC">&nbsp;</td>
    <td colspan="2" valign="top" bgcolor="#9999CC"><br />
    &nbsp;<br />
    &nbsp;<br />
    <table width="305" border="0" cellpadding="0" cellspacing="0"
bgcolor="#9999CC">
    <tr>
    <td align="left" valign="top" class="pageName">&nbsp;</td>
    </tr>
    <tr>
    <td valign="top" class="bodyText"><!-- InstanceBeginEditable name="Main
Content" -->
    <link
href="file:///D:/newprj/pr/noel/sppsl/DYNAMICHOTEL/CSS/Level2_Verdana_Text.css"
rel="stylesheet" type="text/css" media="all" />

<table width="100%" height="20" border="0" align="left">
    <tr>
    <td>&nbsp;</td>
    </tr>
    <tr>
    <td><form id="form1" name="form1" method="POST" action="<?php echo
$editFormAction; ?>">
    <table width="503" border="0" align="center">
    <tr>
    <td width="195" align="right">Age: </td>
    <td width="298"><label>

```

```

<input name="age" type="text" id="age" maxlength="10" />
    </label></td>
</tr>
<tr>
    <td align="right">Book ID : </td>
    <td><label>
        <input name="bookid" type="text" id="bookid" maxlength="10" />
    </label></td>
</tr>
<tr>
    <td align="right">Email Address : </td>
    <td><input name="email" type="text" id="email" maxlength="10" /></td>
</tr>
<tr>
    <td align="right">First Name </td>
    <td><label>
        <input name="fname" type="text" id="fname" maxlength="10" />
    </label></td>
</tr>
<tr>
    <td><div align="right">Last Name </div></td>
    <td><label>
        <input name="lname" type="text" id="lname" maxlength="10" />
    </label></td>
</tr>
<tr>
    <td align="right">Gender: </td>
    <td><label>
        <select name="gender" id="gender">
            <option value="Male">Male</option>
            <option value="Female">Female</option>
        </select>
    </label>
</td>
</tr>

```

```

        </select>
    </label></td>
</tr>
<tr>
    <td align="right">ID Number : </td>
    <td><label>
        <input name="idno" type="text" id="idno" maxlength="10" />
    </label></td>
</tr>

<tr>
    <td height="22"><input name="clientID" type="hidden" id="clientID" /></td>
    <td><label>
        <input name="Add Customer Details" type="submit" id="Add Customer Details"
value="Add Customer Details" />
    </label></td>
</tr>
</table>
    <input type="hidden" name="MM_insert" value="form1">
</form>
</td>
</tr>
</table>
<!-- InstanceEndEditable --></td>
    </tr>
</table>
<br /></td>
<td width="187" bgcolor="#9999CC">&nbsp;</td>
    <td width="299" valign="top" bgcolor="#9999CC"><br />
        &nbsp;<br />
        <table border="0" cellspacing="0" cellpadding="0" width="190">

```



```

        <tr>
        <td colspan="3" class="subHeader" align="center">NEW
DESTINATIONS</td>
        </tr>

        <tr>
        <td width="40" bgcolor="#9999CC">&nbsp;</td>
        <td width="110" valign="top" bgcolor="#9999CC"
class="smallText" id="sidebar"><br />
        <p><br />
        <br />
        <a href="javascript:;"></a></p>
        <br />
        &nbsp;<br />
        &nbsp;<br />
        </td>
        <td width="40" bgcolor="#9999CC">&nbsp;</td>
        </tr>
</table>
        <td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
<tr>
        <td width="366"></td>
        <td width="187"></td>
        <td width="502"></td>
        <td width="5">&nbsp;</td>
        <td width="187"><p align="center" class="style2">KIYAYI .N. VIHENDA</p>
<p align="center" class="style2">BCS/7510/71/DF</p>

```

```
<p align="center" class="style2">Copyright (C) 2008/2009 </p></td>
<td width="299"></td>
<td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
</table>
</body>
<!-- InstanceEnd --></html>
```

Book a Flight Page

```
<!DOCTYPE html PUBLIC "-//W3C/DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"><!-- InstanceBegin
template="/Templates/HOME.dwt" codeOutsideHTMLIsLocked="false" -->
<!-- DW6 -->
<head>
<!-- Copyright 2005 Macromedia, Inc. All rights reserved. -->
<!-- InstanceBeginEditable name="doctitle" -->
<title>Home Page</title>
<!-- InstanceEndEditable -->
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<link rel="stylesheet" href="../mm_travel2.css" type="text/css" />
<script language="JavaScript" type="text/javascript">
//----- LOCALIZEABLE GLOBALS -----
var d=new Date();
var monthname=new
Array("January","February","March","April","May","June","July","August","September"
,"October","November","December");
//Ensure correct for language. English is "January 1, 2004"
var TODAY = monthname[d.getMonth()] + " " + d.getDate() + ", " + d.getFullYear();
//----- END LOCALIZEABLE -----
</script>
<!-- InstanceBeginEditable name="head" --><!-- InstanceEndEditable -->
<style type="text/css">
<!--
.style1 {color: #BDD3FB}
.style2 {font-family: Geneva, Arial, Helvetica, sans-serif}
-->
</style>
</head>
```

[illegible]

Consistently Providing the Highest level of Customer Satisfaction,Upholding the Highest level of Safety and Security and Maximizing Employee Satisfaction. **Our Goals** are,To achieve world class standards in service delivery, product quality and operational performance.To be the Airline of choice in Africa .To develop MIA as a premier hub in Africa. To pursue a business model that will deliver consistent level of profitability. In summary our actions, our behaviour and our attitude at work will be driven by safety, customer satisfaction and quality considerations.

```

</marquee>
<table width="100%" border="1">
<tr>
<td><a href="Change%20Reservation.php">Change_reservation</a></td>
<td><a href="passenger%20details.php">Add_customer record </a></td>
<td><a href="change.php">Reserve a flight </a></td>
<td><a href="Change%20Reservation%20details.php">Change reservation_details
</a></td>
<td><a href="Cancel%20Reservation.php">Cancel_reservation</a></td>
<td><a href="passenger%20details.php">Reservation_details</a></td>
<td><a href="Reservation%20updated.php">Reservation_updated</a></td>
<td><a href="Reservation%20cancelled.php">Reservation_cancelled</a></td>
<td><a href="Confirm%20reservation.php">Confirm_reservation</a></td>
</tr>
</table>
<div align="left">
<script language="JavaScript" type="text/javascript">
document.write(TODAY); </script>
</div> </td>
</tr>
<tr>
<td colspan="7" bgcolor="#003366"></td>

```

```

</tr>
<tr>
  <td width="366" valign="top" bgcolor="#9999CC">
    <table border="0" cellspacing="0" cellpadding="0" width="165"
id="navigation">
      <tr>
        <td width="165">&nbsp;<a href=" ../LoginForm.php">Login Form</a>
      </td>
    </tr>
    <tr>
      <td width="165"><a href="/AIRLINE/home page.php" class="navText">Home
Page </a></td>
    </tr>
    <tr>
      <td width="165"><a href=" ../about%20us.php" class="navText">About Us
</a></td>
    </tr>
    <tr>
      <td width="165"><a href=" ../catalogue.php" class="navText">Catalogue</a></td>
    </tr>
    <tr>
      <td width="165"><a href=" ../Reservation.php"
class="navText">Reservation</a></td>
    </tr>
    <tr>
      <td width="165"><p><a href=" ../contact%20us.php" class="navText">
ContactUs</a></p>
    </td>
    </tr>
    <tr>
      <td><a href=" ../Flight.php">Flight</a></td>

```

```

</tr>
</table>
    <br /> <span class="pageName"></span><br />      <br />
<br /><br /></td>
    <td width="187" bgcolor="#9999CC"><br />
    <td colspan="2" valign="top" bgcolor="#9999CC"><br />
    <br />
    <br />
    <table width="305" border="0" cellpadding="0" cellspacing="0"
bgcolor="#9999CC">
    <tr>
    <td align="left" valign="top" class="pageName"><br /></td>
    </tr>
    <tr>
    <td valign="top" class="bodyText"><!-- InstanceBeginEditable name="Main
Content" -->
    <p><br /></p>
    <p><br /></p>
    <p><br /></p>
    <form id="form1" name="form1" method="post" action="Change
Reservation.php">
    <p align="center"><strong>change</strong></p>
    <p align="center">If you have already booked a flight with us before,enter your
book id in the box<br />
    provided below to change, otherwise</p>
    <p align="center"><a href="passenger_details.php">Add yourself as a new
Customer.</a></p>
    <table width="633" height="106" border="1" cellpadding="2" cellspacing="5">
    <tr>

```

```

<td><div align="right">Booking ID: </div></td>
<td><label>
    <input name="bookingID" type="text" id="bookingID" maxlength="30" />
</label></td>
</tr>
<tr>
<td>&nbsp;</td>
<td><label>
    <input type="submit" name="Submit" value="Change" />
</label></td>
</tr>
</table>
</form>
<p>&nbsp;</p>
<!-- InstanceEndEditable --></td>
</tr>
</table>
<br /></td>
<td width="187" bgcolor="#9999CC">&nbsp;</td>
<td width="299" valign="top" bgcolor="#9999CC"><br />
    &nbsp;<br />
    <table border="0" cellspacing="0" cellpadding="0" width="190">
        <tr>
            <td colspan="3" class="subHeader" align="center">NEW
DESTINATIONS</td>
        </tr>
        <tr>
            <td width="40" bgcolor="#9999CC">&nbsp;</td>
            <td width="110" valign="top" bgcolor="#9999CC"
class="smallText" id="sidebar"><br />

```



```

        <p><br />

        <br />
        <a href="javascript:;"></a></p>
        <br />
        &nbsp;<br />
        &nbsp;<br />                                </td>
        <td width="40" bgcolor="#9999CC">&nbsp;</td>
        </tr>
</table>    </td>
<td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
<tr>
    <td width="366"></td>
    <td width="187"></td>
    <td width="502"></td>
    <td width="5">&nbsp;</td>
    <td width="187"><p align="center" class="style2">KIYAYI .N. VIHENDA</p>
<p align="center" class="style2">BCS/7510/71/DF</p>
<p align="center" class="style2">Copyright (C) 2008/2009 </p></td>
    <td width="299"></td>
    <td width="6" bgcolor="#9999CC">&nbsp;</td>
</tr>
</table>
</body>
<!-- InstanceEnd --></html>

```