Name : *Saravanabavan A/L Ramasamy*

Matrix No : *WET 98045*

Supervisor : *Assoc. Prof. Dr. P. Sellapan*

Moderator : *Mr. Chiew Thiam Kian*

Title : *System for Analyzing Medical Data*

A Graduation Project presented to
The Faculty of Computer Science and Information Technology
University of Malaya
In partial fulfillment of the requirements for the
Degree of Bachelor of Information Technology

INFORMATION TECHNOLOGY
UNIVERSITY OF MALAYA
2001/2002

ABSTRACT

This project can be defined as a stand-alone system, which attempts to integrate with Healthcare Information System to communicate the outside and inside flow of information in hospitals and provide the functions common for all applications.

This web-based Medical Data Analysis System allows all range of users, using Internet Explorer web browser to view the graphical output from the patient's data and gaining mild information about the top ten diseases ranked in Malaysia or specifically University Malaya Medical Center. On the other hand, it allows the authorized user to update the latest patent medical data and diseases name as time changes. The rational behind this development is to give comparison in several aspect such as sex, race, state, height, weight, occupations and age.

A combination of waterfall model and prototyping model was selected for the development process because it supports rapid application development and reduces the risk involved. Software engineering principles based on this methodology were applied throughout the whole development phase of the system such as system analysis, implementation, design, testing, and coding.

The development tools selected for this project were Microsoft Visual Basic 6.0, Intercooled Stata software 6.0 and ACD See. The patient's data are kept in stata's Data Editor and Internet Information Server is used as the web server. Microsoft Windows 2000 has been chosen as the operating system where it is known for it's powerful platform for web application development and deployment.

ACKNOWLEDGEMENTS

In this page, I would like to express my deepest regards to the following people for assisting me in my project. They are:

- FSKTM UM for giving this great opportunity to show my talent in developing a system and gaining experience from it.
- Dr. P.SELLAPAN, my supervisor for his guidance and helping hand.
- Mr. CHIEW THIAM KIAN, my moderator for keeping me alert in my mistakes during this project.
- My fellow friends especially Premjit, Vimalji, Joe Fernandez and Kiran for their ideas to write my thesis in a proper manner.
- Dr. BHARTHI, regarding the patients medical data to be processed
- The person whom my heart beats for, Badmini. Without her, this thesis would never have been completed successfully.
- Last but not least, my family for their moral support and encouragement that kept my spirits running high when the going got tough.

"A special tribute goes to the ALMIGHTY for giving me the mental clarity and physical strength during the arduous times encountered in completing this thesis."

Table of Contents

List of Figures

CHAPTER 1: INTRODUCTION

1.1 OVERVIEW

System for Analyzing Medical Data (SAMD) is a system where medical data of software that could bring out results as a information for the public. This information shows each disease or illness according gender, race, age and others. The information would be web-based system so that anyone could refer it from the internet. It's a stand-alone system and later it might be able to be integrated with other systems such as Pharmaceutical Information System, Patient Information System, Medical Center Information System, Health Insurance Information and Billing System, Health Care Providers Information System.

A System for Analyzing Medical Data would quite certainly be defined as an open system which provides the communication to the outside, together with the inside of the domain of the medical center which also covers the flow of the information within the above mentioned precincts. This system would give access to all demographics pertaining the particular medical center. Thus, it would be created on a platform that is compatible to support users and be able to help them achieve an effective information administration.

System for Analyzing Medical Data (SAMD), facilitates the management of different hospital services by efficiently providing information on the fly to various people in charge of said services and people interested in the particular information. This paperless office effort virtually eliminates space constraints and manual efforts. It also attempts to provide both a conceptual framework in which it would

demonstrate the manner in which web applications could be fitted together as a practical inspiration for this emerging domain.

It also helps the public to generate themselves which medical center that they should go according to their budget and disease that affects them. It also gives a little bit of knowledge to the patient before they attend themselves to a doctor. This would help both parties either the patients or the doctors.

1.2 OBJECTIVES

- To give a clear picture in percentages where a disease could effect by gender, age, race and others. So, the management or the public could take a minor attention to the particular diseases where they might be high chances for them getting it.

- To study and investigate the current Web information system, implementation together with developing tools and their relevance to the project.

- To make sure the high availability of the system where the response time is fast and the confidentiality and reliability of the system is high. Proper deployment of the number and structure of the client-server system should be done so that it satisfies the expectation.

- To fulfill the users need in terms of information

- To study and investigate the current Web information system, implementation together with developing tools and their relevance to the project. This also includes the statistical software that would be chosen to analyze the datasets.

- To let the public know about the diseases that comes from genetically and non-genetically so that they could take extra attention on their health.

1.4 SIGNIFICANCE

In electronic environment, we are moving towards medical world in which the public user will have the possibility to access in a transparent way where they could find out about the diseases that might affect them in past, present or future.

By using the manual system to find our needed resource, process or administrate the medical records will not be that sufficient because they require energy, time and spaces to handle the constantly expanding volume of data. The quality and outcome of public user and administrator user also can be improved through the web-based system that ensures the accurate, relevant, structured and timely information is made accessible to the users at all level. Besides, the web technology has been growing aggressively to play a significant role in the business world. Therefore, this project will be a considerable and important step to start off into creating a web-based medical center information system. It also helps the user to know better about diseases before they attend to any of the medical centers.

This project is significant, as a pioneer in Malaysia, to create that new administration way to provide information about the analyzed medical data. Further more this is the time for us to begin our own research and improve existing foreign methods and algorithms of viewing data and information within a database in a more practical and meaningful manner. This will definitely help to bring up the image of our local medical centers where the whole world could see our capabilities in Information Technologies through the Internet change the way Malaysian medical center strategy works into a better system.

CHAPTER 2: LITERATURE REVIEW

2.1 PURPOSE

Literature review is a background study about the information and knowledge gained to develop this project. It also helps the developers to make a comparison on the past projects and study the strength and weaknesses that the project has. So, the developers could come up with a better version of the past project or create a new improved project that follows the need or the requirement of the users.

Rather than that, it is done to get a better understanding on the development tools that can be applied and also a better knowledge on the methodologies used in the process of developing it. This attempt does not only stop until this project where it might be helpful in the future project or in the working world that applies the same system.

2.2 APPROACH

A lot of information needs to be gathered about the system itself to develop a system. This process involved in developing the system and the methodologies used to develop the system. This particular information can be gained from various sources.

Usually, each source will give different information and facts and it depends on how the search is being done. For example, there are a number of ways of gathering information from the users. One way is through interviews. Another is to use questionnaires. The third is through observation of user activities and behavior.

If Internet is being used to find information, each keyword or phrase that is being searched will yield various sites, which is totally different from one site to another. Information can be obtained from system users, computer programs, procedure manuals and report, forms and documents and also from Internet.

Computer programs can be used to determine the details of data structures or process. Manual procedure specifies user activities in a business process. They can be used to determine detailed user activities, which is important in detailed system design. Reports indicate the kinds of outputs needed by users.

Forms and documents are useful sources of information about system data flows and transactions. If these sources are being used, the most recent and relevant forms and documents must be obtained and examined.

Besides that, the Internet was surfed and information was gathered from various sited on web sites of some local and foreign country medical center

methodologies for system development and information on development tools. Information was also gathered for software that will be used in this project such as Stata, Flash and others. The search engines that were used to gather all this information include MSN search, Yahoo search, AltaVista search, Google search and etc. So, as a conclusion most of the findings for this project came from sites on the Internet that helps a lot in giving information and knowledge.

2.3 FINDINGS

2.3.1 Relational Database

The concept behind a database is simple where it is like a file cabinet. Database stores information, just as file cabinet stores information. A specific application related to a set of information that is called database.

As for the relational database, Oracle, Informix, Microsoft Access 97 and Microsoft SQL Server 7.0 were studied and below is the information about it :-

1. Oracle

Oracle is the most popular database management system or rather relational database management system.The advantages of Oracle's architecture are:

- Large Databases

Supports the largest of databases, potentially terabytes in size.

- Many Concurrent users

Oracle supports large number of concurrent users executing variety of database applications operating on same data. It minimizes data contention and guarantees data concurrency.

- High Transaction Rates

It has the fast processing performance.

- High Availability

At most production sites, Oracle works high hours (24 hours) per day with no down time.

- Controlled Availability

An administrator can disallow use of a specific application so that the

application's data can be reloaded, without affecting other applications,

- Manageable Security

Provides fail-safe security features to limit and monitor data access.

-Database Integrity

Oracle enforces data integrity, "business miles" that dictate the standards for accepting data.

- Distributed Systems

Oracle combines the data physically located on different computers into one logical database that can be accessed by all network users.

- Portability

Oracle software is ported to work under different operating systems. Application developers for Oracle can be ported to any Operating systems with little or no modifications.

- Connectibility

Oracle Software allows different types of computers and operating systems to share information across networks.

- Replicated Environments

Oracle software allows you to replicate objects to multiple sites.Oracle supports replication both at data and schema level.

- Read consistency

As supported by Oracle, does the Following Read consistency as supported by Oracle, does the following :-

*Guarantees that the set of data seen by a statement Is consistent with respect to a single point-in-time, and do not change during transaction execution.

*Ensures that readers of the database data do not wait for readers of the same data. This means two users can read the same set of data, without waiting on each other.

*Ensure that writers for a database data do not wait for readers of the same data. This means persons can update data regardless of the other person reading the same data.

*Ensure that writers only wait for other writers if they attempt to update identical-rows in concurrent transactions. This means suppose a user is changing (updating) a record, say for example changing a employee number 100 to 500, another user should have to wait to change information on that row.

2. Informix

Informix-4GL is a 4th generation developed by the Informix Software, Inc. and specially designed for database applications and it also represents the latest advancement in programming and is designed for a particular class of applications. Not like PASCAL or C (general language) Informix are less complex and it is more closely approximate nature language so it is easy to use or apply it. As Informix 4-GL focus on a specific type of application, such a language can anticipate what user of this language wants to accomplish in their programs. Informix 4-GL is very powerful as well whereby a single simple statement generates a great deal of machine codes. As a result, programs written in Informix 4-GL do not contain nearly as many statements as programs written in a general language. Basically, some advantages of fourth-generation languages like Infonnix-4GL are:

a) They are simple, which speeds up the process of building and maintaining applications.

b) They are generally interactive, which simplifies the debugging process.

c) They appeal to a wide audience because they require no special training.

d) They resulting applications are easy to use and can solve problems efficiently.

As a summary Informix is database language that can be used to store, retrieve, update and delete information.

3. Microsoft Access 97

One of the best and fastest selling relational database packages for Windows on the market Microsoft Access 97. Access comes in two different modes. The first one is an easy to use menu driven interface that lets you issue commands with only a basic knowledge of Access. The second mode is program mode that lets the users to store instructions in a Visual Basic program file and execute them with one particular command.

Access has existed in five main versions. In the context of Access, a database can be viewed as a large repository in which tables, reports, queries, forms and other objects are stored inside.

Access allows the user to indicate how tables should be related to each other. A table can have a one-to-one, one-to-many or many-to-many relationship. A table that has referential integrity allows only one parent record for each child record. Access allows the user to make changes to the structure of a database table. User can add, delete and rearrange fields in the table structure. Users can also control how data will be entered in a table using the Properties sheet of a field.

As a database system Access has many good points and many bad points. Happily most of the bad points relate to the 'class' of application that Access is. It is important to clarify the 'class' that Access falls into. Access is a desktop database package and it has the first perceived bad point where it does not provide good performance when run across a network and when a lot of people using it at same time. However, if one considers what Access is designed as a desktop database with limited multi-user capabilities where the performance are good. In addition,

Access can and does make a good front-end package larger engine such as Oracle and SQL Server.

Against other desktop database packages Access has a bigger advantage where it is likely if you are running Windows as your operating system and using Microsoft Office as your application base, Access integrates well with these packages and data transfer between Access and the other Office components is relatively easy, In addition, against the other desktop databases Access is both rich in features and powerful, Access 2000 would be your choice to buy it.

4. Microsoft SQL Server 7.0

SQL is a special-purpose, nonprocedural language that could supports the manipulation, defination and control of data in relational database management systems. It's a special-purpose language, because we can use it only for handling databases and we can't write general-purpose applications with it. SQL is also known as a data sub language because to write an application, we have to embed SQL in some other language, and it's frequently used that way. A sub language can be used with application languages. A full-featured application language applicable because it usually includes semantics for procedures, where else SQL is nonprocedural. It only specifies what should be done so as conclusion SQL is concerned with results rather than procedures.

The most important feature of SQL is that it provides access to relational databases where it is so fundamental to SQL that many people think the terms SQL database and relational database are synonymous. But it is not like that and further more it doesn't even mention the term relation.

It makes giant strides in performance, reliability, and scalability, giving the organization many opportunities to create intelligent, real world business solutions. These are the following innovation that the SQL server had made:

a) Scalable from laptop to multiprocessor cluster

b) Dynamic row-level locking

c) Dynamic Self-Management

d) Wide array of replication options

e) SQL Server Desktop

f) Integration with Microsoft Office 2000 and Microsoft Visual Studio

These innovations, plus many more changes, make SQL Server 7.0 highly scalable and excellent for data warehousing. In addition, organizations that also run Office 2000 can take advantages of new ways that Office and SQL Server work together.

2.3.2 STATISTICAL SOFTWARE

These are the statistical software's that are available in the market: -

- **StatView** statistics package and **SuperANOVA** software for Mac.
- The **PROPHET system** is a Win32-based software package for data analysis, graphs, statistical analyses, mathematical modeling, and sequence analyses, formerly developed by BBN Technologies.
- Analyse-it, Powerful accurate statistical analysis for Microsoft Excel.
- Aptech, Systems a mathematical and statistical software package including powerful matrix programming language, for DOS, Windows, OS/2 and Unix platforms.
- BMDP Statistical Software, **BMDP** system, BMDP/DIAMOND, **SOLO**, SOLO power analysis, **StatXact**, now owned by SPSS Inc.
- Conceptual Software, **DBMS/COPY** and **DBMS/Engines**, database conversion tools for a lot of statistical software packages.
- Cytel software, Statistical software for exact inference like **StatXact**, **Proc-StatXact** for SAS, **LogXact** and **EaSt**.
- Data Description, **Data Desk**: visual data analysis software for Apple MacIntosh and Power-Mac.
- ECHIP, **Experimentation by Design**, DOE software and training for engineers and scienntists.
- **ForecastX** is an ActiveX component for statistical forecasting and analysis.
- GraphPad **Prism** software for Windows.

SYSTEM FOR ANALYZING MEDICAL DATA

- Idea Works, Various expert systems software, including **Statistical Navigator** for DOS/Win.
- Ivation Datasystems**, Beyond 20/20** software for Windows.
- MathSoft, Software products: **MathCAD**, **AXUM**, **S-Plus**.
- MathWorks**, MATLAB** software.
- Minitab software, FAQs.
- NAG - The Numerical Algorithms GroupNumerical software and libraries, Fortran 90, GenStat, Glim, MLP, AXIOM, Iris Explorer.
- NCSS Statistical Software - Number Cruncher Statistical System.
- NWP Associates**, STATLETS** - Java applets for statistical analysis and graphics. .
- Process Builder**, STRATEGY for Windows**, a Design of Experiment software for 32bit Windows
- **SAS** - SAS system on many platforms, JMP, C compiler.
- **STATGRAPHICS** - statistical analysis package for MS Windows.
- **SHAZAM**

  - SHAZAM software package, available for a lot of PC, workstation and mainframe platforms.
  - Independent newsgroup available,

- **SPSS**

  - SPSS software, meanwhile owner of former **Systat** and **BMDP** products.
  - Independent newsgroup available.

SYSTEM FOR ANALYZING MEDICAL DATA

- **Stata** statistics software package for DOS/Win/Mac/Unix.
- Stat-Ease**, Design-Ease** and **Design-Expert** for Windows, software for Design of Experiments.
- Statistical Solutions, Distributor of statistical software, including **Solas** for missing data analysis and **nQuery Advisor** for planning research studies .
- **S-Plus** software.
- **Statistica**, statistics software package for DOS/Win/Mac.
- **Stat/Transfer**, utility to transfer data between different programs.
- **Statit** software on various UNIX platforms.
- UNISTAT Statistical Package, Windows software with add-in features to MS-Excel or MS-Office.
- Software products: **PV-WAVE**, **IMSL** (C and Fortran numerical and graphical libraries), **Stanford Graphics**.

Here is a brief explanation about few selected statistical software: -

1. Stata Statistical Software

Stata is a statistics/data analysis program, which runs on a variety of platforms. It is relatively easy to learn and estimates a range of econometric models.

Stata can be used to enter commands interactively or in "batch" mode. Commands can be executed one at a time at the Stata prompt, or groups of commands can be entered in do-files and automatically executed in sequence.

The most current release, Stata 5.0, is available on GRACIE. Stata may also be installed on departmental PCs running Windows 3.x or Windows 95.

SYSTEM FOR ANALYZING MEDICAL DATA

The statement below describes how to use Stata 5.0 on GRACIE, the UNIX research server at CBS.

How to Run Stata in Interactive Mode

The following command are issued at the UNIX prompt to invoke Stata in interactive mode:

You should see a Stata welcome message along with some licensing and usage notes. Your cursor will then be positioned next
to the Stata "dot" prompt, where you enter Stata commands.

Use the Stata "exit" command to terminate your interactive session.

This destroys all data in the current session. You may see the following message if you have data in the current session that has not been saved.

no; data in memory would be lost

r(4);

If you want to preserve the data with which you have been working, then you must save the data before you exit. If you want to end your Stata session without saving your data, then issue the "clear" option as part of the "exit" command:

exit, clear

This will end your Stata session and destroy any data that has not been saved.

How to Run Stata in Batch Mode

Batch processing is generally used with Stata do files or programs. Issue the following command at the UNIX prompt to execute a Stata do file as a background process:

stata -b do [filename] &

where filename is the name of your Stata do file. You can also allocate additional memory to your Stata job using the "-k" option. For example, issue the following command to allocate 4000K of memory:

stata -k4000 -b do [filename] &

Stata for windows

Stata for Windows comes in two flavors;

a) Intercooled Stata

- Can analyze datasets with as many as 2,047 variables, and the only limit on observations is the amount of RAM on your computer.
- Computer should have at least 8 megabytes RAM

b) Small Stata

- All the features of Intercooled Stata, but can only process datasets of around 1,000 observations and 99 variables
- Computer should have 8 megabytes of RAM
- Coprocessor optional

Here is some of the stata capabilities:_

a) Cluster analysis

□Hierarchical clustering

-single linkage

-complete linkage

-average linkage

□Nonhierarchical

-kmeans

-kmedians

□Similarity/dissimilarity measures for continuous data

-L2/Euclidean

-L1/absolute/cityblock/manhattan

-L(#)

-Canberra

-correlation

-angular

□Similarity/dissimilarity measures for binary data

-matching

-Jaccard

SYSTEM FOR ANALYZING MEDICAL DATA

-Russell

-Hamman

-Dice

-antidice

-Sneath

-Rogers

-Ochiai

-Yule

-Anderberg

-Kulczynski

-Gower2

-Pearson

□Dendrograms

-full trees

-sub trees

-upper-portion of tree

□Support tools

-generate summary and grouping variables

-attach notes to analyses

□Result management utilities

-dir

-list

-drop

-use

-rename

□User extensible

-users can add new clustering methods and utilities

-full set of tools to ease making additions

b) Data management

□Create Stata datasets

-input data from command line

-input data saved from spreadsheets

-read data using a dictionary

-read any format ASCII data

-convert datasets directly from other statistical packages, spreadsheets and, databases using third-party software

□Built-in spreadsheet editor

-for Windows, Macintosh, and Unix

SYSTEM FOR ANALYZING MEDICAL DATA

□Data management functions

-mathematical and statistical functions

-string functions

-date functions

-time-series operators

-random number generator

-matrix functions

□Data reorganization

-row-column transposition

-data reshaping

-stacking of variables

□Labels

-dataset labels

-variable labels

-value labels (e.g., Male and Female for 0 and 1)

□Notes

-extensive notes can be attached to a dataset

□Sorting

-ascending or descending sorts

-multiple-key sorts

-numeric and string sorts

□Merge datasets

-paired merge datasets

-key merge datasets

-join datasets

-outer joint

-append datasets

□Special datasets

-panel data/cross-sectional time-series

-time to event data

□Utilities

-compress (make dataset as small as possible without loss of accuracy)

-formatted and unformatted disk I/O

□Variable management

-generate new variables

-replace existing variables

-encode and decode string variables

□Variable types

-byte

-integer (int)

-long

-float

-double

-string

-dates

c) Tools for epidemiologists

□Standardization of rates

- direct standardization

- indirect standardization

□Tables for epidemiologists

- 2 x 2 and 2 x 2 stratified tables for longitudinal, cohort study, case-control, and matched case-control data

- odds ratio, incidence ratio, risk ratio, risk difference, and attributable fraction

- confidence intervals for the above

- chi-squared, Fisher's exact, and Mantel–Haenszel tests

- tests for homogeneity

- choice of weights for stratified tables:Mantel–Haenszel, standardized, or user-specified

- exact McNemar test for matched case-control data

- tabulated odds and odds ratios

- score test for linear trend

□Table symmetry and marginal homogeneity tests

- n x n tables where there is a one-to-one matching of cases and controls

- asymptotic symmetry and marginal homogeneity tests

-exact symmetry tests

-transmission disequilibrium test (TDT)

SYSTEM FOR ANALYZING MEDICAL DATA

□Receiver Operating Characteristic (ROC) curves

-calculate area under curve

-estimate maximum-likelihood ROC models assuming a binormal distribution of the latent variable

-test equality of two or more ROC areas

-test equality of ROC area against a "gold" standard ROC curve

-raw and Bonferroni adjusted significance probability

-Sidak's adjustment for multiple comparisons

□Generalized linear models for the binomial family

-individual-level or grouped data

-odds ratios, risk ratios, health ratios, and risk differences

□Pharmacokinetics

-pharmacokinetic measures from time-and-concentration subject-level data

-test that measurement is normally distributed

-analyze data from crossover design experiment

-bioequivalence testing for two treatments

□ICD-9-CM diagnostic and procedure codes

-verify variable contains valid codes

-modify codes to display in standard format

-display description of codes

-display identity of source

d)Generalized linear models

□Link functions

-identity

-log

-logit

-probit

-complementary log-log

-odds power

-power

-negative binomial

-log-log

-log-compliment

□Families

-Gaussian (normal)

-inverse Gaussian

-Bernoulli/binomial

-Poisson

-negative binomial

-gamma

-power

SYSTEM FOR ANALYZING MEDICAL DATA

- Choice of estimation method
  - -maximum likelihood
  - -iteratively reweighted least squares (IRLS)
- Choice of variance estimates and standard errors
  - -inverse Hessian
  - -outer product of the gradients (OPG)
  - -observed information matrix
  - -expected information matrix
  - -robust sandwich/huber/white estimator
  - -robust variance with clustered/correlated data
  - -Newey-West heteroskedastic and autocorrelation consistent (HAC)
  - -jackknife
  - -bootstrap
- User extensible
  - -user-defined link functions
  - -user-defined variance functions
  - -user-defined HAC kernels
- Linear constraints
- Predicts
  - -expected value of dependent variable
  - -Anscombe residual
  - -Cook's distance
  - -deviance residual
  - -diagonal of hat matrix

-likelihood residual

-Pearson residual

-response residual

-score residual

-working residual

□Marginal effects

-marginal effects and elasticities

-standard errors and confidence intervals

-computed at means or specified covariate values

-computed for any predicted statistic

2. The Number Cruncher Statistical System (NCSS).

Number Cruncher Statistical System is a comprehensive and accurate, easy to learn, statistical and data analysis system for Windows

For the past few years, NCSS has been fine-tuning the art of number crunching. Lately , NCSS offers you the latest technology in statistical analysis in a user-friendly format that easily imports and exports all major spreadsheets, databases, and statistical file formats. And this is yours at a price that is about half that of the competition's.

3. STATISTICA

Statistica is not just "another advanced statistics package." It offers not only the speed and capacity to handle datasets/designs of practically unlimited size and unusual comprehensiveness of its procedures - fully integrated with highest quality graphics (that won for Statistica the name of "the King of data visualization tools. Statistica offers COM architecture, high-end technologies. Statistica is based on the COM architecture and high-end technologies, that are usually not found in such "vertical market" applications as data analysis software. As a result, Statistica offers unique functionality and usability features that currently no other competing product can offer. The benefit of using Statistica are:-

a) *For advanced users: Power, Scalability, Compatibility.*

The powerful and unique - in data analysis software - full implementation of the COM architecture and the fully integrated Visual Basic make it a perfect foundation (or a component) of global computing infrastructures (such as the Internet Information Delivery Systems or large, multi-user enterprise installations).

It also offers one of the largest and richest development environments available in the entire software industry, with more than 10,000 data analysis and graphics functions directly exposed to end users and developers. These features also make it a perfect tool to tackle the most demanding problems in data analysis, data mining, or QC/SPC applications.

b) *For occasional users and novices: Simplicity, Customizability, Quickness, and Quality.*

The unique technologies of Statistica are also offering a lot to a desktop user who may not want to know what Visual Basic is or what the "enterprise business intelligence system" is. For example:

- Because of these technologies, the Statistica desktop products are uniquely user friendly and flexible; they can also be extremely "simple" (e.g., run by anyone from an Internet browser).
- Every action can be recorded (in the background) into a reusable, modifiable macro with a click of the mouse, and then instantly assigned to a toolbar button (the macro will have the industry standard Visual Basic format, but you do not even need to know that).
- Every aspect of the user interface can be adjusted to your needs by dragging controls with the mouse. Countless program options can be as hidden or as exposed as you want.

c) *Quality and comprehensiveness.*

In addition to all these benefits - no other application can match the quality of implementation of every detail of Statistica - its graphics, responsiveness/speed, elegance, and built-in intelligence.

Every aspect of it is designed to offer the ultimate level of functionality and a large part of that functionality normally can be found only in designated, specialized applications, if they existed at all, and those that do - would not be integrated and are usually not as comprehensive and well designed. For example:

- The graphics engine, a part of every product, offers more choices and options than any designated graphics package on the market.
- The Statistica Query facility (allowing you to access external databases), just one of many parts of every product, includes features, power, and performance that are offered only in designated, expensive database querying products.

The Statistica Visual Basic language, offers one of the largest and richest development environments available in the industry (with more than 10,000 data analysis and graphics functions, depending on the version, and professional programming tools).

2.3.3 SOFTWARE DEVELOPMENT TOOLS.

1) Active Server Page

Active Server Page (ASP) contains a set of instruction that are processed by IIS on the Web server that can make dynamic or larger data possible in the application. In most of the cases, the ASP codes returns the results of the server side processing by generating a HTML page. The instruction can also manipulate any number of the server side Active-X components that are typically used to access data. The combination of scripting language and both methods above makes the ASP as a high power technology for building superb Web sites.

It is so useful because it is a browser independent technology where all the browser applicable with it such as Netscape Navigator, Internet Explorer or NCSA Mosaic. If there are specific HTML differences among the different Web browser that we would like to exploit in the Web application , we will be able to obtain information about the browser type from IIS and generate a different HTML page for each browser type.

ASP has it's own life cycle where it could be broken into three phases:-

a) Request

This phase represents the beginner spot of contact between the user's web browser and the ASP file. The user activates the ASP code by browsing into the files that contain code. The pioneer user visits the ASP application where IIS creates an object that can be used within the application. If the user has already visit another file in that application, IIS will update the location of the user existing in the object. After the object has begun, IIS starts executing the code in the ASP file.

b) processing

This phase begins once IIS has started the location and request objects. IIS executes the ASP codes and interacts with various databases according to the instructions and also generate the HTML page from ActiveX data that will be sent back to the user's Web browser.

c) response

This phase starts after the ASP scripting executed, IIS again goes to the HTML page to move back to the user's Web browser at this location.

The aspects that ASP brings to the developer are access to data stored in relational databases, encapsulated behind the ActiveX Data Objects object model.

ASP can be used with the Microsoft Peer Web Services version 3.0 on Window NT Workstation, Microsoft Personel Web Server on Window 95&98, Microsoft Internet Information Server version 3.0 on Window NT Server.

2) *JavaScript*

It is scripting language developed by Netscape. However, Netscape collaborate with Sun during the development of LiveScript, changed its name to Javascript, and borrowed several structures from Java's syntax.

JavaScript is relatively simpler and less complicated than a real programming language, but still powerful enough to applied in an effectively. Many of the statements are remarkably natural in their structure and terminology. JavaScript supports distributed processing by enabling various tasks to be performed on the client rather than on the server. And it is also ideal for validating user input and making sure that the data entered is valid. It also can easily manipulate various browser objects and used to conserve bandwidth. The popular browsers are Netscape Navigator and Internet Explorer that supports JavaScript for multi purpose such as setting up a homepage and others.

3) *Visual Basic Script*

Microsoft Visual Basic has a subset that called as VBScript and it is also can be applied with Visual Basic for Application. VBScript capabilities for Web browsers are the ability to provide scripting, automation and customization.

It brings the scripting to a wide variety of environment, including Web client scripting in Microsoft Internet Explorer version 3.0 and in Microsoft Internet Information server version 3.0. VBScript acts to link application using ActiveX Scripting. With that, browsers and other link applications doesn't require special integration code for each scripting component. Only Internet Explorer supports VBScript later it might be supported by the Navigator with the help of a proprietary add-in.

2.3.4 Development models

1) *Waterfall model*

The stages in this model are depicted as cascading from one to anotherand each stages will be completed fully before the next stage begin. If all the consumers requirement are understudied, analyzed for completeness and consistency, and documented as requirements document, then the developer can go on to system designing activities. This model also perform a very dynamic level view of what goes on during the development duration and it suggests the sequence or the way of event they should come out with.

The advantages of this model are:-

a) very straight forward and simple

b) a lot of reviews and evolution

c) a lot of documentation after each step

d) you will have some output after each step

The disadvantages of this models are :-

a) it is very linear and must follow in order (stages by stages)

b) really need a good requirement to use it if it is not thoroughly specified.

2) *Prototyping Model*

A prototype is a partially developed product that enables customers and developers to examine some aspect of proposed system and decide if it is suitable or appropriate for the finished product. In other words, prototyping means building a small version of a system, usually with limited functionality, which can be used to help the user or customer identify the key requirements of a system and demonstrate feasibility of a design or approach.

a) Identify the user's basic requirement.

The system designer (usually in information systems specialist) works with the user only long enough to capture the user's basic information needs.

a) Develop a working prototype.

The system designer creates a working quickly. The prototype may only perform the most important function of the proposed system, or it may consist of the entire system with a restricted file.

b) Use the prototype.

The user is encouraged to work with the system to determine how well the prototype meets his or her needs and to make suggestions for improving the prototype

c) Revise and enhance the prototype

The system builder notes all changes requested by the user and refines the Prototype accordingly. After the prototype has been revised, the cycle returns to stage 3. Stage3 and 4 are repeated until the user is satisfied.

Prototyping is often used to design a good user interface: the part of the system with which the user interacts. Since the prototyping model allows all or part of system to be constructed quickly to understand or clarify issues, it has the same objective as engineering prototype, where requirements or design require repeated investigation to ensure that the developer, user and the customer have a common understanding both of what is needed and what is proposed. One or more of the loops for prototyping requirements, design or the system may be eliminated, depending on the goals of the prototyping. Application prototyping has two primary uses. On one hand, it is an effective device for clarifying user requirements. Written specifications are typically created as a vehicle for describing application features and the requirements that must be met. Developing and actually using a prototype can be a very effective way of identifying and clarifying the requirements an application must meet.

A second use of application prototyping is to verify the feasibility of a system design. Analyst can experiment with different application characteristics, evaluating user reaction and response.

2.3.5 Operating System

1) Microsoft Windows 2000 Server

Microsoft Windows 2000 is one of the leading operating systems in the Internet and Intranet world. This popularity achieved by the Microsoft skills in placing Windows 2000 Server as a useful corporate solution, with a variety of dynamic tools and the ability to standardize on both development and deployment in Windows 2000 server platform. In the other hand, it also works smoothly and pretty well as an Internet platform, especially on small scale or fragmentizes the internet level in a departmental size.

The advantages in Windows 2000 Server are it is stable, huge variety of functional tools, less bugs and it has a better security system than the other version of Microsoft Windows. It is also supports virtual domains and has the ability to delegate administration to other users. In the management view, there's a Window based management console, browser based administration and common line scripting. In the security view, it features user authorization such as username and password, as configured by the system administrator and editing capabilities to meet auditing guidelines.

2.3.6 Web Browser

1) Internet Explorer

Internet Explorer is tool to explore and access or browse information and data on the Web. It also provides a very helpful and dynamic toolbars of certain particular functions and commands to mange the browser effectively. To reach a particular Web page, you only have to type in the homepage address in the address bar and click search that tells the search engine to browse to the page you interested in. You can also go to a new page by clicking the hyperlink icons that navigate to the new page you wanted. Internet Explorer also available in Windows NT, 95,98 and 2000.

CHAPTER 3: METHODOLOGY

3.1 PROPOSED SYSTEM DEVELOPMENT METHODOLOGY

The best system development methodology for this project is the combination of prototyping and waterfall model.

The waterfall model has a very high level view of what is going on during system development, and the way of events that are expected to encounter. Each stages has to be completed before a new stage begins so each stage mustn't be to complicated so that the system could run faster.

The developer doesn't know all the key aspects that affect the required output. He might have to trash from one activity to another and then go back to make modification such as adding, deleting and editing. To avoid this problem, prototyping model is used where it is known as a sub-process. It is a partially developed object that enables the users to examine some aspects of the proposed system.

Rather than that, it also helps to access alternative design strategies, useful for verification and validation so that the functions works correctly and systematically.

3.2 PROPOSED STATISTICAL SOFTWARE METHODOLOGY

In this project, STATA statistical software is chosen because :-

1. Stata provides a broad range of statistics
2. Stata is intended for researchers of all disciplines.
3. Stata has complete data-management capabilities. Stata is not just a statistics package. It is a full data-management system with statistical capabilities.
4. Stata is easy to use.
5. Stata is fast.
6. Stata is accurate.
7. Stata is available for Windows, Macintosh, and Unix (including Linux) computers. Stata datasets, programs, etc. can be shared across platforms without translation.
8. Stata is so programmable that you can add new commands to it.
9. Stata has Internet capabilities. New features can be installed over the Internet with a single click. Official updates can be installed over the Internet with a single click.
10. Analyses can be documented and reproduced.
11. Stata provides commands to analyze panel data (cross-sectional time-series, longitudinal, repeated-measures, and correlated data).
12. Stata provides commands to analyze cross-sectional data.
13. Stata provides commands to analyze time-series data.
14. Stata provides commands to analyze survival-time data (duration, time-to-failure data).
15. Stata provides commands to analyze cohort study, case-control, and matched

case control data.

16.Stata provides commands to analyze count and binary data.

17.Stata provides commands to analyze complex survey data.

18.Stata provides byte, integer, long, float, double, and string variables.

19.Stata understands dates.

20.Stata has been year-2000 compatible since January 1995.

21.Stata has matrix operators.

22.Over half of the web site is devoted to supporting users of Stata.

23.Outstanding technical support.

□Stata is easy to use

Stata has simple, consistent command syntax. Here's how you do an OLS regression:

- reg mvalue gender age income educ1-educ6

Most commands have the same syntax -- whether the command estimates a model, produces descriptive statistics, or performs a data-management task:

- logit outcome gender status exp if age>39
- graph income educ if state=="Texas"
- drop if select > 10
- by gender: tab case exposure
- by agegrp: summarize income yrswk

Stata is fast

An OLS regression (reg command) takes less than 0.01 seconds with 1,000 observations and 10 covariates. Change the command to logit (maximum-likelihood logistic regression) and it still takes less than 0.30 seconds to estimate 11 coefficients (10 covariates) on 1,000 observations.

Increase the number of observations to 10,000 and the linear regression takes 0.10 second and the logistic regression 0.287 seconds.

Now increase the number of observations to 100,000. The OLS regression takes only 1.03 seconds and the logistic regression just 2.876 seconds.

Note: All timings were performed on a 350 MHz Pentium II running Intercooled Stata for Windows. Timings exclude the time to display the results -- which takes roughly 0.02 seconds but varies according to video card.

Data management is one of Stata's strengths

With a handful of basic commands, you can perform just about any data-management task or data transformation.

Here's how you match-merge two datasets:

- sort id
- save newdata
- use olddata, clear
- sort id
- merge id using newdata

Here's how you create a lagged variable:

- sort year

- gen ylag = y[_n-1]

Here's how you create a lagged variable for each subject's data:

- sort subject year

- by subject: gen ylag = y[_n-1]

The Windows and Macintosh versions of Stata have a spreadsheet editor, which you can use to enter, change, or view data. The editor also has a protected mode to prevent accidental changes to your data.

Stata is designed for researchers who must be able to document their analyses

Stata can be used interactively and in batch mode. Log files of interactive sections can be re-run as batch files. This makes it easy to duplicate analyses and to document fully your data-management steps. Log files even contain a record of changes done interactively in the spreadsheet editor.

3.3 PLATFORM CONSIDERATION

Windows 2000 Server has been designed from the upwards as an linked multipurpose operating system. It's a powerful platform for web application development and deployment. It's also a robust and secure operating system.

In this project, Windows 2000 Server was chosen because :-

a) easy installation procedures

b) various kind of development tools

c) high availability of Web publishing software

d) low administration cost

e) user friendly

3.4 PROPOSED DATABASE MANAGEMENT SYSTEM METHODOLOGY

In this project, SQL Server has been selected due to the well integrated with other Microsoft products and providing universal data access. It also has better distribution transaction support and recovery.

Further more, it also has a proven track record and wide support. SQL Server has the largest number of third party ,front end support product worldwide, from those designed strictly for application development to add in SQL Server access modules for standard PC-based DBMSs such as dBase and Paradox. There're even access modules that let the users query the database from the leading spreadsheet program.

3.5 PROPOSE FOR WEB SERVER METHODOLOGY

IIS has been chosen for the Web Server methodology where it has strong security features such as it only allows the administrator to control the password length, uniqueness and how often a password must be changed.

It also well integrated with Windows 2000 server as it is the chosen platform operating system for this project.

CHAPTER 4: SYSTEM DESIGN

4.1 INTRODUCTION

Important note: *The following scenario of the system, be it architecture or component concepts, is based on randomly picked diseases which would fill up the system and work as it is supposed to. This is due to the fact that as of 'print time', the reliable source was unable to deliver the required information from the health care center due to examination preparations etc. Therefore, the actual system which would house the actual top ten diseases will be developed once the information is received, using the same parameters.* ***Thus, the behavior of the related components and the specifications remain the same.***

Design is a creative process that gives explanation to the solutions of the problems occurs. It is also a meaningful engineering representation of something that is to be built. Design focuses on four major areas of concern:

a) architecture

b) data

c) interface

d) component

The main purpose of this phase is to transform or change all the requirements analyzed in the previous phase – System Analysis, into system characteristics.

4.2 ARCHITECTURAL DESIGN

Architectural design represents the structure of data and program components that are required to build the computer-based system. It also considers the architectural style that the system will take, the structure and properties of the component that constitute the system, and the relationship that occur among all architectural components of a system.

For this project, architectural design is based on modular decomposition approach. Decomposition is a structured system approach where designs are partitioned to a smaller parts that are called module or components. It is a top-down approach that based on assigning functions to components.

The developer or designer begins with a high level description or explanation of the functions that are to be implemented and builds lower-level explanations of how each component will be organized and related to other components. In this method or approach, the system development begins from a high level description and goes down to a lower level description.

The System for Analyzing Medical Data consist only two main modules and that are:

a) Top Ten Diseases Module

b) Authorized User Module

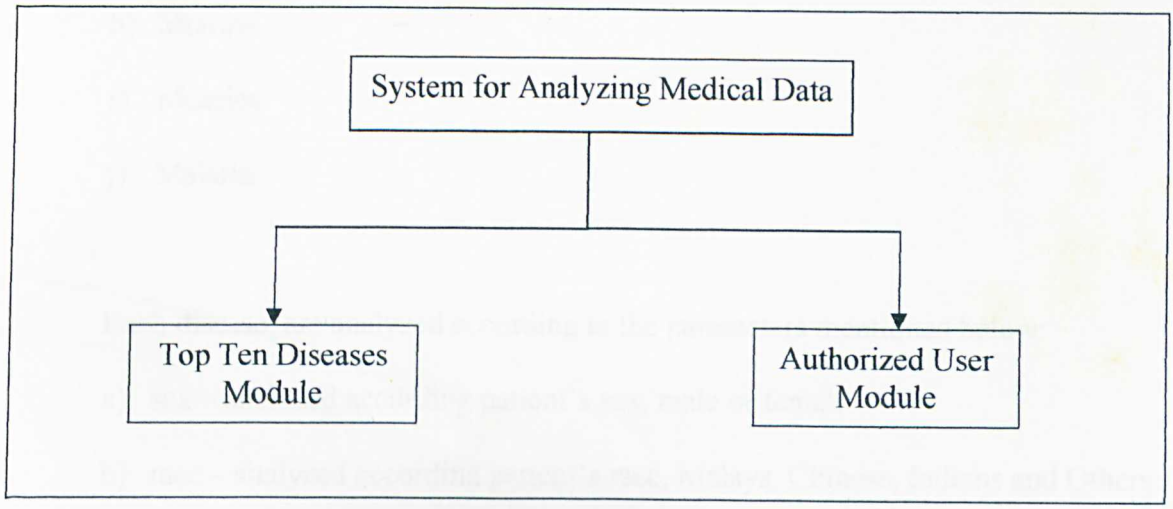


Figure 4.1: Top-down view of System for Analyzing Medical Data

4.2.1 Top Ten Diseases Module

Top Ten Diseases Module is one of the sub module for the System for Analyzing Medical Data. This module has top ten diseases in Malaysia or University Malaya Medical Center which lets the users to look up on the chosen disease. This module can be used by any user and it also contains general information about the selected disease. It also has a picture of the selected disease when this module is generated. The top ten, randomly picked diseases are:

a) Cancer

b) Diabetes

c) Herpes

d) Vericella

e) Cholera

f) Tuberculosis

g) Coxsackie's

h) Mumps

i) Measles

j) Malaria

Each disease, are analyzed according to the parameters mentioned below:

a) sex – analyzed according patient's sex, male or female

b) race – analyzed according patient's race, Malays, Chinese, Indians and Other

c) state – analyzed according patient's hometown, 14 states in Malaysia including Sabah and Sarawak

d) occupation – analyzed according patient's occupation, professionals (eg. lawyers, executives, engineers and others) and non-professionals(eg. operators, clerk, drivers and others).

e) Height – analyzed according patient's height, from 100cm to 200cm (each block is separated by 20cm)

f) Weight – analyzed according patient's weight, from 40kg to 90kg (each block is separated by 10 kg)

g) Age – analyzed according patient's age, from 10 years to 80 years (each block is separated by 10 years)

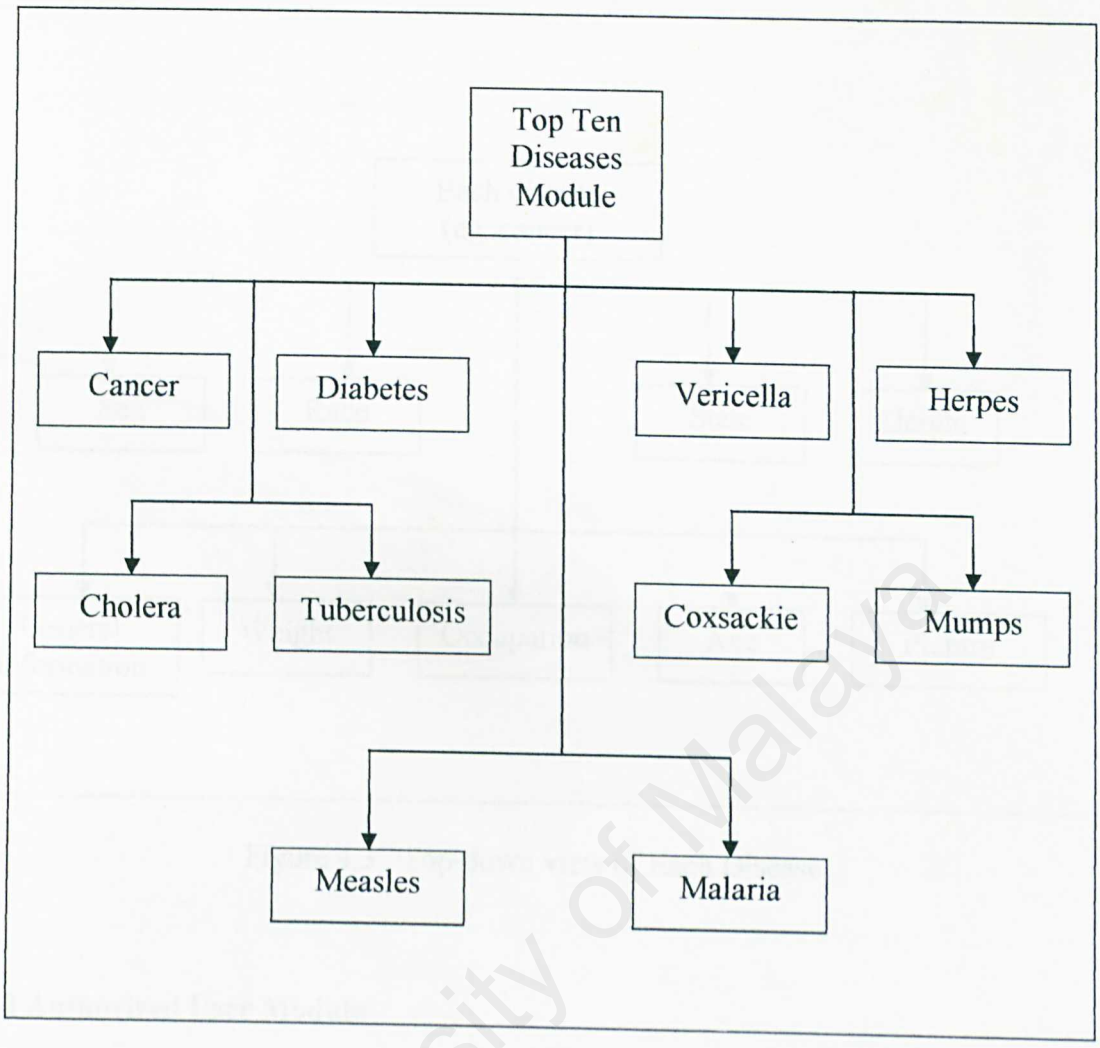


Figure 4.2 : Top-down view of Top Ten Diseases Module

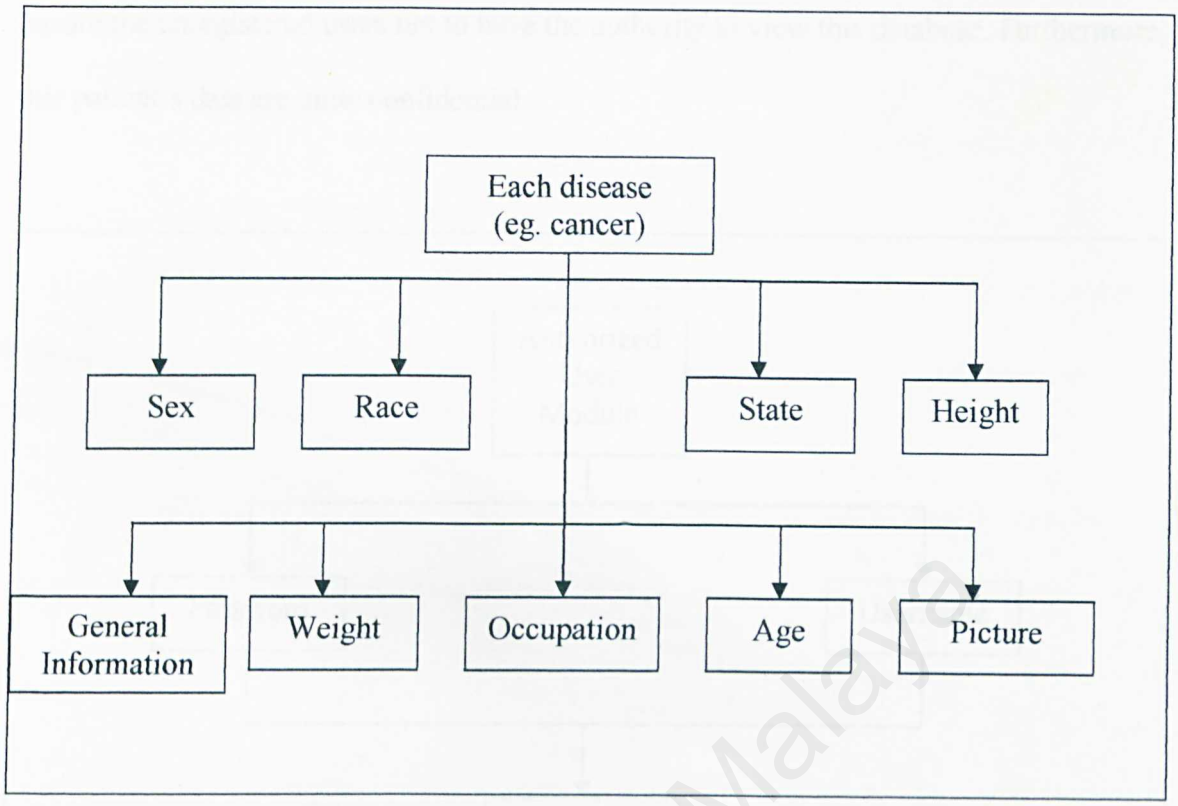


Figure 4.3 : Top-down view of Each Disease

4.2.2 Authorized User Module

The Authorized User Module is another sub module for the System for Analyzing Medical Data module. The Authorized User Module lets only the authorized admin to view this specific module to modify, edit, delete and add the data that has been stored in a data analyzing software called Stata. Stata analyses the data and generate new graph for a graphical view in percentage or figures.

The Authorized User Module is the module that provides functions for the System for Analyzing Medical Data database management in the Stata. The admin personnel has his own special password and username to run this software. This security makes or

ensure the unregistered users not to have the authority to view this database. Furthermore, this patient's data are quite confidential.

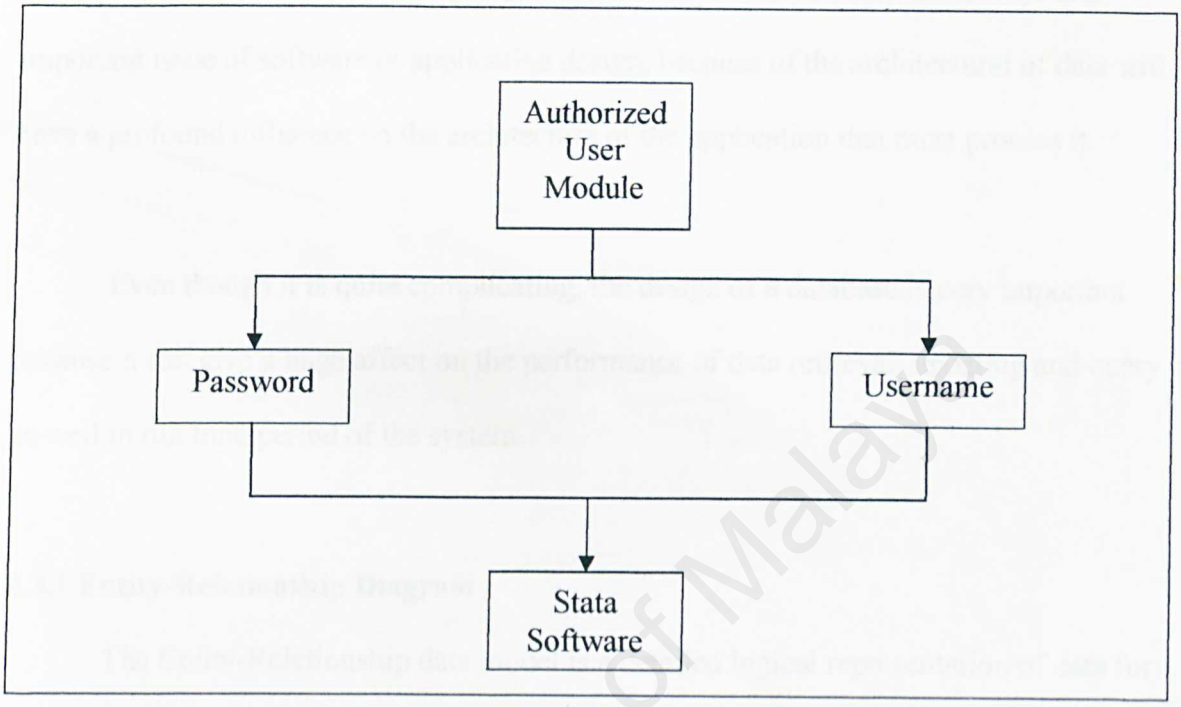


Figure 4.4 : Top down view of Authorized User Module

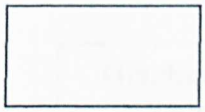
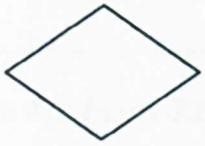
4.3 DATABASE DESIGN

To store data and the format of data type is often a vital decision making in the design of an information and analyzing system. The structure of data has always and important issue of software or application design, because of the architectural of data will have a profound influence on the architecture of the application that must process it.

Even though it is quite complicating, the design of a database is very important because it can give a huge affect on the performance of data retrieval, updating and query as well in run time period of the system.

4.3.1 Entity-Relationship Diagram

The Entity-Relationship data model is a detailed logical representation of data for an organization. The model is expressed in terms of entities in the business environment, but in this case the web application environment, the relationships between entities, the attributes of both entities and relationships are expressed. There are two types of entities used in Entity-Relationship diagrams and they are:

SYMBOLS	*DEFINATION*
	Real entity – a place, person, things or objects
	Something or anything created to join two entities

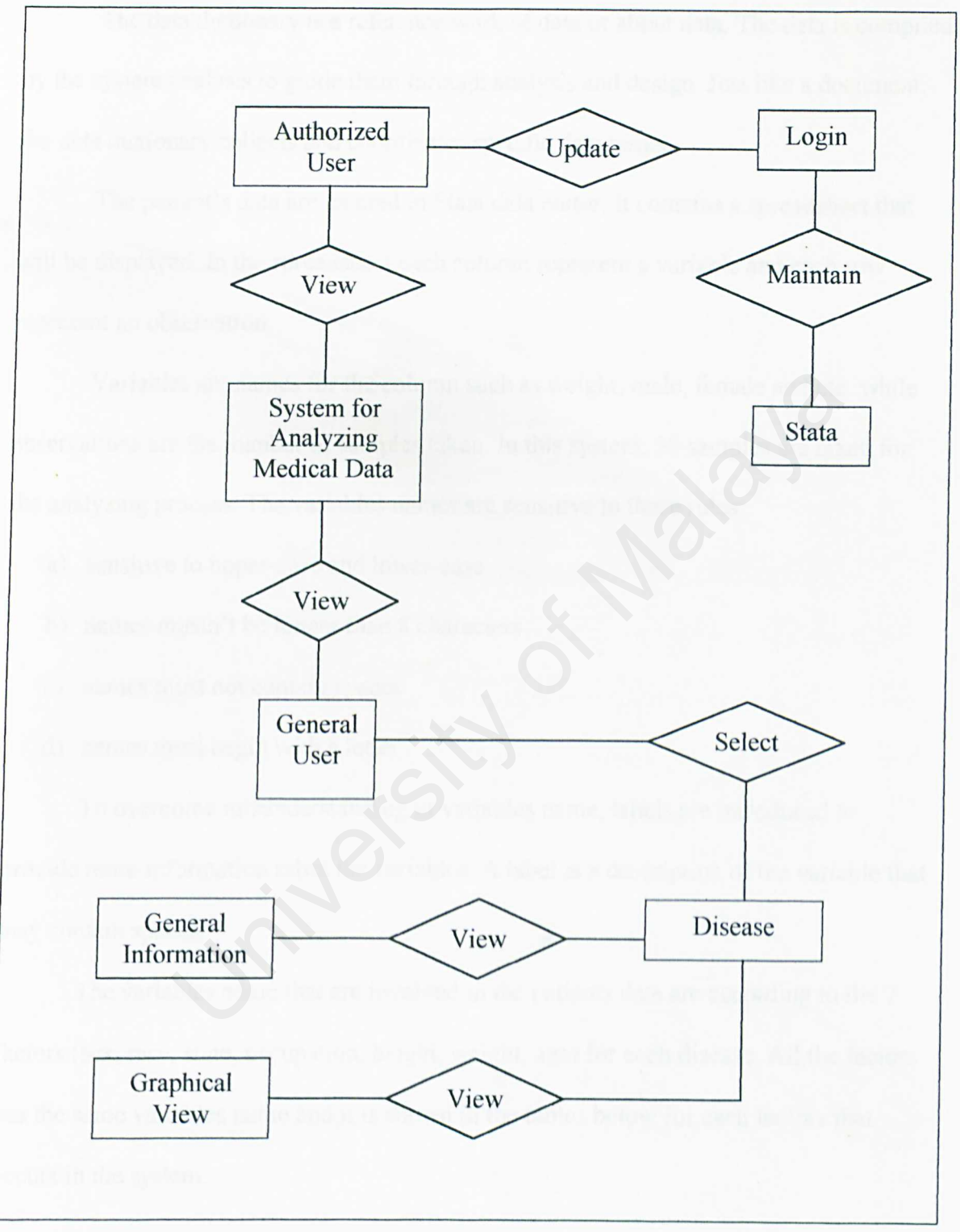


Figure 4.5: Entity-Relationship Diagram of System for Analyzing Medical Data

4.3.2 Data Dictionary

The data dictionary is a reference work of data or about data. The data is compiled by the system analysts to guide them through analysis and design. Just like a document, the data dictionary collects and coordinates specific data terms.

The patient's data are entered in Stata data editor. It contains a spreadsheet that will be displayed. In the spreadsheet each column represent a variable and each row represent an observation.

Variables are names for the column such as weight, male, female and etc. while observations are the number of samples taken. In this system, 50 samples are taken for the analyzing process. The variables names are sensitive to these rules:

a) sensitive to upper-case and lower-case
b) names mustn't be longer than 8 characters
c) names must not contain spaces
d) names must begin with a letter

To overcome misunderstanding in variables name, labels are introduced to provide more information sabot the variables. A label is a description of the variable that may contain spaces.

The variables name that are involved in the patients data are according to the 7 factors (sex, race, state, occupation, height, weight, age) for each disease. All the factors has the same variables name and it is shown in the tables below for each factors that occurs in the system.

4.3.2.1 Variables for Sex

This table shows the variables name of the patient's sex.

Variable Name	***Variable Label or Description***
sex1M	Store the number of **male** patients for the selected disease
sex2F	Store the number of **female** patients for the selected disease

4.3.2.2 Variables for Race

This table shows the variables name of the patient's race.

Variable Name	***Variable Label or Description***
race1m	Store the number of **Malay** patients for the selected disease
race2c	Store the number of **Chinese** patients for the selected disease
race3i	Store the number of **Indian** patients for the selected disease
race4o	Store the number of **Other race** patients for the selected disease

4.3.3.3 Variables for State

This table shows the variables name of the patient's hometown or living state.

Variable Name	***Variable Label or Description***
stat1KL	Store the number of **Kuala Lumpur** patients for the selected disease
stat2Slg	Store the number of **Selangor** patients for the selected disease

stat3Kdh	Store the number of **Kedah** patients for the selected disease
stat4Prk	Store the number of **Perak** patients for the selected disease
stat5Phg	Store the number of **Pahang** patients for the selected disease
stat6Mlk	Store the number of **Melaka** patients for the selected disease
stat7NS	Store the number of **Negeri Sembilan** patients for the selected disease
stat8Jhr	Store the number of **Johor** patients for the selected disease
stat9Kln	Store the number of **Kelantan** patients for the selected disease
stat10Tg	Store the number of **Terengganu** patients for the selected disease
stat11Ps	Store the number of **Perlis** patients for the selected disease
stat12PP	Store the number of **Pulau Pinang** patients for the selected disease
stat13Sk	Store the number of **Sarawak** patients for the selected disease
stat14Sh	Store the number of **Sabah** patients for the selected disease

4.3.3.4 Variables for Occupation

This table shows the variables name of the patient's occupation.

Variable Name	***Variable Label or Description***
Occu1	Store the number of **professional** patients for the selected disease
Occu2	Store the number of **non-professional** patients for the selected disease

4.3.3.5 Variables for Height

This table shows the variables name of the patient's height.

Variable Name	***Variable Label or Description***
H_100	Store the number of patients height, **below 100cm** for the selected disease
H100_119	Store the number of patients height, **100cm to 119cm** for the selected disease
H120_139	Store the number of patients height, **120cm to 139cm** for the selected disease
H140_159	Store the number of patients height, **140cm to 159cm** for the selected disease
H160_179	Store the number of patients height, **160cm to 179cm** for the selected disease
H180_199	Store the number of patients height, **180cm to 199cm** for the selected disease
H_200	Store the number of patients height, **above 200cm** for the selected disease

4.3.3.6 Variables for Weight

This table shows the variables name of the patient's weight.

Variable Name	***Variable Label or Description***
W_40	Store the number of patients weight, **below 40kg** for the selected disease
W40_49	Store the number of patients weight, **40kg to 49kg** for the selected disease
W50_59	Store the number of patients weight, **50kg to 59kg** for the selected disease
W60_69	Store the number of patients weight, **60kg to 69kg** for the selected disease
W70_79	Store the number of patients weight, **70kg to 79kg** for the selected disease
W80_89	Store the number of patients weight, **80kg to 89kg** for the selected disease
W_90	Store the number of patients weight, **above 90kg** for the selected disease

4.3.3.7 Variable for Age

This table shows the variables name of the patient's age.

Variable Name	***Variable Label or Description***
Age_10	Store the number of patients age, **below 10 years** for the selected disease
A10_19	Store the number of patients age, **10 to 19 years** for the selected disease
A20_29	Store the number of patients age, **20 to 29 years** for the selected disease
A30_39	Store the number of patients age, **30 to 39 years** for the selected disease
A40_49	Store the number of patients age, **40 to 49 years** for the selected disease
A50_59	Store the number of patients age, **50 to 59 years** for the selected disease
A60_69	Store the number of patients age, **60 to 69 years** for the selected disease
A70_79	Store the number of patients age, **70 to 79 years** for the selected disease
Age_80	Store the number of patients age, **above 10 years** for the selected disease

4.4 PROCESS DESIGN

4.4.1 Data Flow Diagrams

Data Flow Diagrams (DFD) is a technique used to show the graphical characterization of the data processes and flows in a system. The data flow diagrams gives us an overview of system inputs and outputs, processes and the flow of data through each process. Data flow is not the same as control flow. A process in a system shows how input is changed to output.

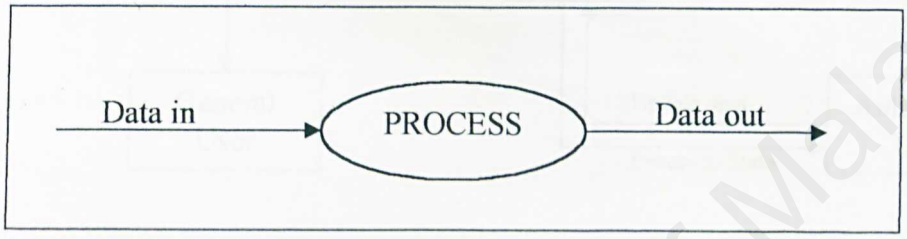
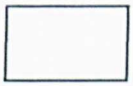
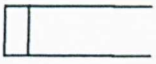


Figure 4.6 : Process changes input to output

The table below shows the basic symbols of a data flow diagram and its definition.

Symbol	*Definition*
	Transformations of data to other data
	Sources and destination of data
	Data in static storage
	Data on the move

SYSTEM FOR ANALYZING MEDICAL DATA

4.4.1.1 Context Diagram

This figure below shows context diagram for System for Analyzing Medical Data

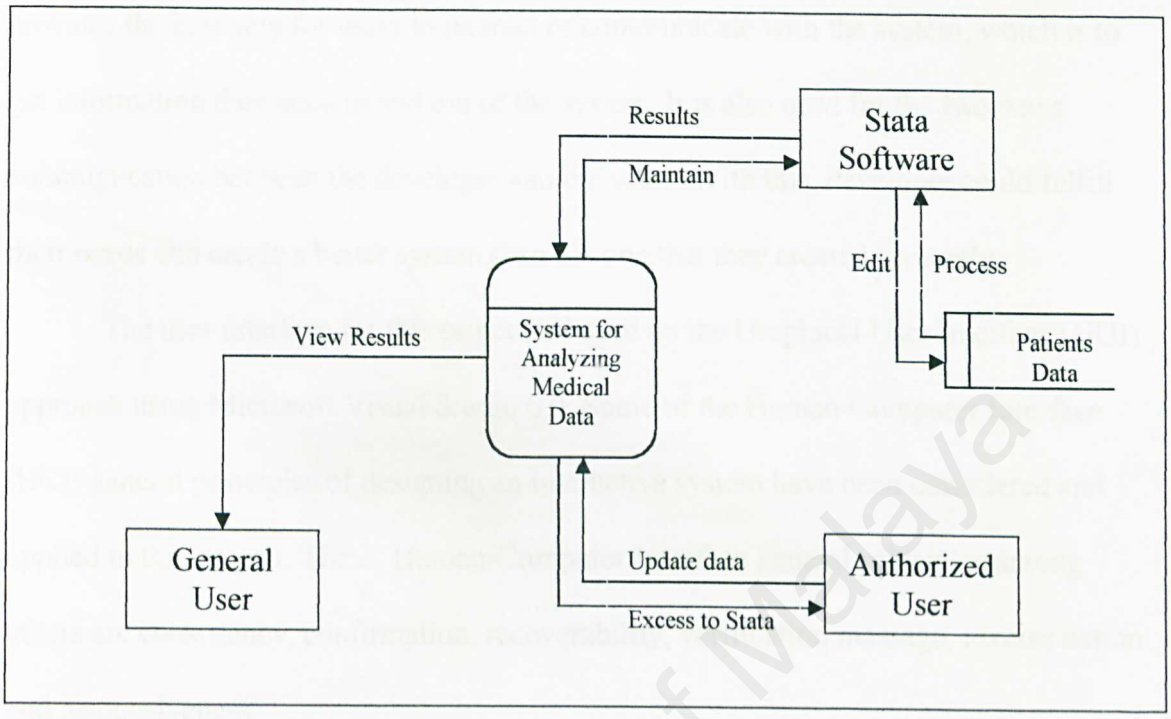


Figure 4.7 : Context Diagram for System for Analyzing Medical Data

4.5 USER INTERFACE DESIGN

The interface is a system for almost all users. Interface design is chosen because it provides the best way for users to interact or communicate with the system, which is to get information they need in and out of the system. It is also used for the two ways communication between the developer and the users. With this, developer could fulfill their needs and create a better system than the one that they created presently.

The user interface for this project is based on the Graphical User Interface (GUI) approach using Microsoft Visual Studio 6.0. Some of the Human-Computer Interface (HCI) general principles of designing an interactive system have been considered and applied in this project. These Human-Computer Interface general principles among others are consistency, confirmation, recoverability, verification massage, reverse action and responsiveness.

4.5.1 The System for Analyzing Medical Data Module

This module is named Medical Data Analysis. The interface lets the general users to select their preferred disease and get to know about what the system provides to them. It is also the main user interface for all users where authorized users also require it to access Stata Software to update the patients data.

The figure below shows the user interface design that had been made as the main form in the system.

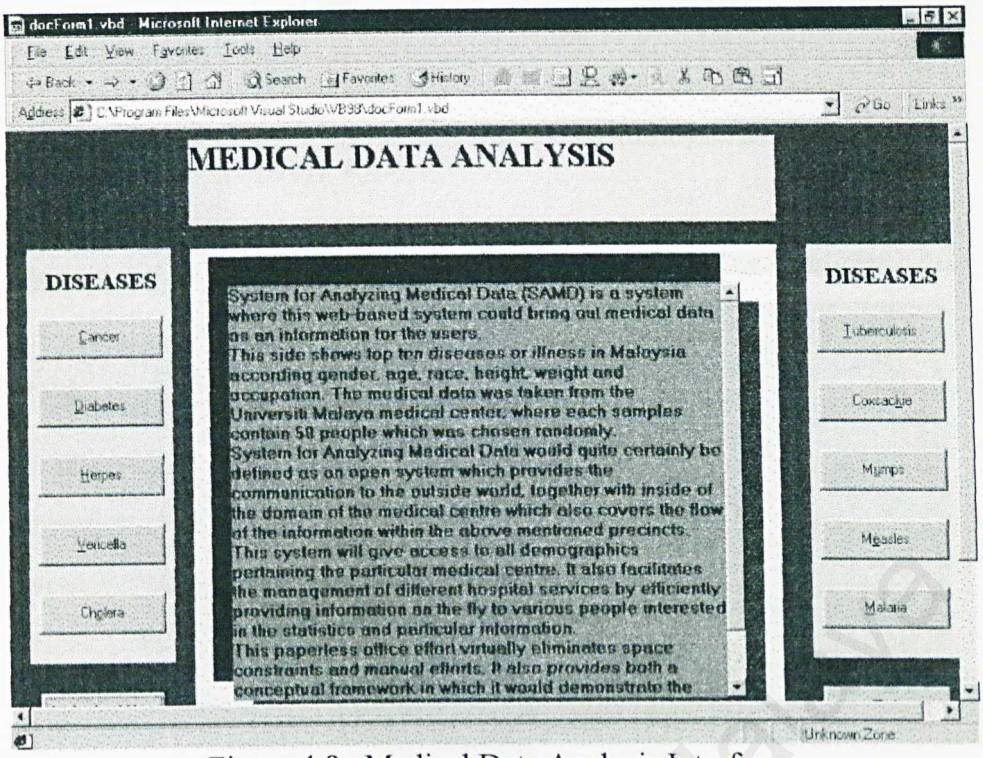


Figure 4.8 : Medical Data Analysis Interface

4.5.2 Top Ten Disease Module

4.5.2.1 Selected Disease Module

This module is named according to the name of selected disease, e.g. cancer disease. The interface lets the users to view results of patients data in a graphical view. The analysis is done by Stata software where graph are generated when they click any one of the 7 factors (sex, race, state, occupation, height, weight, age).

This figure below shows the user interface design that had been made as the selected disease interface (e.g. cancer disease).

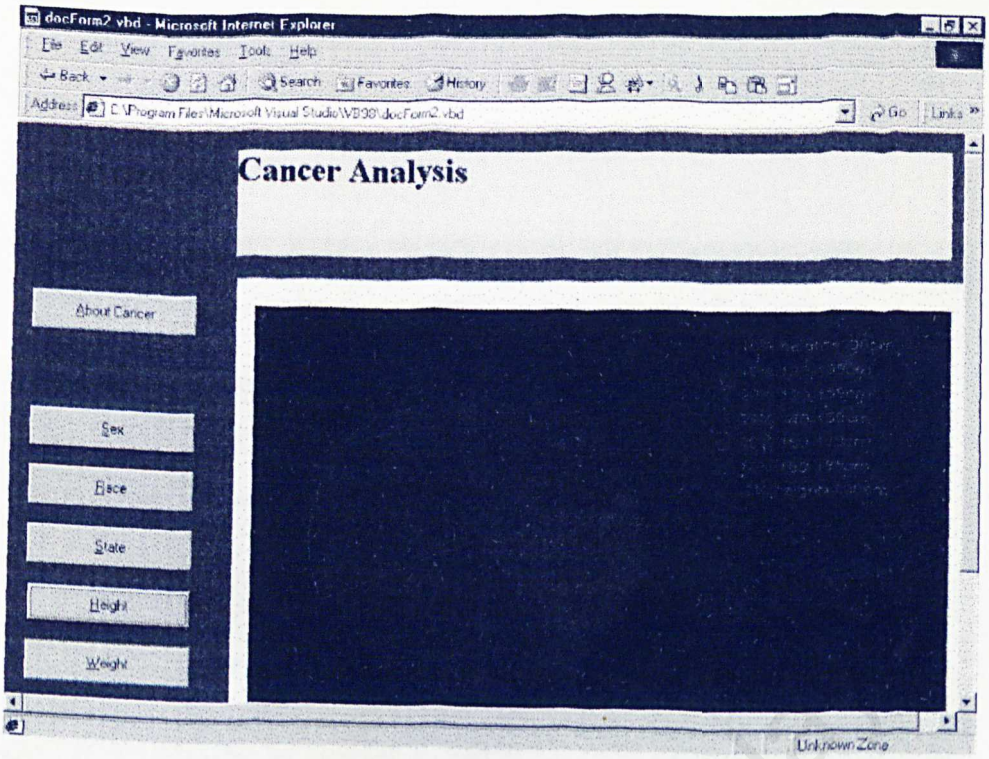


Figure 4.9: Selected Disease Interface

4.5.2.2 General Information Module

This module is also named according to the selected disease, where it contains useful and general information about the selected disease for the users. This system is user-friendly system, where it allows the users to save the information about the selected disease for their own purpose. The text in this module had been edited using Microsoft Words and will be updated by the authorized user as time changes.

This figure below shows the user interface design that had been made as the general information interface. (e.g. General Information On Cancer)

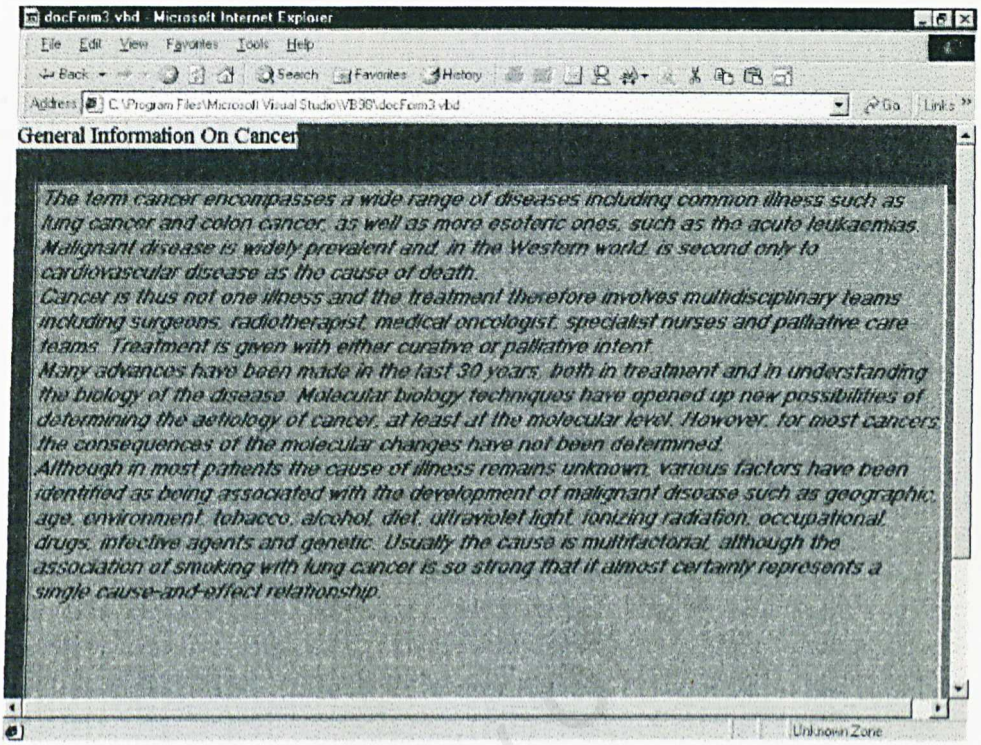


Figure 4.10: General Information on Selected Disease Interface.

4.5.3 Authorized User Module

This module is created for a security and editing purpose for the patient data. It is a login module for the registered users of the system where it contains username and password. This module only allows the authorized users to access to the Stata software that contains patients data in its database called Stata Data Editor. The administrator can update or edit the data as the time changes and generate new graph with the new data and display it in the system.

SYSTEM FOR ANALYZING MEDICAL DATA

This figure below shows the user interface design that had been made as the authorized user or login module.

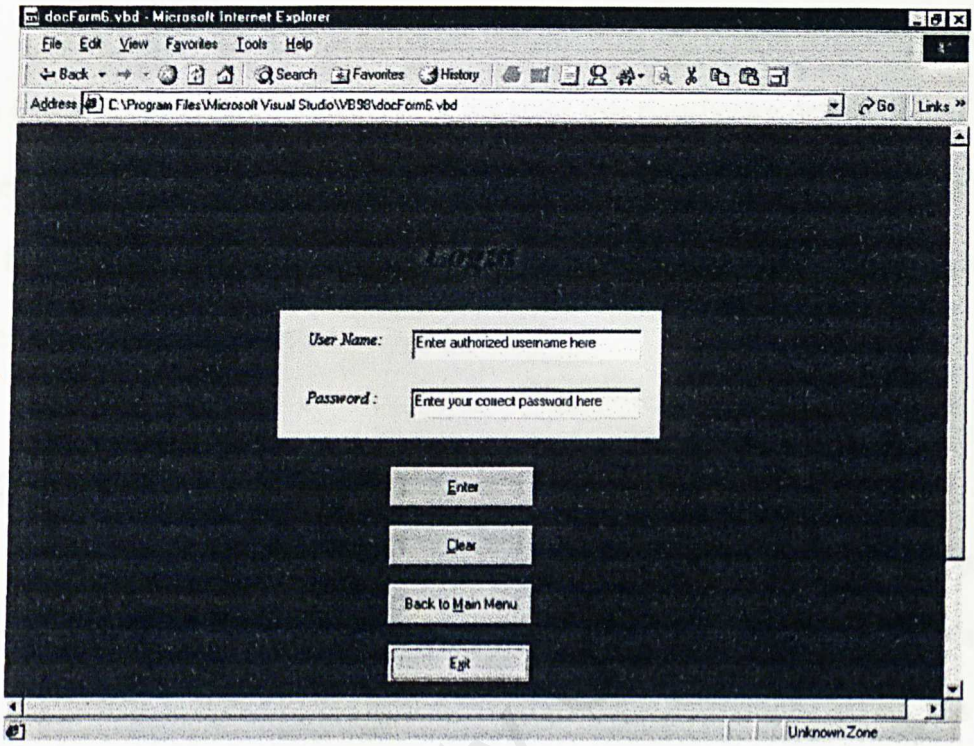


Figure 4.11: Authorized User Interface

CHAPTER 5: SYSTEM IMPLEMENTATION

5.1 INTRODUCTION

System Implementation is a process that converts the system requirement and design into program codes. It involves coding steps that translates a details design representation of software into programming language realization. Codes that had been assign to a component or objects would apply the functions or process that had to be done in order to make them applicable.

5.2 DEVELOPMENT ENVIRONMENT

Development environment has certain impact on the development of a system. It is all about using the suitable software and hardware in the developing process of a system. It also help to speed up the system development and determine the success of the project. All the hardware and software tools that has been used to develop the entire system are discussed in this chapter.

5.2.1 Hardware Requirements

The hardware specifications of the System for Analyzing Medical Data project are shown in the table below:

SYSTEM FOR ANALYZING MEDICAL DATA

SERVER	WORKSTATION
1) Running on Windows 2000 Server and Internet Information Server. 2) Consists of 256 MB RAM and hard disk 3) A Pentium processor 733 MHz and a NIC (Network Interface Card)of Ether 10/100 Mbps speed.	1) Running on Windows 98 or 2000, Microsoft Visual Studio 6.0, Internet Explorer 5.0, Intercooled Stata 6.0. 2) Consists of 128 MB RAM with a hard disk. 3) A Pentium processor 733 MHz and a NIC (Network Interface Card)of Ether 10/100 Mbps speed. 4) Other standard PC component

5.2.2 Software Tools Requirements

In the development of System for Analyzing Medical Data project, the software applied is basically consisted of components and tools. The components includes all the technologies used to support the functionality of the system that has been developed. Whereas the tools applied are those development applications used to develop and design.

5.2.2.1 Descriptions of Development Applications or Tools

The list below shows the categories application or tools used for the System for Analyzing Medical Data project development:

a) Application coding tools

- Microsoft Visual Studio 6.0 –

  Creates and refines web page for the whole system. (Visual Basic ActiveX Document Migration Wizard)

SYSTEM FOR ANALYZING MEDICAL DATA

b) Database Implementation Tools

- Intercooled Stata 6.0 –

  Insert and edit patients data in the Stata Data Editor spreadsheet.

c) Graphics and Interface Modeling Tools

- Intercooled Stata 6.0 –

  Generate graphs using patients data in the data editer.

- Microsoft Visual Studio 6.0 –

  Creating and editing the web page interface design.

5.3 INSTALLATION AND SETUPS

The first step before starting off with any project development work are the installation of Server and development tools (workstations). In using Microsoft's products, it is essential to know the sequence of products installation to ensure smooth execution without system errors. Installation has to be done in both server and workstation.

The list below, is the sequence or flow of installation process on the server:

a) Windows 2000

b) Microsoft SQL Server 7.0

For the workstation, the installation process are:

a) Windows 98 or Windows 2000

b) Microsoft Visual Studio 6.0

c) Intercooled Stata 6.0

d) Microsoft Office 2000(With Photodraw included)

e) ACD See

5.3.1 Create Database

Database for this for System for Analyzing Medical Data are created by using Stata Data Editor or from a data file.

5.3.1.1 Entering Data in the Stata Editor

Click on the Editor button. The Stata Editor window, containing a spreadsheet, will be displayed. In the spreadsheet each column represents a variable and each row represents an observation. The highlighted cell is the one in which the next variable you

enter will be stored. The address of the current cell is displayed above the spreadsheet in the form **var*n*[*x*]**, where n represents the number of the variable and x represents the number of the observation. For example, the second cell in the fourth column would be identified as var4[2].

a) To enter data observation-by-observation:

Enter the data for the first variable in the first cell in the upper left corner of the spreadsheet. Press the **Tab** key to move to the next cell to the right. After you have typed the last variable in your first observation and pressed Tab, click on the first cell in the second row to begin entering the next observation. After this, Stata will know how many variables you have in each observation and will move automatically to the next row.

b) To enter variable-by-variable:

Enter the data for the first variable in the first cell in the upper left corner of the spreadsheet. Press the **Enter** key to move down to the next row. Stata automatically numbers each observation in the gray row heading area to the left of the first column.

c) Missing values:

If you are missing data for a particular observation, simply press Tab or Enter at the missing variable. Stata will automatically insert a period (.) in that cell to indicate a missing value.

Important Note: Stata will not allow empty rows or columns In the middle of your dataset. Always add a new variable in the next empty column to the right and enter a new observation in the next empty row down.

d) Naming Your Variables

By default Stata identifies your variables as var1, var2, var3 . . . . You will want to give your variables meaningful names. To name a variable:

- Double-click on any cell in the variable's column.
- The Stata Variable Information dialog box will appear with the default variable name highlighted.
- Type in the name of the variable, observing these rules:

e) Variable Labels:

In the Stata Variable Information dialog box you may optionally enter a label for each variable. A label is a description of the variable that may contain spaces. Its purpose is to provide more information about the data than may be gained from the variable name, which is limited to eight characters. For example, a variable containing a state's name Terengganu might Tgn be given the more descriptive label "Terengganu".

f) Format:

Stata stores data in the following formats:

float: Real numbers, 8.5 digits of precision
double: Real numbers, 16.5 digits of precision

byte: Integers between -127 and 126
int: Integers between 32,768 and 32,766
long: Integers between -2,147,483,648 and 2,147,483,646
str*n*: string (non-numeric) where *n* represents the number of characters in the variable, from 1 to 80. For example, str1 is a one-character long string variable.

At this point it is not necessary for you to change the information in the Format text box.

When you have entered the variable name and variable label, click on OK. Repeat for each variable. The variable name will now be displayed in the gray column heading area above the first observation.

e) Copy and Paste Your Data

If data are repeated in your dataset, there is no need to type them more than once.

Use the copy and paste features of the Editor. Select the cell or cells you wish to copy:

- To select a variable in every observation, click once on the variable name (in the gray area above the first row).
- To select an entire observation, click once on the observation number (in the gray area to the left of the first column).
- To select a range of cells, click and drag your mouse to select a range of cells.
- From the Edit menu select Copy Editor Data.
- Click on the top left cell of the area where you want the data to be copied.
- From the Edit menu select Paste.

f) Saving Your Data

You cannot save your data until you exit from the editor.

1. Exit from the Editor:

2. From the File menu select Save As.

3. In the Save Stata Data File dialog box:

- specify the drive on which you want to save your data,
- select a subdirectory on that drive if you wish, and
- enter the name of the file.

File names must be no longer than 8 characters long. Click on OK. Stata will automatically add the extension .**dta** to the file name.

5.3.1.2 Reading in a Data File

Rather than entering data directly into the Stata editor, you may wish to use a data file created in another software package, such as a word processor or a spreadsheet. Of these, spreadsheet files can be handled most conveniently in Stata. Stata provides three commands (insheet, infile, infix) to read in a data file. Be sure to select the command appropriate for your file.

Before you can use it in Stata, the data file must be in text form. If you have not already done so, call up the file in the other software package and save it in text (ASCII) format.

a) Insheet for Spreadsheet Files

If your file was created in a spreadsheet (like Excel, Lotus 1-2-3, Quattro Pro) with the ***variables separated by a comma or a tab***, use **insheet**. Enter the following in the Command window:

insheet using ***filename***

For example, if the file mydata.txt, created in a spreadsheet and saved in text format, is stored on the floppy disk in the A: drive, enter:

insheet using a:mydata.txt

To display your file, enter the command **list** . Stata may be able to read variable names from the text file. Sometimes the insheet command will assign default names to your variables: v1, v2, v3 . . . . To give the variables more useful names, you may enter the insheet command again, followed by the names of the variables, with a space separating the variable names. **insheet *varname1 varname2 varname3*** For example, if your data contains the variables height, weight and sex, enter in the Command window: **insheet height weight sex** . Now, enter the **list** command to see the variable names displayed.

b) Infile for Space-Delimited Files

Use the **infile** command if:

- variables are separated by spaces and the file contains no string (non-numeric) variables
- all string variables are just one word
- all strings are enclosed in quotes

If your file does not meet these conditions, refer to the Stata Reference Manual for help on how to use the infile (fixed format) or infix (fixed format) commands. To use the infile command, enter in the Command window:

infile *varname1 varname2 varname3* . . . using *filename*

For numeric variables, simply enter the variable name. However, the name of a string (non-numeric) variable must be preceded by **str*n*** where n is a number from 1 to 80 that represents the length of the string.

For example, if you were reading in the file myfile.txt on a disk in the A: drive containing the numeric variables height and weight and the one-character string variable sex, you would enter: **infile height weight str1 sex using a:myfile.txt**

To view your data, in the Command window enter: **list**

Note that string variables will be in quotation marks, missing values of string variables will be represented by empty quotation marks ("") and missing values of numeric variables will be represented by a period.

c) Saving Your Data

Once you have read your file into Stata, save it in Stata format.

1. From the File menu select Save As.
2. In the Save Stata Data File dialog box:
   - specify the drive on which you want to save your data,
   - select a subdirectory or folder on that drive if you wish, and
   - enter the name of the file.

File names must be not longer than 8 characters long. Click on OK. Stata will automatically add the extension .**dta** to the file name.

CHAPTER 6: SYSTEM TESTING

6.1 INTRODUCTION

Testing is a critical element of software quality assurance and represents the ultimate review of specification, design and code generation.

The increasing visibility of software as a system element and the attendant "cost" associated with a software failure are motivating forces for well-planned, through testing. It is not unusual for a software development organization to expend between 30 - 40 percent of total project effort on testing.

The objectives of system testing are:

a) Testing is a process of executing a program with the intent of finding an errors

b) A good test case is one that has a high probability of finding an undiscovered error.

c) A successful test is one that uncovers an undiscovered error.

These objectives imply a dramatic change in view point. They move counter to the commonly held view that a successful test is one which no errors are found.

A software engineer must understand few basic principles that guide system testing and the principles are:

a) All test should be traceable to users requirements

b) Test should be planned long before testing begins

c) Testing should begin 'in the small' and progress toward testing 'in the large'

d) Exhaustive testing is not possible.

e) To be most effective , testing should be conducted by an independent third party.

In developing a system, testing usually involves several stages. First, each program component is tested on its own, isolated from the other components in the system. This testing is called or known as unit/module testing.

The main objective of unit testing is to confirm that the unit is correctly coded and that it carries out the functions it is supposed to carry out. This stage of testing verifies that the component functions properly with the types of input and output expected from studying the component's design. After each components has been tested, the interaction between these components must be tested again to ensure that the components can be integrated.

When the individual components are working correctly and meet the objective, these components are combined into a working system. Integration testing is done on the groups of integrated modules to verify that the system components work together as described in the system and program design specifications.

System testing is the final testing procedure. A system test is a series of different tests designed to fully exercise the system to uncover its limitations and measure its capabilities. The goal is to test an integrated system and verify that it meets specified requirements. System testing takes place at higher level, the testing focuses on behavior rather than function or functional structure.

6.2 UNIT TESTING

The developer of the System for Analyzing Medical Data applied or used the following three types of testing categories:

6.2.1 Ad Hoc Testing

Ad Hoc or Ad Lib testing means the developer of the system, simply play or test with the functioning unit, where he tries what ever comes to his mind to make the system fail. One of the weakness in this testing is the developer can never be sure on what was or was not tested. It'll be quite complicating for him at the end of the testing. Whereas, the good thing about this testing is fast and a efficient way of debugging code errors during the primary stage of development.

6.2.2 White-Box Testing

White-Box testing, sometimes called Glass-Box testing, is a test case design method that uses the control structure of the procedural design to derive test case. Using white-box testing methods, the developer can derive test case that:

a) Guarantee that all independent paths within a module have been exercise or tried at least once.

b) Exercise all logical decisions on their true and false sides.

c) Execute all loops at their boundaries and within their operational bounds.

d) Exercise internal data structures to ensure their validity.

6.2.3 Black-Box Testing

Black-Box testing, also called as behavioral testing, in which it focuses on the functional requirements of the software. This black-box testing enables the developers to derive sets of input conditions that will fully exercise all functional requirements for a program. Black-box testing is not an alternative to white-box testing techniques. Rather, it is a complementary approach that is likely to uncover a different class of errors than white-box method.

Black-box testing attempts to find errors in the following categories:

a) incorrect and missing functions
b) interface errors
c) errors in data structures or external database access
d) behavior or performance errors
e) initialization and termination errors.

Unlike white-box testing, which is performed earlier in the testing process, black-box testing tends to be applied during later stages of testing. Because black-box testing purposely disregards control structures, attention is focused on the information domain.

Here is the summary of units that were independently unit-tested:

a) Insertion of new data into database.
b) Modification of existing data in the database.
c) Retrieving data from database.

d) Validation of user identity before granting permission to database.

e) Uploading the picture file.

f) Opening and closing of connection to database in Stata software.

6.3 INTEGRATION TESTING

Integration testing is a systematic technique for constructing the program structure while at the same time conducting tests to uncover errors associated with interfacing. The objective is to take unit tested components and build a program structure that has been detected by design.

There is often tendency to attempt nonincremental integration; that is to construct a program using a 'big bang' approach. All the components are combined in advance. The entire program is tested as a whole. And chaos usually results. A set of errors encountered. Correction is difficult because isolation of causes is complicated by the vast expanse of the entire program. Once these errors are corrected, new ones appear and the process continues in a seemingly endless loops.

Incremental integration is the antithesis of a 'big bang' approach. The program is constructed and tested in small increments, where errors are easier to isolate and correct; interfaces are more likely to be tested completely; and a systematic test approach may be applied.

For this project, incremental approach was used where the units are added one by one to the set of an integrated unit. Here are a few types of different incremental integration strategies:

a) Top-down Integration – modules are integrated by moving downward through control hierarchy.

b) Bottom-up Integration – begins construction and testing with modules or components at the lowest level in the program structure.

c) Regression Testing – re-execution of some subset of tests that have already been conducted to ensure that changes in program structure have not propagated unintended side effects.

d) Smoke testing – designed as a pacing mechanism for time critical projects, allowing the software teams to assess its project in a frequent basis.

6.4 SYSTEM TESTING

System testing is actually a series if different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work to verify that system elements have been properly integrated and perform allocated function. The list below, shows several types of system testing that had been applied in this project:

a) Recovery Testing

Recovery testing is a system test that forces the system to fail in a variety of ways and verify that recovery is performed properly. If recovery is automatic, reinitialization, checkpointing mechanism, data recovery and restart are evaluated for correctness. In this project, confirmation message will appear when user attempt to quit or exit from the system. Error message also appears when the users input the incorrect password or username in the login form.

b) Security Testing

Security testing attempts to verify that protection mechanisms build into a system will, in fact, protect, it from improper penetration. In this project, security testing has been done to the login form where it only allows the registered or authorized admin to access Stata software in which they could edit or modify data according the changes in patient data. It would be better, if the authorized user update the data every month so that the users could get analysis of the new data and it also helps the medical centre in the management side.

c) Stress Testing

Stress testing executes a system in a manner that demands resources in abnormal quantity, frequency or volume. A variation of stress testing is a technique called 'sensitivity testing'. In some situations, a very small range of data contained within the bounds of valid data for a program may cause extreme and even erroneous processing or profound performance degradation. Sensitivity testing attempts to uncover data combinations within valid input classes that may cause instability or improper processing.

In this project, sensitivity testing is not required because Stata software could overcome this problem because it can manage the datasets in a proper way.

d) Performance Testing

As a reminder, the System for Analyzing Medical Data is a sub-module for Smart Healthcare Information System. So the integration that was mentioned is the integration in this sub-module only and not the integration system with the Smart Healthcare Information System.

Performance testing is designed to test the run-time performance of software within the context of an integrated system. Performance testing occurs throughout all the steps in the testing process. Performance testing is often coupled with stress testing and usually require both hardware and software instrumentation. That is, it is often necessary to measure resource utilization (e.g., processor cycles) in an exacting fashion. External instrumentation can monitor execution interval, log events (e.g., interrupts) as they occurs, and sample machine states on

a regular basis. By instrumenting a system, the developer can uncover situation that lead to degradation and possible system failure.

In this project, performance testing had been done to Stata software and hardware that are required for the system. It was not causing any problems and the functions behaved according to the requirements because all the software is friendly to the operation system (Microsoft Windows 2000).

CHAPTER 7: SISTEM EVALUATION

7.1 INTRODUCTION

Evaluation is a process that occurs continuously, drawing or sketching on a variety of sources and information in all the phases of the system.

The objective of this phase is to determine:

a) the extent in which the expected outcomes have been realized and notified
b) the prescriptive value of the process where extraneous factors were taken consideration.

7.2 SYSTEM STRENGTHS

- Provide uploading features to enable administrators to upload System for Analyzing Medical Data's picture and graph files into the server for the general users.
- Provide a well managed database in Stata software's Data Editor that makes an easy job for the administrators to edit and modify data according to the variables assigned in the spreadsheet.
- Provide an user-friendly environment where the general users could download the graphs and picture. The information on diseases also can be downloaded by using copy and paste functions.
- The user interface is simple and user-friendly for users to get information to their needs.

- Error message will appear in the login form (authorized user module) in which user without the correct username and password could not excess to Stata software to edit or modify data.
- Conformation message will appear when user wants to exit or quit from the system to make sure that they really willing to do that or not.
- A tag would appear when the user pause or hold their mouse on the buttons available on the forms. This will make the users confident and comfortable with their choices.
- The system is designed for easy navigation, links are provided to help users browse the pages. The 'Back to Main Menu' button are kept in all pages for the users incase the user is "lost".
- Provides attracting colorful graphs to be downloaded for users for their own purpose.
- The Stata software provides batch delete by function variable or observation, where it helps the authorized user to manage the patient's data fast and well.

7.3 SYSTEM LIMITATION

- There's only two types of graph (bar and pie chart) that can easily generated, to make comparison and understand by the users. Stata can provide more than that such as matrix, tables, summarize the data and etc.
- The authorized username and password cant be changed in the login form itself but has to be changed from the source code of the system and it doesn't hide the password with symbols (e.g. *, #) to make sure that others couldn't recognize it when the admin input his username and passwords.
- The system has limited disease to talk or view about and the patients data are according to the selected disease. As time changes there might be other diseases in the top ten list and this would make the admin to change edit a lot in the forms like diseases name and general information about the diseases. It will take quite some time for the admin to do changes in the list.
- This system is browser dependent, where it only supported by Internet Explorer and not the Netscape Navigator.

7.4 PROBLEMS AND SOLUTIONS

Throughout this project, many difficult kept unfolding one after another as development work progressed due to many reasons.

During the system analysis phase, since there was no prior experience in developing a system, it was hard to determine to which extent to define the scope of the system so that it can de completed within the given time frame. However, this was overcome by analyzing and studying all of the capabilities that that Visual Basic and Stata software technologies can do before determining the scope of the system.

During design phase, one of the major obstacles is to apply the theoretical information gather in the previous phase into practice use. The developer found it difficult to apply and produce the best solution of design in the first thesis. Mainly, this was due to lack of experience and insufficient knowledge of designing a system. Reading through some of the senior documentation and revising some of the text book from previous subjects were quite helpful in solving the matter.

During the implementation phase, due to no prior experience in Visual Basic and Stata, there was a bit learning curve in understanding how the Stata works due to lack of reference book. There is only tutorials in internet that can be done to get a deeper understanding in it.

Technical problems that were not familiar have to be solved through various ways, reading developers reference books, getting help from suoervisor, friend and through internet.

In short, build a web application from scratch, starting with system requirements, specification to designing , implementing and testing the system within a very limited time constraint, was not an easy task to accomplish.

CHAPTER 8: SUMMARY AND CONCLUSION

On the whole this project has achieved to deliver the system in specified time frame and fulfilled the objective s and requirements as determined during system analysis phase.

This system not only provide graphical view for the users to get a clear picture on statistical analysis, it also helps them to get some knowledge in the diseases that they interested in. Adding to that, they also can predict the effect of the disease by analyzing the main seven aspects that related to the disease.

During the literature review phase, the information and data on the current Web technologies such as ASP, Visual Basic, client server architecture, and security issues enlighten the developer on how internet technologies actually works.

During the development of this project, there was a lot of valuable lessons learned and plenty of hand-on experience gained. Learning to program in Visual Basic would be a added valuable skills to the developer further undertaking. Rather than that, getting to know about Stata is also valuable because the usage of the software is growing in the statistical analysis market.

While programming skills are essential, good practice on software engineering techniques must also be applied efficiently. This project has provided a excellent opportunity to apply the theoretical knowledge obtained in subject such as Software Engineering and System Analysis and Design courses on system development life cycle (SDLC), data modeling, design architectural, testing techniques and such into beneficial and good use. Apart from technical knowledge, the project also help polished the

developer non-technical knowledge, such as communication skills, organizing skills and problem solving skills.

In conclusion, this project was a unforgettable and a great learning opportunity, both practically and theoretically.

APPENDIX A: SAMPLE CODES

Login Form.

```
Private Sub UserDocument_Initialize()
   Call Form_Load
End Sub

Private Sub Command1_Click()
If Text2.Text = "bavan" And Text1.Text = "badmini" Then
MsgBox ("You're accepted and welcome to STATA")
ExAppl = Shell("start c:\stata\wstata.exe", vbHide)
Else: MsgBox ("Your password or username is incorrect")
End If
End Sub

Private Sub Command2_Click()
Text1.Text = ""
Text2.Text = ""
End Sub

Private Sub Command3_Click()
Text1.Text = "Enter authorized username here"
Text2.Text = "Enter your correct password here"
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm1.vbd"
End Sub

Private Sub Command4_Click()
Form23.Show
End Sub

Private Sub Form_Load()
Text1.Text = "Enter authorized username here"
Text2.Text = "Enter your correct password here"
End Sub
```

Main Menu Form

```
Private Sub Command1_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm2.vbd"
End Sub

Private Sub Command10_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual
Studio\VB98\docForm21.vbd"
End Sub

Private Sub Command11_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm6.vbd"
End Sub

Private Sub Command12_Click()
Form23.Show
End Sub

Private Sub Command2_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm4.vbd"
End Sub

Private Sub Command3_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm7.vbd"
End Sub

Private Sub Command4_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm9.vbd"
End Sub

Private Sub Command5_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual
Studio\VB98\docForm12.vbd"
End Sub

Private Sub Command6_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual
Studio\VB98\docForm14.vbd"
End Sub

Private Sub Command7_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual
Studio\VB98\docForm17.vbd"
End Sub
```

```
Private Sub Command8_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual
Studio\VB98\docForm19.vbd"
End Sub

Private Sub Command9_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual
Studio\VB98\docForm22.vbd"
End Sub
```

Selected Disease Analysis Form(e.g. cancer)

```
Private Sub Command12_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\cancer.jpg")
End Sub

Private Sub Command13_Click()
Hyperlink.NavigateTo "C:\my documents\sara's system\stata files\graph\cancer"
End Sub

Private Sub UserDocument_Initialize()
   Call Form_Load
End Sub

Private Sub Command1_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm3.vbd"
End Sub

Private Sub Command10_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm4.vbd"
End Sub

Private Sub Command11_Click()
Form23.Show
End Sub

Private Sub Command2_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\sex.wmf")
End Sub

Private Sub Command3_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\race.wmf")
End Sub

Private Sub Command4_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\state.wmf")
End Sub

Private Sub Command5_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\height.wmf")
End Sub
```

```
Private Sub Command6_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\weight.wmf")
End Sub

Private Sub Command7_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\occupation.wmf")
End Sub

Private Sub Command8_Click()
Picture2.Picture = LoadPicture("C:\my documents\sara's system\stata
files\graph\cancer\age.wmf")
End Sub

Private Sub Command9_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm1.vbd"
End Sub

Private Sub Form_Load()
Picture2.Picture = LoadPicture("C:\my documents\sara's
system\disease\picture\cancer.jpg")
End Sub
```

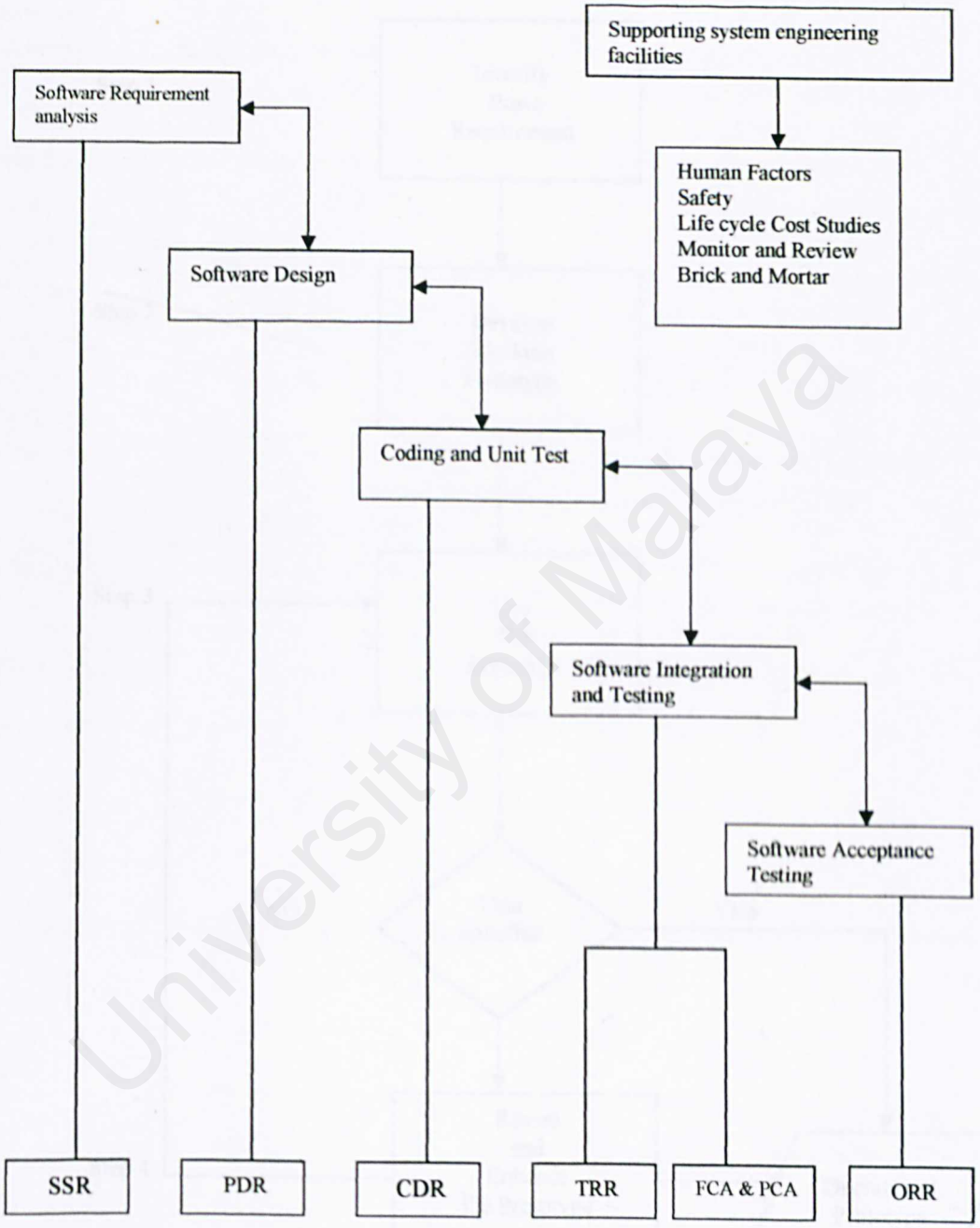
General Information On Selected Disease(e.g. cancer)

```
Private Sub Command1_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm1.vbd"
End Sub

Private Sub Command2_Click()
Hyperlink.NavigateTo "C:\Program Files\Microsoft Visual Studio\VB98\docForm2.vbd"
End Sub

Private Sub Command3_Click()
Form23.Show
End Sub
```

APPENDIX B : FIGURES AND TABLES



SSR = System software review
PDR = Preliminary design review
CDR = Critical design review
TRR = Test readiness review
FCA = Functional configuration audit
PCA = Physical configuration audit
ORR = Operational readiness review

FIGURE 1 :- THE WATERFALL MODEL

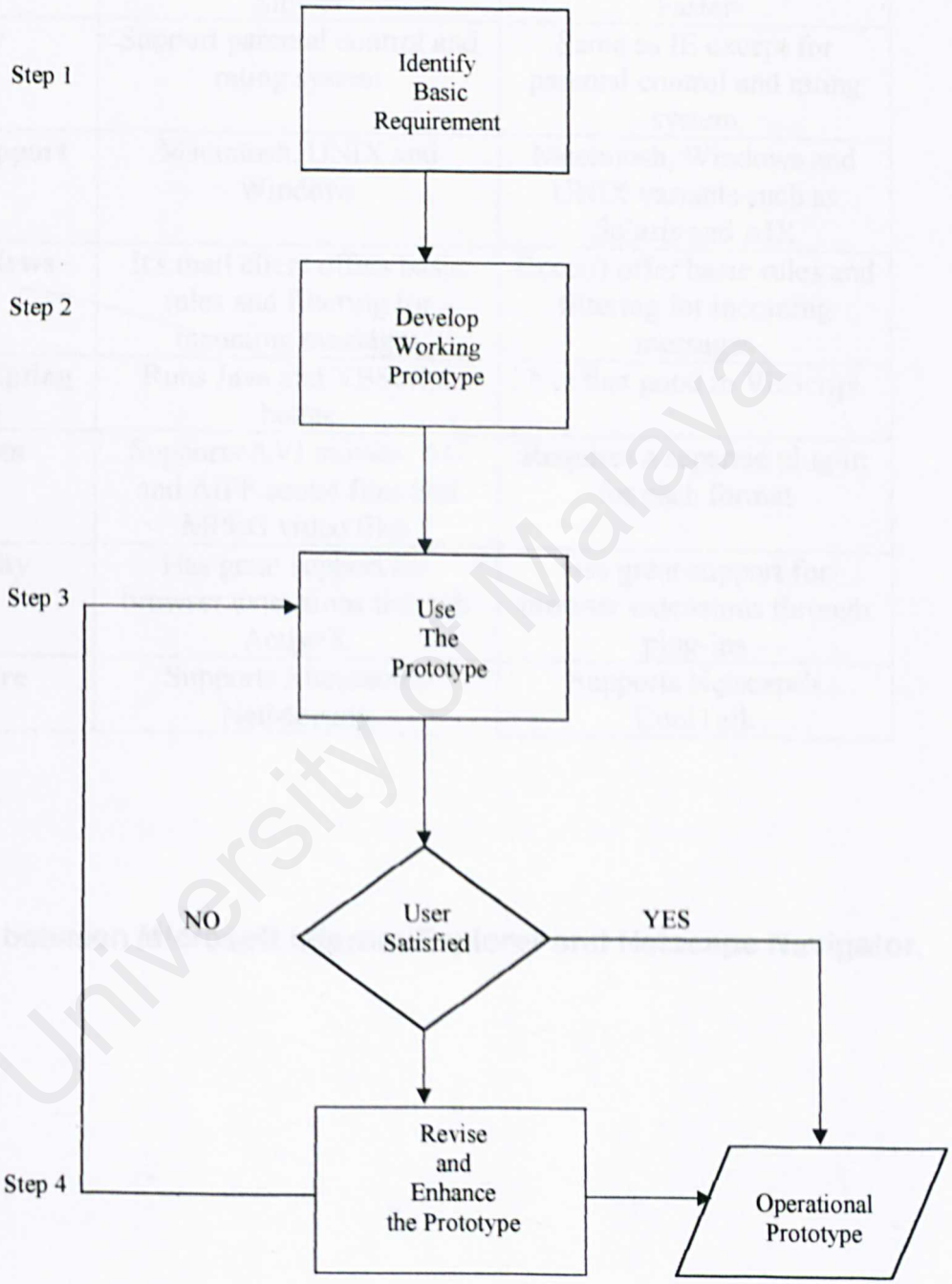


FIGURE 2 :- PROTOTYPE MODEL

FEATURES	INTERNER EXPLORER (IE)	NETSCAPE NAVIGATOR (NN)
Speed	Slower	Faster
Security	Support parental control and rating system	Same as IE except for parental control and rating system
Platform Support	Macintosh, UNIX and Windows	Macintosh, Windows and UNIX variants such as Solaris and AIX
Mails and News	It's mail client offers basic rules and filtering for incoming messages	Doesn't offer basic rules and filtering for incoming messages
Java and Scripting Support	Runs Java and VBScript better	Not that good in VBScript
Multimedia	Supports AVI movies, AU and AIFF sound files and MPEG video files	Requires a separate plug-in for each format
Extensibility	Has great support for browser extensions through ActiveX	Has great support for browser extensions through plug-ins
GroupWare	Supports Microsoft's NetMeeting	Supports Netscape's CoolTalk

Table 1:-

Comparison between Microsoft Internet Explorer and Netscape Navigator.

FEATURES	Active Server Page (ASP)	Java Server Page (JSP)	Perl
First Introduced	4-5 years	2-3 years	Longer than ASP and JSP
Platforms	Microsoft Windows	Most popular, including the Solaris Operating Environment, Microsoft Windows, Mac OS, Linux and UNIX	Most platforms and Windows NT included
Web Server	Microsoft IIS or Personal Web Server	Any Web Server	Any Web Server
Database Connectivity	Via ODBC	Via ODBC and JDBC	Via DBI libraries and able to make connections to most Database Server

Table 2:-- Comparison of the different scripting tools available

FEATURES	Oracle	SQL Server 7.0	Microsoft Access 97
Platforms	Most platforms	Only Windows NT, 95, 98 and 2000	Most platforms
Database handling	Large	Large	Large
Connections Capabilities	TCP/IP Sockets, Named Pipes (Windows NT) and Unix Sockets (Unixes)	TCP/IP Sockets, Named Pipes (Windows NT)	TCP/IP Sockets, Named Pipes (Windows NT) and Unix Sockets (Unixes)
Security	File-safe security	Integrated security with Windows NT	Security offered via SSH and Access Control Lists

Table 3:-

Comparison of the different Database Servers that is popular in the market.

APPENDIX C : REFERENCES

- Alka Jarvis and Vern Crandall, "**Inroads to Software Quality, How to Guide and Toolkit**", Prentice Hall, 1997.

- Kendall & kendall, "**System Analysis and Design**", 4th Edition, United States of America, Prentice Hall, 1999.

- Shari Lawrence Pfleeger, "**Software Engineering Theory and Practice**", 1st Edition, United States of America, Prentice Hall, 1998.

- Sellappan, P., "**Software Engineering Management & Methods**", Malaysia, Sejana Publishing, 2000.

- Raymond McLeod Jr., "**Management Information Systems**", Prentice Hall, United States of America, 1998.

- M. Morris Mano, "**Digital Design**", Prentice Hall, United States of America, 1991.

- David B. Bromilow, "**MIMS Annual Malaysia DIMS 1997/1998**", 8th Edition, MediMedia International Group, Phillippines, 1997.

- Chris Denny, "**Teach Yourself Internet Programming with Visual Basic 6 in 21 Days**", Sams Publishing, 1999.

SYSTEM FOR ANALYZING MEDICAL DATA

- http://www.cc.umb.edu/Stataguide/stata.html
- www.programmersresource
- http://www.baylorhealth.com
- www.idm.internet.com

APPENDIX D : ACRONYMS

ADO	**ActiveX Data Objects**
ASP	**Active Server pages**
CDO	**Collaboration Data Objects**
CGI	**Common Gateway Interface**
DBMS	**Database Management Systems**
DFD	**Data Flow Diagrams**
DLL	**Dynamic Link Library**
DSN	**Data Source Name**
ERD	**Entity Relationship Diagram**
GIF	**Graphics Interchange Format**
GUI	**Graphical User Interface**
HCI	**Human Computer Interface**
HTML	**Hypertext Markup Language**
HTTP	**Hypertext Transfer Protocol**
IE	**Microsoft Internet Explorer**
IntSHIS	**Integration of Smart Health Care Information Systems**
IS	**Information System**
IIS	**Internet Information Server**
ISAPI	**Internet Server Application Programming Interface**
JPEG	**Joint Photographic Experts Group**
Ms	**Microsoft**
NIC	**Network Interface Card**

ODBC	**Open Database Connectivity**
SQL	**Structured Query Language**
SHIS	**Smart Healthcare Information System**
SMTP	**Simple Mail Transfer Protocol**
URL	**Uniform Resource Locator**
VBScript	**Visual Basic Scripting Tool**
WWW	**World Wide Web**