CHAPTER 4

SYSTEM DEVELOPMENT

"The goal of Computer Science is to build something that will last at least until we've finished building it"

Anonymous

4.0 System Development

4.1 Introduction

This chapter includes a detailed analysis of the structure of the software design environment and its language as well as a system that has been developed during the project. The development of the system, MediX, comprises of several phases. These include requirements specification and analysis, system design, system implementation and system testing.

4.2 Requirements Specification and Analysis

The purpose of this phase is to analyze and to gain an in-depth understanding of the current system (if available) as well as to collect pertinent data for the new system. Thus, the software engineering activity should proceed from a clear, quality assessed, statement of requirements [Hunt, 1997]. Good physicians determine their own diagnosis but listen carefully to the valuable clues provided by the patient's symptoms and self-diagnosis. This same principle can be applied effectively in requirements analysis. A thorough requirements analysis prior to developing software can significantly improve the quality of the end product. The key benefit of this analysis is a better understanding of the end users' actual and underlying needs.

To accomplish this, questionnaires were used and interviews were conducted. Once the survey was completed, analysis of the data was captured. Meanwhile, personal interviews with the end users are a key component of defining the actual requirements in this project, too. A series of visits to government and private hospitals were undertaken. In the end, the various visits and interviews with a variety of users, proved to be a critical step in the overall development process. This is to ensure that users were involved as early as possible in the project and to verify initial assumptions and requirements of the system that is to be developed. This exercise gave an in-depth knowledge of the functions that the system should have.

Both the functional and non-functional requirements of the system were captured, i.e. the requirements specification. This is to minimize the errors as errors in requirements can be pervasive, dangerous and costly if both the functional and non-functional requirements are not captured properly [Faulk, 1995].

4.2.1 Functional Requirements

Functional requirements are statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations. The functional requirements for the system were elicited by talking to GPs, health care administrators and nurses from selected medical centres. The functional requirements for the system to be developed are divided into two. These include the functional requirements for the MediX administration and the MediX user as listed below.

4.2.1.1 Administration Section

i) Add/Modify/Delete Records of Medical Expertise

This function will allow an administrator to add new, modify and delete existing records of medical expertise in the database. In add sub function, the particulars of the expertise that will be included are the expertise's id, name, gender, specialty and credentials. Moreover, the hospital or the clinic that the expertise is working will be included, too. Meanwhile, for modify and delete sub functions, a search box on the expertise's id will be provided prior to modification or deletion.

ii) Add/Modify/Delete Records of Specialty

An administrator can add new, modify and delete existing records of specialties in the database. This functional requirement is necessary, as the administrators need not enter the specialty of each of the record for the medical expertise. Thus, this reduces redundancy in the database. In addition, the integrity of the data is upheld at all times, too.

iii) Add/Modify/Delete Records of Hospital/Clinic

Similarly, an administrator can add new, modify and delete existing records of hospitals and clinics in the database. The particulars of the hospital or clinic that are needed are the name, address, telephone and fax number as well as email address (if available). This function is important as any modifications made on the address for a particular hospital will be reflected at all of the corresponding records in the medical expertise's record. Hence, the integrity of data will not be compromised.

iv) Report Generation

Reports will be generated automatically by entering the required fields. This function is important for the administrators as reports can be printed out whenever it is needed by the management.

v) Display of Chart/Graph

Charts or graphs can be displayed when there is a need to compare certain fields. The administrator must enter the desired fields to generate the bar or pie charts. This function is important too, as the management can keep track of the medical expertise that are in demand or to be out-sourced.

vi) Analysis

The analysis function is required to analyze as well as to graph the records in the database. A cross-tabulation, one way frequency for tables or histograms can be generated using simple commands. All of these can be tabulated by using a statistical package.

vii) Add/Delete User

This function will enable an administrator to add co-administrators for the system. A delete function will be available to delete an existing administrator, too.

viii) Change Password

The change password function will allow an administrator to change his/her password occasionally, to safeguard the integrity of the database.

4.2.1.2 User Section

i) Search Medical Expertise by Hospital/Clinic at Rural/Urban Area

This function is important for general users who would like to search a particular medical expertise in a specific hospital or clinic by the specified area.

ii) Search by Specialty

This function will enable a general user to search a specialty by typing a keyword. The exact name of the specialty is not required, as the search function will display all the specialties that "sounds like" the keyword.

iii) Search by Symptom

The search by symptom function will allow a general user to search by a particular symptom or sickness. A user needs to type in a keyword and when the search function is invoked, a list of medical expertise at various hospitals and clinics will be displayed. In other words, general users need not know the scientific term for a symptom to invoke the search function.

4.2.2 Non-Functional Requirements

These are constraints on the services or functions offered by the system. They include timing constraints, development process constraints and standards that must be met before a system is delivered to its users.

4.2.2.1 Administration and User Sections

i) User Friendliness

The interface of the system should be easy to use for both the administrators and users. as they may not have enough knowledge in computing skills. Thus, this system should include menus such as pull down or pop-up, descriptive captions and help guides to assist users of the system. Moreover, meaningful captions will simplify users interaction with the system as they will spend less time reading the user manual.

ii) Reliability

Reliability of the system should be taken into account so as to produce accurate results and information pertaining to the medical expertise, hospitals and clinics.

iii) Security

The system will have a security clearance (e.g. password) for administrators who need to maintain the records in the database. This is to ensure that the integrity of the database is not compromised at any time.

iv) Efficiency

The system should be efficient to provide a good response time for the users' entire request. Hence, a delay should not occur in the midst of retrieving any kind of information from the database.

v) *Maintainability and Expandability*

This system must be able to be maintained and expanded (if desired) for future enhancements. Therefore, an easy transition is necessary to change the specification as the system evolves.

4.3 System Design

"*The system design phase is concerned with synthesizing or putting all the parts together into a viable, workable system*" [Sellappan, 1998]. In other words, system design is concerned with how the system functionality is to be provided by the different components of the system. The design processes for MediX are divided into architectural, program, user interface and database designs.

4.3.1 Architectural Design

An architectural design is the initial design process of identifying the sub-systems and establishing a framework for subsystem control and communication. Large systems can be decomposed into subsystems that provide some related set of services [Sommerville, 1995]. Architectural design usually requires the following activities: system structuring and modular decomposition.

4.3.1.1 System Structuring

In system structuring, the system is structured into a number of principal subsystems. A sub-system is a system in its own right or in other words, it is an independent software unit. Its operations do not depend on the services provided by other sub-systems. They are composed of modules and each of this sub-system has defined interfaces that communicate with other sub-systems.

MediX consists of two major sub-systems, namely the MediX Administration and MediX User. A structure chart for MediX is as shown in fig. 4.1.

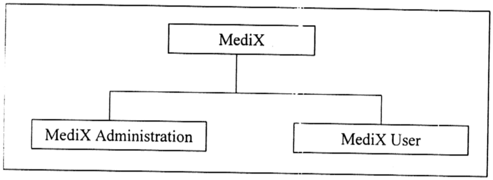


Fig. 4.1 : MediX Structure Chart

4.3.1.2 Modular Decomposition

The modular decomposition is based on assigning functions to components. In other words, each identified sub-system is decomposed into modules. A module can be defined as a system component that provides one or more services to other modules. It makes use of services provided by other modules. It can be considered as a dependent system. They are usually composed from a number of other simpler system components [Sommerville, 1995].

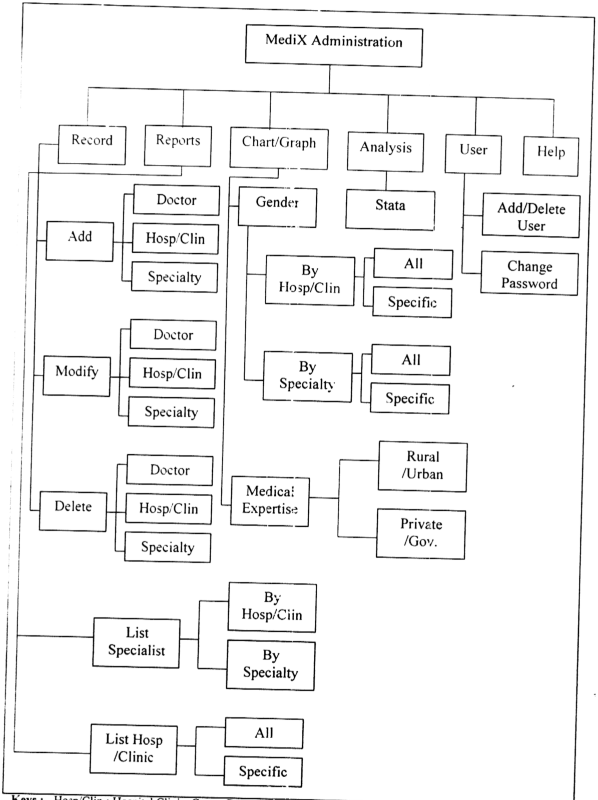
The two modules of MediX, which are the administration sub-system and user sub-system are explained in the following sections.

4.3.2 Program Design

The program design is concerned with translating the design into well-structured program modules. The section below describes the top-down modular approach of MediX.

4.3.2.1 MediX Administration

Fig. 4.2 shows the MediX administration structure chart.



Keys :- Hosp/Clin : Hospital Clinic, Gov. : Government

Fig. 4.2 : MediX Administration Structure Chart

i) Record Module

This module consists of three other sub-modules. They are add, modify and delete. An administrator is allowed to add, delete or modify any of the records of medical expertise, hospital/clinic or specialty in the database. For the add sub-module, the administrator will be prompted for the new record after which it will be saved into the database. Meanwhile, for modify and delete sub-modules, the administrator will be prompted to select an item from the database for modification or deletion.

ii) Reports Module

This module permits an administrator to generate and print reports. A report on a list of specialist can be generated by extracting the list of hospital/clinic or by specialty from the database. In addition, an administrator can generate a report on a list of hospital/clinic by extracting the list of all or a specific hospital/clinic.

iii) Chart/Graph Module

This module allows an administrator to generate charts and graphs based on the selection criteria. A chart will be displayed when the administrator selects the gender function at all hospitals/clinics or a specific hospital/clinic. Meanwhile, if gender by specialty function is selected, then a chart is displayed for all or a specific specialty. For the medical expertise function, a chart can be generated for the selection made for rural/urban areas or by private/government hospitals/clinics.

iv) Analysis Module

This module consists of an integrated statistical package, Stata. Various statistical analyses on the medical expertise can be performed and printed as well.

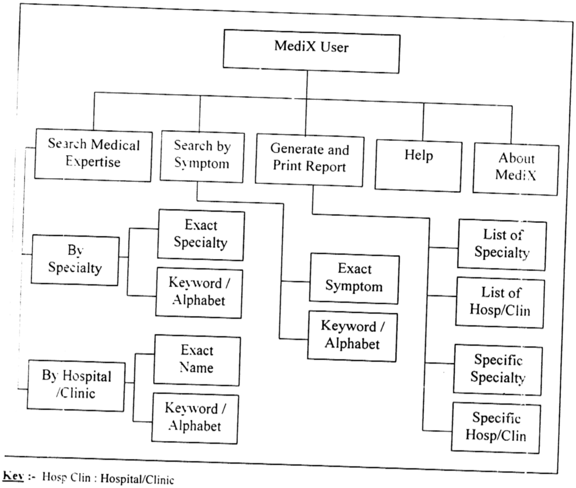
v) User Module

This module allows an administrator to add or delete users for the administration section. When a new user is created, the user will have a unique user name and password. The new user is permitted to access the administration section and he/she then becomes the co-administrator. The delete function deletes an existing administrator from the database. He/She will no longer be able to access the administration section.

The change password function allows an existing administrator to change his/her password. The user name, old password and new password are required to make the change. The user needs to retype the new password to ensure that no typing errors were made. This function is necessary to safeguard the database from unauthorized users.

4.3.2.2 MediX User

A MediX User structure chart is shown in fig. 4.3.



Key :- Hosp Clin : Hospital/Clinic

Fig. 4.3 : MediX User Structure Chart

i) ***Search Medical Expertise Module***

Two search criteria are required before this search can be performed. They are search by specialty and search by hospital/clinic.

A user can search for an exact name or by a related keyword/alphabet for the search function by specialty. For the function search by hospital/clinic, the search can be either a general or specific hospital/clinic. If a specific search category is chosen, the system searches for the respective specialty in a government hospital, government clinic, private hospital or private clinic. Meanwhile, if the general search category is chosen, the system will search for all of the specialties at all the hospitals and clinics in Malaysia.

The user then can highlight and double-click on the respective record. A detailed information on the record is displayed and the user is allowed to print, if desired.

ii) Search by Symptom Module

In order for the user to search by symptom, a user can search for the exact name of symptom or by a related keyword/alphabet. Similarly, the user can highlight and double-click on the respective record. A detailed information on the record is displayed and the user is allowed to print, if desired.

iii) Generate and Print Report Module

The user can generate and print report based on the search criteria. If specialty is chosen, then a report on the list of specialties in various hospitals and clinics are displayed. A report on the list of hospitals/clinics is displayed when the user chooses the field of hospitals/clinics. Specific specialty or specific hospital/clinic report is displayed when only one specialty or hospital/clinic is selected.

iv) About MediX

A brief description on the developer, name and version of the system is given.

4.3.3 User Interface Design

"The user interface of a system is often the yardstick by which that system is judged. An interface that is difficult to use will, at best, result in a high level of user errors" [Sommerville, 1995]. It is with that in mind, the user interfaces for the MediX were designed by taking into account the needs, experience and capabilities of the users.

Two types of user-system interaction were used for this system. The first is the direct manipulation interface, which is a form-based interface. The form-based interface presents the user with a form to fill in. The fields in the form are labeled with the field names to indicate the information that is required. A user can input the information needed and can proceed to the next text box by either using a mouse or the tab key.

The second type of user-system interaction used is the menu interface. Menu items can be selected by pointing with a mouse or by moving a cursor using keyboard commands. Pull-down and pop-up menus are used for the menu interface in this system.

The other area of concern is the use of colour. Colours have been limited for this system as to give a more "conservative" look. Therefore, the colours used throughout this system were consistent. Moreover, a significant number of people are colour-blind and if they were to use the system they may interpret certain functions wrongly.

Error messages are provided by the system to aid users who make mistakes during the operation.

4.3.3.1 User Interface Design for MediX Administration

An introductory screen is presented first. The system presents the user with a logon screen prompting the user to input the administrator's user name and password. If login is successful, the main menu is presented. The administrator can select from the menu bar to execute the desired function as was shown in fig. 4.2. After completing the execution, the administrator can logout from the current screen and the MediX introductory screen is shown once again or he/she can exit from the MediX system.

4.3.3.2 User Interface Design for MediX User

An introductory screen is presented to the user. If the user clicks on the command button to proceed to the next screen, the application invokes a new screen to display the information. The new screen consists of label buttons that permits a user to search for medical expertise or symptom. The print label button allows the user to print the desired records. User may exit from the current screen by clicking the exit button. The introductory screen is presented once more. However, there is no exit button on the introductory screen; hence users are not allowed to exit from the MediX. Only verified administrators can logon and then exit from the MediX system.

4.3.4 Database Design

Microsoft Access 97 is a relational database management system, which is used to create and manage the database for MediX. Data can be retrieved from the database in the client/server-based system by using the ODBC driver for Access. The following table lists the general profile of the database used.

Table 4.1 : General Profile for MediX Database

Database	Microsoft Access 97
Name of Database	MediDB
Data Source Name (DSN)	Medi
Description	To maintain and keep records of medical expertise, hospitals, clinics and specialties

The E-R model is a conceptual data model that views the real world as entities and relationships. It visually represents data objects. The E-R model of the MediX database is as shown below.

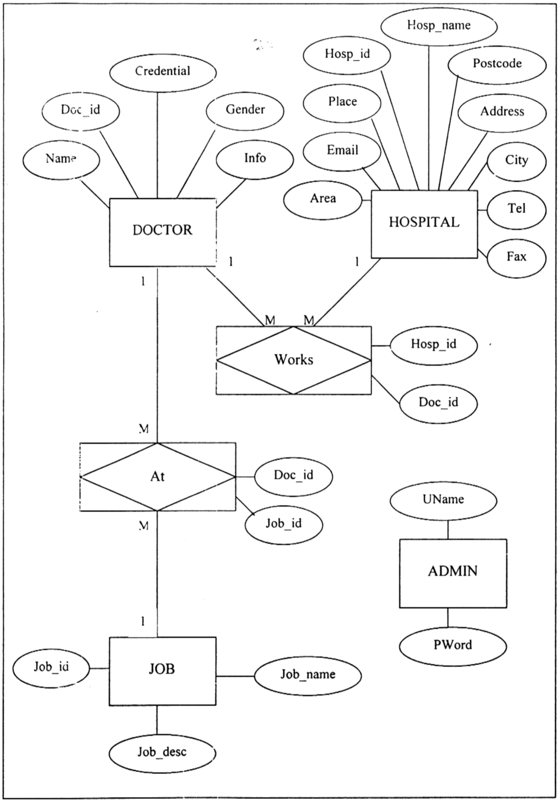


Fig. 4.4 : E-R Model of the MediX Database

MediX database has six tables, namely, Doctor, Hospital, Job, Admin, Works and At. The following section describes the storage of each table in the database.

i) Doctor Table

This table stores the medical expertise's personal particulars.

Table 4.2 : Doctor Table in MediX Database

Field Name	*Data Type*	*Description*
Doc_id	Text	Doctor's ID
Name	Text	Name of the doctor
Gender	Text	Gender of the doctor
Credential	Memo	Credentials of the doctor
Info	Memo	Any extra information pertaining to the doctor

ii) Hospital Table

This table stores information on the hospitals and clinics.

Table 4.3 : Hospital Table in MediX Database

Field Name	*Data Type*	*Description*
Hosp_id	AutoNumber	Hospital id generated automatically
Hosp_name	Text	Name of the hospital/clinic
Address	Text	Address of the hospital/clinic
City	Text	City of the hospital/clinic
Postal_code	Number	Postal code of the hospital/clinic
Tel	Text	Telephone number of the hospital/clinic
Fax	Text	Fax number of the hospital/clinic
Email	Text	Email address of the hospital/clinic
Place	Text	Categorized by government/private hospital/clinic
Area	Text	Categorized by rural/urban hospital/clinic

iii) Job Table

This table stores the type of specialties available.

Table 4.4 : Job Table in MediX Database

Field Name	*Data Type*	*Description*
Job_id	AutoNumber	Job id generated automatically
Job_name	Text	Name of job/specialty
Job_desc	Memo	Description of job/specialty

iv) Works Table

This table stores the medical expertise's id in the respective medical centres.

Table 4.5 : Works Table in MediX Database

Field Name	*Data Type*	*Description*
Doc_id	Text	Doctor id
Hosp_id	Number	Hospital id for this doctor id

v) At Table

This table stores the medical expertise's id with his/her respective specialty.

Table 4.6 : At Table in MediX Database

Field Name	*Data Type*	*Description*
Doc_id	Text	Doctor id
Job_id	Number	Job id for this doctor id

vi) Admin Table

This table stores the administrator's account.

Table 4.7 : Admin Table in MediX Database

Field Name	*Data Type*	*Description*
Uname	Text	User name for the administrator
PWord	Text	Password for this administrator

4.4 System Implementation

System implementation is a process that converts the system design through written program codes. Three-tier client-server architecture was adopted for the implementation of this system. The section that follows discusses the system development tools and the successful integration of the tools used.

4.4.1 System Development Tools

The following tools were used for the development of MediX as shown in the table below:

Table 4.8 : Tools for System Development

Software	*Description*
Microsoft Windows 95	Operating system
Microsoft Visual Basic 6.0 Professional Edition	Program coding
Microsoft Access 97	Database implementation
Seagate Crystal Reports 6.0	Report generation
Intercooled Stata Ver. 6.0	Statistical analysis
Macromedia Flash 3.0	Interface design

Microsoft Access 97, Seagate Crystal Reports 6.0, Macromedia Flash 3.0 and Intercooled Stata Ver. 6.0 were all integrated in the Microsoft Visual Basic 6.0

Professional Edition running under the Microsoft Windows 95 operating system (OS).

4.4.1.1 Operating System

Windows 95 is the operating system used in the development of MediX.

4.4.1.2 Program Coding

Microsoft Visual Basic 6.0 Professional Edition develops rich data forms quickly. The Standard module of VB 6.0 was used to create forms and modules for both the MediX administration and MediX user. The system was developed with a modular approach where each of the modules was developed separately and they were later integrated into a fully functional system. This is to ease for future modifications and enhancements of the system.

i) Program Coding for MediX Administration

A number of forms were used for program coding in MediX administration. The forms and their descriptions in MediX administration are as shown in table 4.9. Some of the sample screens designed for the developed system of the MediX administration have been captured and are shown in Appendix C. Meanwhile, the modules and their descriptions in MediX administration are as shown in table 4.11.

Table 4.9 : Name of Forms and Descriptions for MediX Administration

Form Name	*Description*
FrmAboutAdm	Description of the developer, name and version of MediX
FrmAddDelUser	Screen to add and delete an administrator
FrmADMDoc	Screen to add, delete and modify particulars of medical expertise
FrmADMHosp	Screen to add, delete and modify particulars of hospital and clinic
FrmADMJob	Screen to add, delete and modify particulars of specialty
FrmLogin	Screen to log on to the administration's section
FrmMenu	Screen that displays the menu for the administration's section
FrmPWord	Screen to change an existing password
FrmRepArea	Screen to generate a report for the area selection
FrmRepGen	Screen to generate a report for the gender selection
FrmRepField	Screen to generate a report for the specialty selection
FrmRepHospMenu	Screen to generate a report for the hospital/clinic selection
FrmRepPlace	Screen to generate a report for the place selection
FrmWelcome	Introductory screen for MediX system

ii) Program Coding for MediX User

Similarly, a number of forms were also used for program coding in MediX user. The forms and their descriptions as well as the modules in MediX user are as shown in table 4.10 and table 4.11 respectively. Meanwhile, sample screens designed for the developed system of the MediX user have been also captured and are shown in Appendix C.

Table 4.10 : Name of Forms and Descriptions for MediX User

Form Name	***Description***
FrmAbout	Brief description of the developer, name and version of MediX
FrmDetail	Screen to display the details of the respective specialist
FrmDisplay	Screen to display all the search criteria for users
FrmWelcome	Introductory screen for MediX

Table 4.11 : Modules and Descriptions for MediX Administration and MediX User

Module Name	***Description***
MdlInit	Public procedures and functions to handle the initialization of the combo boxes in MediX Administrator and User
MdlMain	Public procedures and functions used in MediX Administrator and User
MdlSQL	Public procedures and functions that involve read from or write to database

4.4.1.3 Database Implementation

Microsoft Access 97 was used to implement the MediX database by following the database design specified in section 4.3.4.

4.4.1.4 Report Generation

It is likely that most applications of any complexity will need to use one or more reports to summarize or print the data. In view of that, the Seagate Crystal Reports 6.0 [Seagate Crystal Reports for Rational] was used for all of the report generation in this system. The reports' names and their descriptions for the MediX system are as shown in table 4.12.

Table 4.12 : Names of Reports and Descriptions for MediX

Report Name	***Description***
RptAreaJob	This report is in a form of a pie chart. It shows the ratio of medical expertise for a specific field at rural and urban areas.
RptGenAll	Displays a pie chart of all male and female medical expertise at all hospitals and clinics.
RptGenFieldAll	Displays a bar chart of all male and female medical expertise in their respective specialty.
RptGenFieldIndi	Displays a bar chart of all male and female medical expertise in a particular specialty.
RptGenIndi	Displays a pie chart of all male and female medical expertise at a specific hospital/clinic.
RptHospital	A list of all medical expertise in their respective hospitals and clinics.
RptJob	A list of all medical expertise in a specific field at various hospitals and clinics.
RptListHosp	A list of all hospitals and clinics.

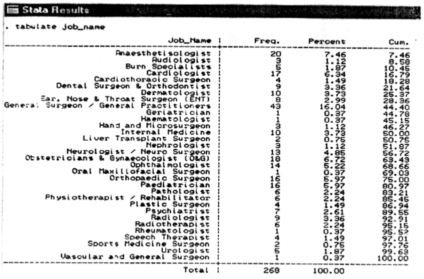
4.4.1.5 Statistical Package

Stata is a statistical package for managing, analyzing and graphing data. Stata is available for a variety of platforms but, regardless of platform, Stata is command-driven. The user-interface model is type a little, get a little, so that the user is always in control [StataCorp, 1999]. Stata's model for a dataset is that of a table - the rows are the observations and the columns are the variables. Stata is very fast as Stata keeps the data in memory.

Stata was used for the analysis section in MediX. In MediX, to use statistics as a tool in pursuing the analysis, some commands need to be written first. The written commands were put into the *.do file where that file was executed whenever the dataset was executed. Examples of the analysis that are provided in MediX are as shown below:

i) One-way (Frequency)

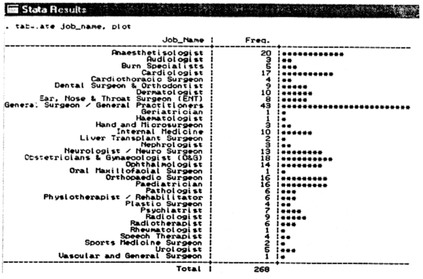
. tabulate job_name



```
Stata Results
. tabulate job_name

                                  Job_Name |      Freq.     Percent        Cum.
-------------------------------------------+-----------------------------------
                      Anaesthetisologist |         20        7.46        7.46
                             Audiologist |          3        1.12        8.58
                         Burn Specialists |          5        1.87       10.45
                             Cardiologist |         17        6.34       16.79
                     Cardiothoracic Surgeon |          4        1.49       18.28
             Dental Surgeon & Orthodontist |          9        3.36       21.64
                            Dermatologist |         10        3.73       25.37
            Ear, Nose & Throat Surgeon (ENT) |          8        2.99       28.36
 General Surgeon / General Practitioners |         43       16.04       44.40
                             Geriatrician |          1        0.37       44.78
                            Haematologist |          1        0.37       45.15
                   Hand and Microsurgeon |          3        1.12       46.27
                        Internal Medicine |         10        3.73       50.00
                  Liver Transplant Surgeon |          2        0.75       50.75
                             Nephrologist |          3        1.12       51.87
                Neurologist / Neuro Surgeon |         13        4.85       56.72
      Obstetricians & Gynaecologist (O&G) |         18        6.72       63.43
                          Ophthalmologist |         14        5.22       68.66
               Oral Maxillofacial Surgeon |          1        0.37       69.03
                      Orthopaedic Surgeon |         16        5.97       75.00
                             Paediatrician |         16        5.97       80.97
                              Pathologist |          6        2.24       83.21
         Physiotherapist / Rehabilitator |          6        2.24       85.45
                          Plastic Surgeon |          4        1.49       86.94
                             Psychiatrist |          7        2.61       89.55
                              Radiologist |          9        3.36       92.91
                          Radiotherapist |          6        2.24       95.15
                          Rheumatologist |          1        0.37       95.52
                         Speech Therapist |          4        1.49       97.01
                  Sports Medicine Surgeon |          2        0.75       97.76
                                Urologist |          5        1.87       99.63
             Vascular and General Surgeon |          1        0.37      100.00
-------------------------------------------+-----------------------------------
                                    Total |        268      100.00
```

Fig. 4.5 : A Sample Screen for Tabulation Of Job

. tabulate job_name, plot

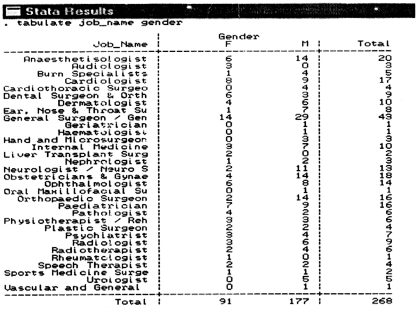


```
Stata Results
. tabulate job_name, plot

                                  Job_Name |      Freq.
-------------------------------------------+------------+-----------------------------
                      Anaesthetisologist |         20 |************
                             Audiologist |          3 |**
                         Burn Specialists |          5 |***
                             Cardiologist |         17 |**********
                     Cardiothoracic Surgeon |          4 |**
             Dental Surgeon & Orthodontist |          9 |*****
                            Dermatologist |         10 |******
            Ear, Nose & Throat Surgeon (ENT) |          8 |*****
 General Surgeon / General Practitioners |         43 |*************************
                             Geriatrician |          1 |*
                            Haematologist |          1 |*
                   Hand and Microsurgeon |          3 |**
                        Internal Medicine |         10 |******
                  Liver Transplant Surgeon |          2 |*
                             Nephrologist |          3 |**
                Neurologist / Neuro Surgeon |         13 |********
      Obstetricians & Gynaecologist (O&G) |         18 |**********
                          Ophthalmologist |         14 |********
               Oral Maxillofacial Surgeon |          1 |*
                      Orthopaedic Surgeon |         16 |*********
                             Paediatrician |         16 |*********
                              Pathologist |          6 |***
         Physiotherapist / Rehabilitator |          6 |***
                          Plastic Surgeon |          4 |**
                             Psychiatrist |          7 |****
                              Radiologist |          9 |*****
                          Radiotherapist |          6 |***
                          Rheumatologist |          1 |*
                         Speech Therapist |          4 |**
                  Sports Medicine Surgeon |          2 |*
                                Urologist |          5 |***
             Vascular and General Surgeon |          1 |*
-------------------------------------------+------------+-----------------------------
                                    Total |        268
```

Fig. 4.6 : A Sample Screen for Tabulation Of Job with Plot

ii) Two-way (Cross-tabulation)

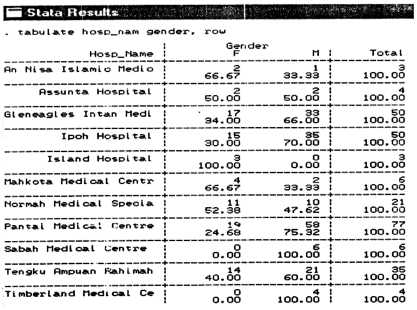
. tabulate job_name, gender



```
Stata Results
. tabulate job_name gender

                           |        Gender
                  Job_Name |         F          M |     Total
---------------------------+----------------------+----------
     Anaesthetisologist    |         6         14 |        20
              Audiologist  |         3          0 |         3
        Burn Specialists   |         1          4 |         5
            Cardiologist   |         8          9 |        17
Cardiothoracic Surgeo      |         0          4 |         4
Dental Surgeon & Orth      |         6          3 |         9
          Dermatologist    |         4          6 |        10
Ear, Nose & Throat Su      |         1          7 |         8
General Surgeon / Gen      |        14         29 |        43
            Geriatrician   |         0          1 |         1
          Haematologist    |         0          1 |         1
Hand and Microsurgeon      |         0          3 |         3
      Internal Medicine    |         3          7 |        10
Liver Transplant Surg      |         2          0 |         2
            Nephrologist   |         1          2 |         3
Neurologist / Neuro S      |         2         11 |        13
Obstetricians & Gynae      |         4         14 |        18
        Ophthalmologist    |         6          8 |        14
Oral Maxillofacial Su      |         0          1 |         1
  Orthopaedic Surgeon      |         2         14 |        16
          Paediatrician    |         7          9 |        16
            Pathologist    |         4          2 |         6
Physiotherapist / Reh      |         3          3 |         6
        Plastic Surgeon    |         2          2 |         4
           Psychiatrist    |         3          4 |         7
            Radiologist    |         3          6 |         9
        Radiotherapist     |         2          4 |         6
         Rheumatologist    |         1          0 |         1
       Speech Therapist    |         2          2 |         4
Sports Medicine Surge      |         1          1 |         2
              Urologist    |         0          5 |         5
Vascular and General       |         0          1 |         1
---------------------------+----------------------+----------
                     Total |        91        177 |       268
```

Fig. 4.7 : A Sample Screen for Cross-Tabulation of Job and Gender

. tabulate hosp_name, gender, row

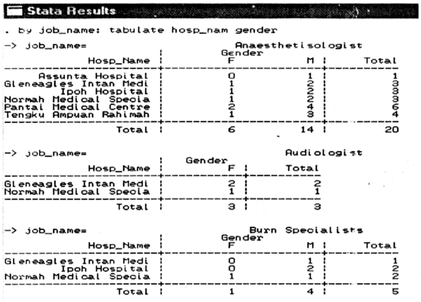


```
Stata Results
. tabulate hosp_nam gender, row

                       |        Gender
             Hosp_Name |         F          M |     Total
-----------------------+----------------------+----------
An Nisa Islamic Medic  |         2          1 |         3
                       |     66.67      33.33 |    100.00
-----------------------+----------------------+----------
      Assunta Hospital |         2          2 |         4
                       |     50.00      50.00 |    100.00
-----------------------+----------------------+----------
Gleneagles Intan Medi  |        17         33 |        50
                       |     34.00      66.00 |    100.00
-----------------------+----------------------+----------
         Ipoh Hospital |        15         35 |        50
                       |     30.00      70.00 |    100.00
-----------------------+----------------------+----------
       Island Hospital |         3          0 |         3
                       |    100.00       0.00 |    100.00
-----------------------+----------------------+----------
Mahkota Medical Centr  |         4          2 |         6
                       |     66.67      33.33 |    100.00
-----------------------+----------------------+----------
Normah Medical Specia  |        11         10 |        21
                       |     52.38      47.62 |    100.00
-----------------------+----------------------+----------
Pantai Medical Centre  |        19         58 |        77
                       |     24.68      75.32 |    100.00
-----------------------+----------------------+----------
Sabah Medical Centre   |         0          6 |         6
                       |      0.00     100.00 |    100.00
-----------------------+----------------------+----------
Tengku Ampuan Rahimah  |        14         21 |        35
                       |     40.00      60.00 |    100.00
-----------------------+----------------------+----------
Timberland Medical Ce  |         0          4 |         4
                       |      0.00     100.00 |    100.00
```

Fig. 4.8 : A Sample Screen for Cross-Tabulation of Hospital and Gender with Row Percentages

iii) Three-way (By group)

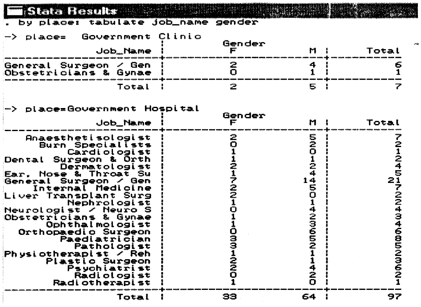
. by job_name: tabulate hosp_nam gender



```
. by job_name: tabulate hosp_nam gender

-> job_name=                      Anaesthetisologist
                         |        Gender
               Hosp_Name |         F          M |     Total
-------------------------+----------------------+----------
         Assunta Hospital |         0          1 |         1
Gleneagles Intan Medi |         1          2 |         3
           Ipoh Hospital |         1          2 |         3
Normah Medical Specia |         1          2 |         3
Pantai Medical Centre |         2          4 |         6
Tengku Ampuan Rahimah |         1          3 |         4
-------------------------+----------------------+----------
                   Total |         6         14 |        20

-> job_name=                      Audiologist
                         |    Gender
               Hosp_Name |         F |     Total
-------------------------+-----------+----------
Gleneagles Intan Medi |         2 |         2
Normah Medical Specia |         1 |         1
-------------------------+-----------+----------
                   Total |         3 |         3

-> job_name=                      Burn Specialists
                         |        Gender
               Hosp_Name |         F          M |     Total
-------------------------+----------------------+----------
Gleneagles Intan Medi |         0          1 |         1
           Ipoh Hospital |         0          2 |         2
Normah Medical Specia |         1          1 |         2
-------------------------+----------------------+----------
                   Total |         1          4 |         5
```

Fig. 4.9 : A Sample Screen for Three-way by Group of Job

. by place: tabulate job_name gender



```
. by place: tabulate job_name gender

-> place= Government Clinic
                         |        Gender
                Job_Name |         F          M |     Total
-------------------------+----------------------+----------
General Surgeon / Gen |         2          4 |         6
Obstetricians & Gynae |         0          1 |         1
-------------------------+----------------------+----------
                   Total |         2          5 |         7

-> place=Government Hospital
                         |        Gender
                Job_Name |         F          M |     Total
-------------------------+----------------------+----------
   Anaesthetisologist |         2          5 |         7
      Burn Specialists |         0          2 |         2
          Cardiologist |         1          0 |         1
Dental Surgeon & Orth |         1          1 |         2
         Dermatologist |         2          2 |         4
Ear, Nose & Throat Su |         1          4 |         5
General Surgeon / Gen |         7         14 |        21
     Internal Medicine |         2          5 |         7
Liver Transplant Surg |         2          0 |         2
          Nephrologist |         1          1 |         2
Neurologist / Neuro S |         0          4 |         4
Obstetricians & Gynae |         1          2 |         3
       Ophthalmologist |         1          3 |         4
  Orthopaedic Surgeon |         0          6 |         6
        Paediatrician |         3          5 |         8
          Pathologist |         3          2 |         5
Physiotherapist / Reh |         1          1 |         2
      Plastic Surgeon |         2          1 |         3
         Psychiatrist |         2          4 |         6
          Radiologist |         0          2 |         2
      Radiotherapist |         1          0 |         1
-------------------------+----------------------+----------
                   Total |        33         64 |        97
```

Fig. 4.10 : A Sample Screen for Three-way by Group of Place

iv) Discrete Variable Histogram

```
. hist area, ylab
```

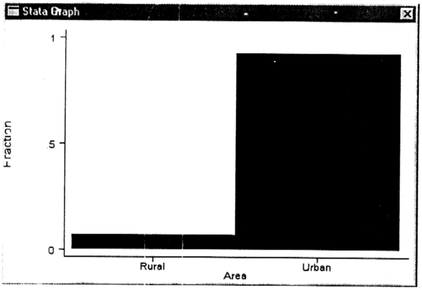


Fig. 4.11 : A Sample Graph of the Ratio of Medical Expertise at Rural and Urban Areas (shown in Histogram)

v) Discrete Variable Histogram by Group

```
. sort hosp_name
. hist gender, by(hosp_nam) b2title (hosp_nam) total ylab
```

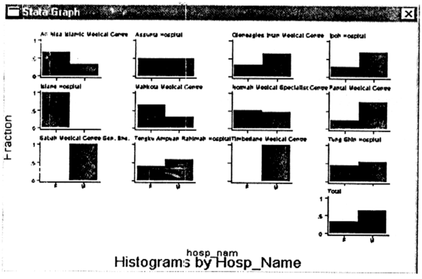


Fig. 4.12 : A Sample Graph of the Ratio of Medical Expertise by Gender at Each Medical Centre (shown in Histogram)

4.4.1.6 Interface Design

The Macromedia Flash 3.0 [Macromedia® Flash™ 3] was used as the interface design for the introductory screen of MediX system. The compelling animations in Flash were easy to use. Moreover, it did not sacrifice the performance or increase the file size. The frmWelcome at both of the administration and user sections use the Flash software.

4.4.2 Integration of Tools

VB was chosen primarily to integrate all of the softwares that were needed in the development of the system. A variety of program types-other than just the standard EXE, such as DLLs and OCXs could be created, too.

4.4.2.1 Integration of Microsoft Access 97

The ODBCDirect functionality provides a truly efficient mechanism for dealing directly with Open Database Connectivity (ODBC) – compliant database engines such as Microsoft Access, which was used in the development of this system. VB 6.0 introduces ADO as the powerful new standard for data access. It supports database connectivity by using the ADO objects whereby the VB application is used to access the database that is built in the Microsoft Access. VB plays an important role by acting as the front-end tool for users to add, modify and delete as well as to display the contents of thc database.

4.4.2.2 Integration of Seagate Crystal Reports

An application called the Crystal Report Designer was used to open the Access database (other database formats can also be handled). The tables and fields of the report that will be used were set. The layout of the report was designed graphically and the report was saved into a separate crystal report file (*.rpt). The crystal report file, then. can be accessed by a VB application by using the crystal report control (*.ocx). But, before that, the dynamic link library 'crpe.dll' must be installed on any computer running crystal reports.

4.4.2.3 Integration of Stata 6.0

The Stata application does not come with the *.ocx file. Therefore, it was integrated in VB by a series of program codes. The program codes were written on a form at the Standard module of VB 6.0.

4.4.2.4 Integration of Macromedia Flash 3.0

The Flash application was used to create the interface design of the MediX system. The layout of the interface was designed using animations and it was saved into a separate Flash Player file format (*.swf). The shockwave control (*.ocx) must be integrated in the VB application to access that file (*.swf).

4.5 Testing

Once the coding and the integration of softwares have been completed, the MediX system is put through a series of testing phases.

Testing has a dual function; it is used to establish the presence of defects in a program and it is used to help judge whether or not the program is usable in practice [Sommerville, 1995]. As Pfleeger states "*A test can be considered successful only when a fault is discovered or a failure occurs as a result of our testing procedures*" [Pfleeger, 1998]. Therefore, testing can only demonstrate the presence of errors. It cannot show that there are no errors in a program.

A number of stages in the testing process were undertaken. They include unit testing, module testing, integration testing and system testing which are discussed in the following sections.

4.5.1 Unit testing

Unit testing is performed to ensure that the individual components are tested and operate correctly as expected. Each unit of the system is tested independently, without other system components by using a set of test cases. A good test case is one that uncovers a large number of errors in the software.

In MediX, the unit testing was performed for all of the data inputs to verify the expected outputs. If the required output was displayed, then the test is free from error. For example, if a duplicate doctor id was entered, then an error message should be displayed. In addition, testing was performed at all of the command buttons, too. If the delete command button was clicked, a confirmation message will be displayed to verify the deletion of that record.

4.5.2 Module testing

A module is a collection of dependent components such as an object class, an abstract data type or some looser collection of procedures and functions [Sommerville, 1995]. A module encapsulates related components so it can be tested without other system modules. During module testing, the analysis of the implemented algorithms and their data usage are given the primary concern. These are determined by items such as statements, sequences of statements, and values of variables and expressions [Linnenkugel et al., 1990].

This testing was carried out on each module of the MediX system. They include add, modify and delete modules for the expertise, hospital/clinic and specialty. Other modules include the report, chart/graph, analysis and user were also tested.

4.5.3 Integration testing

At this stage, all of the individual components are combined together into a working system. In other words, integration testing puts components together in an organized way to help isolate faults as the combined components are tested together. In fact, it is explicitly stated that the goal of integration testing is to check the compatibility between component interfaces.

The integration testing on MediX was carried out on the modules of expertise, report, analysis and the interface design. It is to ensure that the Microsoft Access 97, Seagate Crystal Reports, Stata and Flash are compatible between the components in VB. This is also to ensure that the tools used for the development were correctly integrated for the release of MediX.

4.5.4 System testing

The final testing stage that was performed on MediX was the system testing. In the system testing, the overall behaviour of the system against the functional and the performance requirements were checked. It is the final validation step, and is performed once the whole system is developed [Bertolino, 1997]. This testing process is concerned with finding errors, which result from unanticipated interactions between sub-systems and system components. It is also concerned with validating that the system meets its functional and non-functional requirements. In other words, a system test is a series of different tests designed to fully exercise the system to uncover its limitations and measure its capabilities [Ow and Yaacob, 1998].

4.6 Summary

This chapter mainly discusses the development of a system to keep track of medical expertise. All of the system development activities were discussed in detail at the respective sections. Furthermore, this chapter also investigates the successful integration of the different tools used for the development. It can be concluded that a system with reporting needs, capability of performing analysis using a statistical package, database design, and an easy to use interface is possible to be integrated within a system.