Faculty of Computer Science & Information System
University Of Malaya
50603 Kuala Lumpur

Clinic Information System

NAME : MIFTAH ARFAN BIN SAID
MATRIC : WET000214
SUPERVISOR : MRS. NORJIHAN
MODERATOR : MRS. SRI DEVI
SESSION : 2002/2003

ABSTRACT

This project can be defined as an open system, which attempts to integrate and communicate the outside and inside flow of information in hospitals and provide the functions common for all applications.

The wed-based Clinics information System allows users, using any web browser but best view with Internet Explorer and Netscape Communicator, to search for health treatment, and on the other hand lets the authorized health treatment provider to update patients, medical data. The rationale behind this development is to eliminate the difficulties faced when trying to search through conventional channel such as calling up hospitals or clinics, and flipping through yellow pages.

Waterfall model approach was selected for the development process because it support rapid application development and reduces the risks involved. Software engineering and system analysis principles based on this methodology were applied throughout the development phases: system analysis, design, coding, testing, and implementation.

The development tools selected for this project were Microsoft Visual InterDev 6.0 along side with Microsoft Paint; whereas the back-end database used was Microsoft Access with Active Server Page as the web server. Both server deployed on Window XP server platform.

ACKNOLEDGEMENT

First of all, I would like to thank to Mrs. Norjihan, my ever-patient supervisor whom was a constant and much appreciated source of support and encouragement.

Particular thanks go to my moderator, Mrs. Sri Devi for her ideas and opinions, which led to may improvement and enhancements.

I also wish to mention the valuable contribution of the Dr. Ramli, the owners of Bakti Clinic whom had taken time off busy schedule to have an interview with me. And not forgetting at the person-in-charge at Min Clinic, Dr. Fazilah and Dr. Said from Pusat Rawatan Islam whom was very helpful in assisting me getting the information required.

Also to all my family and course mate whom had lent a helping hand and supporting goes my deepest gratitude.

Miftah Arfan bin Said

Computer Science and Information Technology Faculty April 2003

TABLE OF CONTENT

APPENDICES

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 - INTRODUCTION

CHAPTER 1 - INTRODUCTION

1.1 PROJECT BACKGROUND

Recently, keeping the information of patients, medicines, and others in many government or private clinics involved a huge amount of paper documents, resulting in a lot of overhead and inefficiency. The need of a more efficient system resulted in the development of Clinic Information System.

The Clinic Information System that will be developed is a management tool designed specially to implement efficiency at clinics. The main purpose of the system is to make it easy to manage information at clinics and at the same time to provide needed information to make strategic decisions. Living in the busy and time consuming world right now, the systems allow the users to access the information needed from any where that have the internet connection.

Clinic Information System is going to be a Windows application and using the Active Server Page for the internet connection. Visual Basic 6.0 and Microsoft Access are going to be used for developing the interface and database.

For this project, the system is going to be a web-based database. By using the web-based database, the doctors and staffs can update the information about the patients and others thing easily.

1.2 PROJECT OBJECTIVES

The objectives of this project are:

- Develop a paperless medical information recording
- Create a more systematic and effective way of managing medical information using computer.
- Streamlining the treatment flow of a patient in the clinics, while allowing doctors and other staff to perform to their peak ability, in an optimized and efficient manner.
- Increasing the productivity of the services.
- Prevent the lost of any medical information.
- Latest information can be acquired easily.
- Stock of medicines and others medical equipment can be acquired precisely.

1.3 PROJECT SCOPE

The reason for developing the CIS is to fulfill the entire requirement needed by the doctors and staffs by providing them all of the function that are needed to make the searching, entering or correcting process of data easier by using the computer system. The system will provide computerized management of patient information replacing the old ways of keeping data. The project scope is divided by six main categories for outpatient. That is registration, patient billing, pharmacy, payment collection and maintenance.

a) Clinic Information System

- This is a web-based system that can be accessed from anywhere. All operation involving updating, adding, checking, and deleting of records from database is done by the authorized user only.

- Transactions involve registration, diagnostic, information recording, drugs dispensing and payment.

b) Language

This system is using English language so that is much easier for the records to be transferred to a different clinic or hospital incase needed.

c) User

This system is going be used by all of the staffs in the clinic. The user will be able to add or check patient's record whenever a patient comes to the clinic for treatment or checkup. Because the system is a web-based system, the staffs especially the doctors can update all the information about the clinics any where and anytime. Besides that patients can also check-up the doctor's schedule so they can come to the clinic at the right time.

1.4 IMPORTANCE OF PROJECTS

Healthcare is the most important aspects in human life. This CIS project is developed to help manage patient's information for clinics while allowing doctors and others staff to perform to their peak ability, in an optimized and efficient manner.

1.5 TARGET AUDIENCE

Clinic Owner

This Clinic Information System is intended for clinic owner that offers health care treatment. The System is intended to provide an easier and more systematic way of handling patient's and others medical record. Main processes need to be done using this system is:

- Store patient's information
- Update, delete and add patient's record.

1.6 PROJECT PLANNING

This project begins on the third semester of 2002/2003. Topic confirmation date is 5 March 2003. This project is divided into two phases. The first phase will be executed this semester and the last phase will be executed next semester. The first phase involves System Analysis and System design.

The second phase involves coding and testing process. Coding and testing process will start from 15 June 2003 and ends on 30 September 2003. Documentation process is being done all the way. The table below shows the summary each phases while Gantt chart for overall project schedule.

Phase	Activity
Early Review and System Analysis	- Determine the system objectives - Determine system needs - Provide project schedule - Choose system development model
System Design	- system interface design
3. Execution / Coding	- Learning Visual Basic 6.0 - Learning ASP - Learning Microsoft Access
System Testing	- Design test data - Testing modules - Compare test result with real result
System Maintenance	- Improved changes for system

Table 1.1: Phase Description

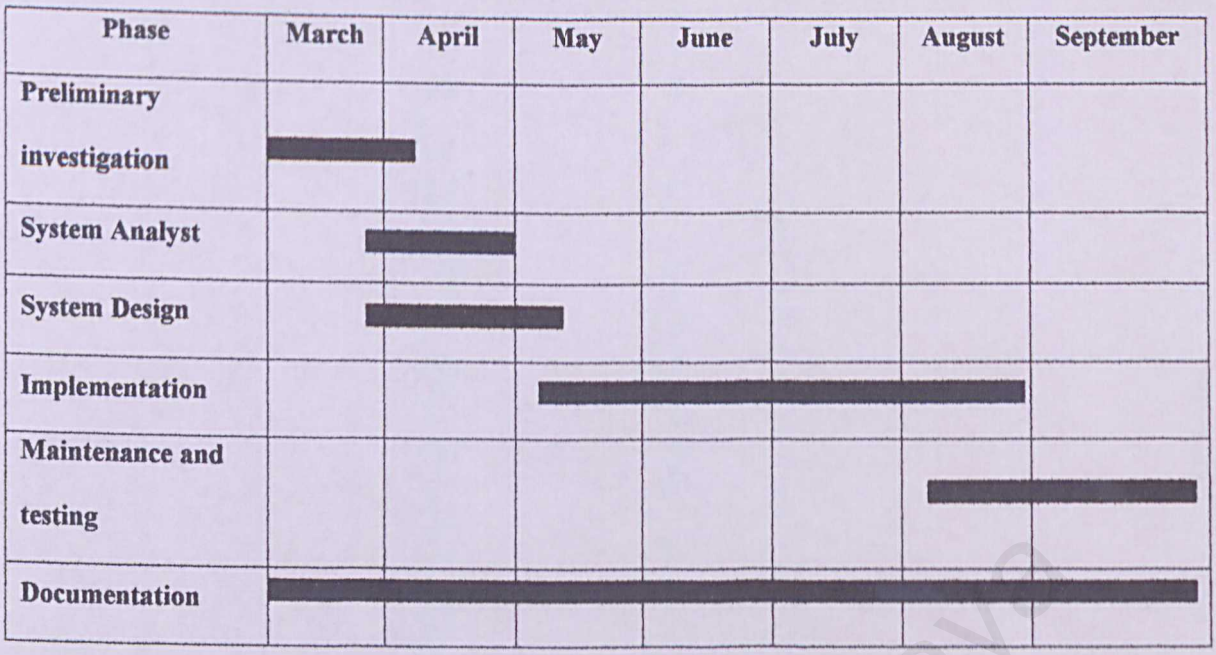


Figure 1.1: Gantt chart for Project Development Plan

CHAPTER 2 - LITERATURE REVIEW

CHARTER 2 - LITERATURE REVIEW

2.1 ROLE OF LITERATURE REVIEW

There are several roles of literature review that has been identified:

i. Collect data that are related to the system that is going to be developed.

ii. Evaluation and reviews of systems that uses the same concept or relevant to the concept has been developed to determine the weakness and the strength of the system and to improve the weakness that been identified.

iii. To get a better idea of the concept involved in the system that is going to be developed while comparing several other existing software that will be used to produce the best result and solution.

2.2 FINDINGS/REVIEW METHOD

Generally, system development process is not complete if there are no collection and review regarding the system that is going to be developed. Accuracy of information is vital in determining whether the system will accomplish in achieving its objectives. Information can be gathered from a lot of sources and every source provides different

type of information depending on the techniques used. Some techniques had been identified to analyze and review the existing system. The techniques are data collection and writing method.

2.3.1 Approach to Literature Review

1. Interviews

This method had been chosen to gain information on how the system that already exists operates. Respondents are people who are already using a Clinic Information System or about to use a Clinic Information System. Informal interviews are also held from time to time to gather more information and opinions on the system that is going to be developed. This helps to identify and understand problems that may arise when the system is developed. Most of the respondents are people who are involved in the medical fields such as doctors, pharmacist, and medical staffs.

2. Observation

Observation is a fact finding technique wherein the systems analyst either participates in or watches a person a person perform activities to learn about the system. Observation is being made by visiting clinics around Petaling Jaya as a guest. Several weakness of the

manual system used by the majority of clinics has been identified. Careful evaluation has been made to compare the systems that exists and how it operates at the clinics.

3. Documentation/ Books and Magazines

Review and analysis had been made on documents and books related to the system that is going to be developed. Data collection from books and magazines is being made to gain extra information from the reviews.

Most of the documents are provided by the clinics and most of the books being used are located in the University Malaya main library.

4. Internet Surfing

Resources from internet help a lot in the success of developing Clinic Information System. Most of the information used as reviews is retrieved from the Internet sources. Several of information retrieved from the internet is being used to compare the advantages and disadvantages of the existing system being used worldwide. Below are a few search engines that been used to search for the related information.

i) http://www.google.com

ii) http://www.yahoo.com

iii) http://www.infoseek.com

iv) http://www.excite.com

v) http://www.lycos.com

2.3.2 Writing Method

1. Document Analysis

Analysis is being made to all data that has been collected and it has been summarized to a more simple form so that it is easier to understand and meets the objective of the system development phase.

2. Comparative

Summary of the results has been made from comparative studies of several systems that already exist. The system that is going to be developed will use the existing system as a guide to develop a better system.

2.4 DEFINITION

System

The word system has several meanings:

(1) A way to get things done

(2) A group of several parts working together for the same purposes

(3) Group of organized opinions to perform something.

Information System

Arrangement of people, data, process information presentation, and information technology that interact to support and improve day-to-day operations in a business as well as support the problem solving and decision-making needs of management and users.

Clinic

Place or institution where people get their can get their health treatment or advice. Usually, the private clinics are owned by one or a group of doctors.

Database

A collection of interrelated files that stored in a medium (usually hard disk).

2.5 WEAKNESS OF MANUAL SYSTEM

Manual system has always been the best method in recording information until information technology era takes place. Although it has been used for centuries there are many weakness of the manual system.

- The problem of losing forms filled by patients often happens.
- It takes time to retrieve a patient's record.
- Patient's medical record will increase by time and more spaces are needed to store these records.
- Data redundancy might be occurred, causing problems in management level.

2.6 ADVANTAGES OF COMPUTERIZED SYSTEM

1. Fast response

Updating and searching process can be done in just seconds.

2. High capacity of data storage

A computer has the ability to store high capacity of information thus eliminate the problem needing more space for string patient's record.

3. Friendly user interface

Easy to use and not take a lot of time to learn to use it.

4. Updating process made easy

The use of computers makes it easier for the administration staff to update records and eliminates the problem of data redundancy.

2.6 SYSTEM THAT EXISTS

Based on information findings, there is several other clinic or health information systems that related to health care. The function of this system will be explained as a guide in developing the Clinic Information System.

2.6.1 PROCARE 2000

PROCARE 2000 is a hospital information system that uses UNIX operating system. This system enables owner to make sure all transactions are recorded and well maintained. This includes recording information of patients, billing, payment collection and drug dispensing. This system is integrated with the Finance and Inventory department that enables the system to automatically order drugs and medicine that are almost out of stock.

The Advantages of PROCARE 2000 are

- Multi level password security.
- Automatically generate receipt when payment is made.
- Automatically add up all charges for the patient including late charges and additional charges.
- The amount need to be paid is deducted from deposit for inpatients
- Support appointment.

The weakness of PROCARE 2000 is the user interface. User interface for systems using UNIX operating system is dull not user friendly. Basically, the interface is complex and it is hard for a beginner user to understand how to use the system. This will have a large impact on training periods and transactions processing by user.

2.6.2 Pantai Medical Care

SCS computer system Sdn Bhd develops this software. It only operates on Window platform. The advantages of this system are

- User friendly interface.
- Automatic reminder due appointments.
- Supports networking.
- Integrated with barcode system
- Several options given to print bills in details and summary.

The weakness of this system is the form that created by the system are too complex making it hard to understand the system. Each form requires a lot of detailed information to be filled by the user. Only experienced user will be able to master the system effectively. New user will have to learn how the system works carefully.

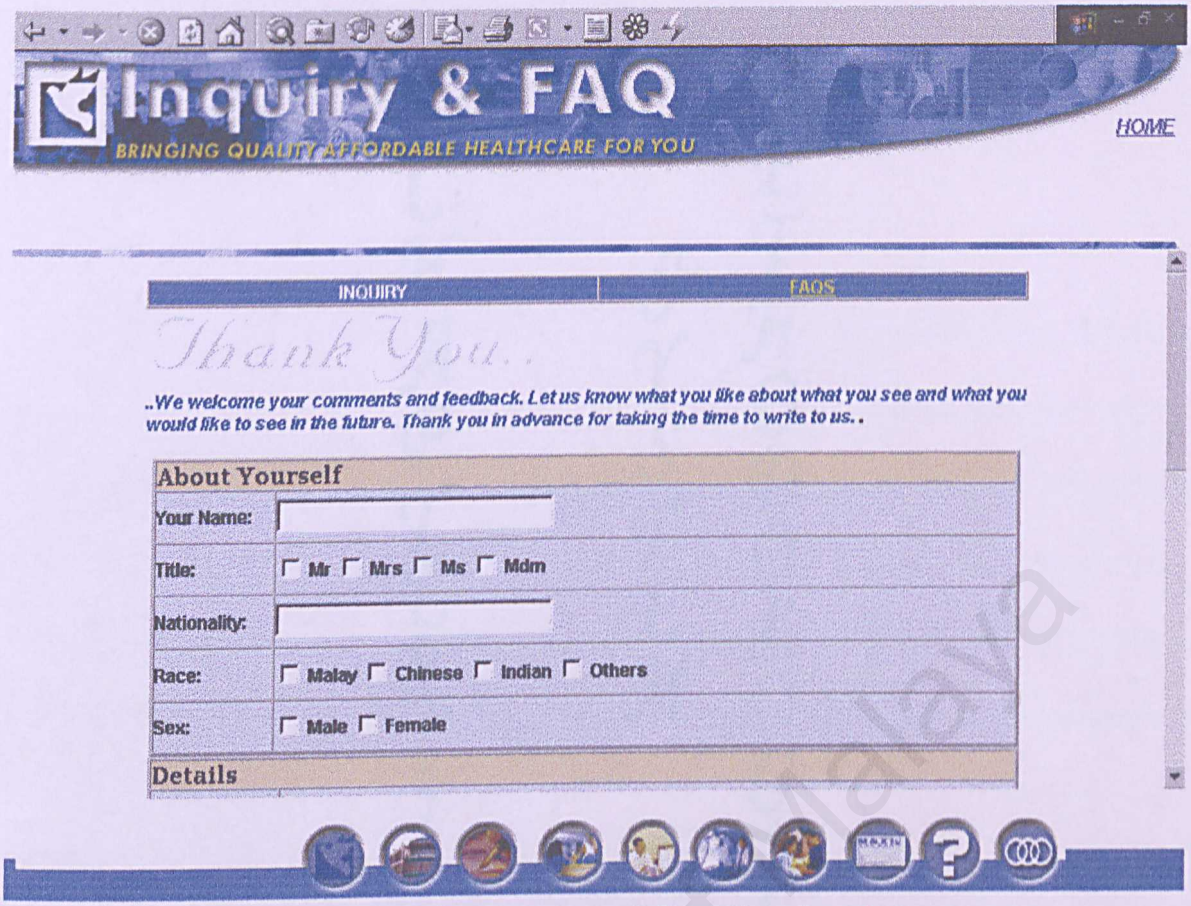


Figure 2.1: Pantai Medical Care User Interface

CHAPTER 3 - SYSTEM METHODOLOGY

CHAPTER 3 - SYSTEM METHODOLOGY

3.1 INTRODUCTION

A systems methodology is a very formal and precise system development that defines a set of activities, methods, best practice, deliverables, and automated tools for system development and project managers to use to develop and maintain most or all information systems and software. System methodologies ensure that a consistent, reproducible approach applied to all projects. Methodologies reduce the risk associated with shortcuts and mistakes. Finally, methodologies produce complete and consistent documentation from one project to the next.

3.2 WATERFALL MODEL

For this project, the system development methodology purposed is a waterfall model. The waterfall model is a useful in presenting high-level view of what goes on during system development, and the sequence of events that are expected to encounter. In waterfall model where stages are depicted as cascading from one to another implies that one development stage should be completed before the next begins.

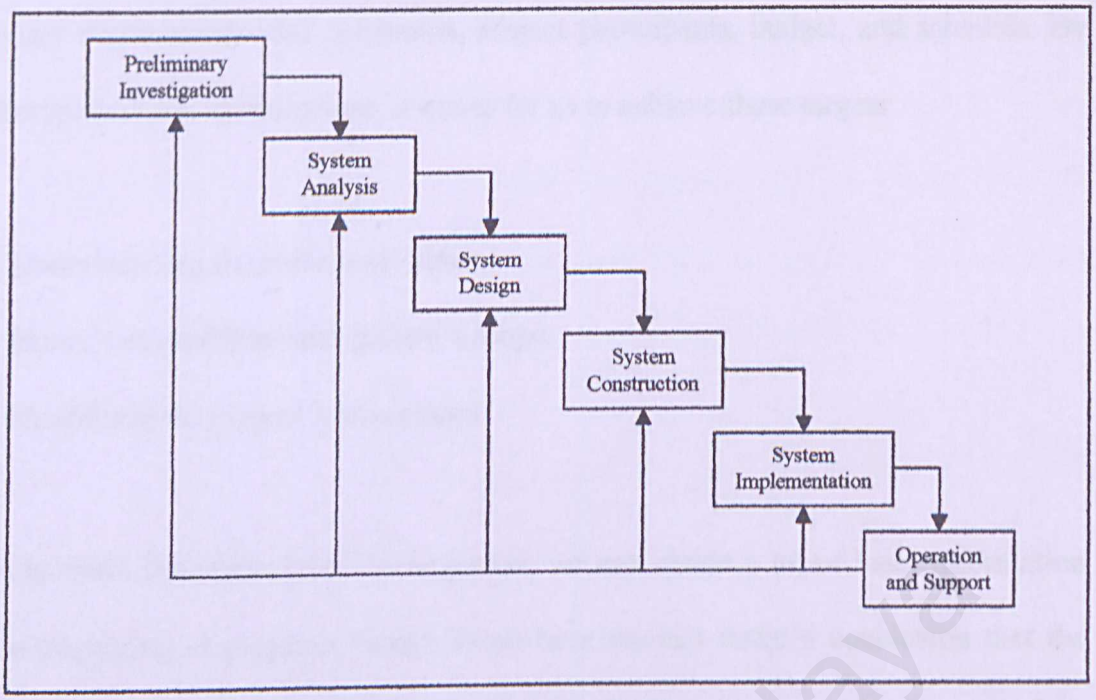


Figure 3.1: Waterfall Model

3.2.1 Phase 1: Preliminary Investigation

The purpose of preliminary investigation phase is twofold. First, it answers the question, "is this project worth looking at?" to answers the question, we must define the perceived problems, opportunities, and directive that triggered the project. Secondly, preliminary investigation phase must also establish the project character that establish scope,

preliminary requirements and constrains, project participants, budget, and schedule. By doing the preliminary investigation, it easier for us to achieve these targets:

1) Understanding the present problems.
2) Identifying problems and project's scope.
3) Identifying the project's advantages.

The results from the preliminary investigation, we can obtain a lot of basic information from the beginning of project's report. From here we can make a conclusion that the system that going to be developed is needed so that all of the aspects and problems below can be solved

1) Performance

Present clinic information's management is using the manual system that includes many physical files that slowing the management process. With this new system, it's hope that the performance of information management can be upgraded.

2) Information

The system that going to be developed can make all the information management job can be done easily, precisely, complete, and right on time.

3) Control

When recording the entire patient's information by using the manual system, there are going to be a lot of redundancy data and collusion happens, making all the data that been produce is not so precise. Because of that, the system is developed to make sure those things never happen.

4) Time and Economy

Because of using the manual method to manage the patient's information is very time consuming, a lot of time and energy are wasted. By using this system, all of the problems above can be solved.

3.2.2 Phase 2: System Analysis

Base on the preliminary investigation phase and literature review at the beginning of the report, a decision can be made, whether the problems can be solved by using the computer system or not. If using the computer system will solve the problem, then we will proceed to this phase. In this phase also we have to identified all the present system's problems, weakness and advantages that it had, and the criteria that must have in the system that is going to be developed. Because of that we had to study and understand all of the important point such as the steps and the workflows that been used in developing the old system so that we can decide the operation steps in the new system. A few

characteristic and element that fits to the user demand must be added in the new system so that the new system will meet the user's needs. The main objectives in this phase are:

1) Analyzing the system requirement.
2) Deciding the system requirement.
3) Documentation of the system requirement.
4) Making decision.

Information that be acquired from phase 1 will be analyzed using the structured analysis methodology such as flow diagram and others. In this situation, the manual system, which is information management used in private clinics had been studied and estimated again. The weaknesses in the manual system are

1) The system is using too much physical file and printed documents that had to be kept, searched, and revised whenever a modification needed. The possibility of data that not be updated even there is a changes have been made make all the useless data still being kept.
2) There are no varieties in data storage techniques making the users feeling bored of using the manual system.
3) There is no specific period of data keeping, updating, and making reports.
4) Manual system that not have a perfect control and not so systematic must not be allowed to exist because the data that have to manage are complex and important.

In analyzing the system requirement, all of the facts that been described above can be concluded, that this manual system

1) Must be done by using the computer using a database management system that can be used easily.
2) The data that being kept must be updated systematically by the administrator and the report must be printed for the user's revision.
3) Only a specific users that can store, update,

So, after all the analyzing that has been made, we know that the manual or old system must be replaced with the new database management system where this system will take care all of the requirement aspects that had been outlined.

3.2.3 Phase 3: System Design

Information system design is defined as those tasks that focus on the specification of a detailed computer-based solution. It is also called physical design. Most of us define the process of design too restrictively. We envision ourselves drawing blueprints of computer-based systems to be programmed and developed by ourselves or our own programmers. Thus we design inputs, outputs, files, database, and others computer components. There are four activities in this phase:

1) Studying the system's needs and requirements.
2) Designing the database file.
3) Designing input system.
4) Designing the output system.

There are many strategies or technique for performing system design. They include modern structured design, information engineering, prototyping, JAD, RAD, and object oriented design. These strategies are often viewed as competing alternative approaches to system design, but in reality, certain combination complement one another.

3.2.3.1 Model Driven approaches

Structured design, information engineering, object oriented design are the examples of model driven approaches. Model-driven design emphasizes the drawing of pictorial system models to document the technical or implementation aspects of new system. Ultimately, the system design models become the blue prints for constructing and implementing the new system. Today, model driven approaches are almost always enhanced by the use of automated tools. Some designers draw system models with general purpose graphic software such as Visio Professional or Corel Flow.

For this project, we are going to use the modern structured design. The modern structured design techniques help developers deal with the size and complexity of programs. Modern structured design is a process-oriented technique for breaking up large program

into a hierarchy of modules that result in a computer program that easier to implement and maintain. Synonyms are top down program design and structured programming.

The concept is simple. Design a program as a top-down hierarchy of modules. A module is a group of instructions – a paragraph, block, subprogram, or subroutine. The top-down of these modules is developed according to various design rules and guidelines. Structured design is considered process-oriented techniques because its emphasis is on the process building block in our information system- specifically software process. Structured design seeks to factor program into the top down hierarchy of modules that have the following properties

1) Modules should be highly cohesive; that is each, modules should accomplish one and only one function. This makes the modules reusable in future programs.
2) Modules should be loosely coupled; in other words, modules should be minimally depending on one another. This minimizes the effect that future changes in one module will have on other modules.

3.2.3.2 Top- down Methodology

There are six steps that must be followed in using this methodology before a conceptual design can be developed. The steps are

1) Deciding the entity set and relationship set that include in the scope research. Beginning with the important entity, followed by the other entities.
2) Deciding the attribute set that had a connection to entity set.
3) Deciding the attribute set for the relationship set.
4) Choosing the primary key in every entity.
5) Deciding the domain for each entity.
6) Combine all the entity set, relationship set, and attribute graph to form a complete conceptual graph.

3.2.3.3 System Design Specification.

The purpose of this first design task is to specify application architecture. Application architecture defines the technologies to be used by one, more or all information systems in terms of its data, process, interface, and network components. This task is accomplished by analyzing the data models and process models that were initially created during requirement analysis. Given the data models, process models, and target solution, distribution decision will need to be made. As decisions on how data, process, and interface are made, they are documented. An example is the physical data flow diagram (PDFD) that is used to establish physical process and data stores (database) across a network.

The system users may involved in this activity to help address business data, process, and location issues. The key inputs to this task are the facts, recommendations, and opinion

that are solicited from various sources and the approved system proposal from analysis phase. The system design for the Clinic Information System is divided into three main parts that are

1) System structure design
2) Database design
3) Interface design

System structure design

For designing the system structure, the structured chart had been used. This is because this chart will show all the relationship between the modules.

Database design

Typically the next system design task is to develop the corresponding database design specification. The design of data goes far beyond the simple layout of records. Databases are a shared resource. Many programs will typically use them. Future programs may use databases in ways not originally envisioned. Consequently, designer must be especially attentive to designing the databases that are adaptable to future requirements and expansion.

We must also analyze how programs will access the data in order to improve performance. We may already be somewhat familiar with various programming data structures and their impact on performance and flexibility. These issues affect database organization decisions. Other issues to be addressed during database design include record size and storage volume requirement. Finally, because databases are shared recourses, we must also design internal controls to ensure proper security and disaster recovery techniques, in case data is lost or destroyed.

The purpose of this task is to prepare technical design specifications for a database that will adaptable to future requirement and expansion.

Interface design

Once the database has been design and possibly prototype built, the systems designer can work closely with system users to develop input, output, and dialogue specifications. Because end-users and managers will have to work with inputs and outputs, the designer must be careful to solicit their ideas and suggestions, especially regarding format. Their ideas and opinions must also be sought regarding and easy-to-learn and easy-to-use dialogue for the new system.

For inputs, it is crucial to design the data capture method to be used. For instance, we may design a form on which data to be input will be initially recorded. We want to make it easy for the data to be recorded on the form, but we also want to simplify the entry of

data from the form into the computer or onto a computer-readable medium. This is particularly true if the data is to be input by people who are not familiar with the business application.

For interface or dialogue design, the design must consider such factors as terminal familiarity, possible errors and misunderstandings that the end-user may have or may encounter the need for additional instructions or help at certain points, and screen content and layout.

The system users should be involved in this activity. The inputs, outputs, and interface dialogues are what they will and work with. The degree to which they are involved is emphasized in design efforts that involve prototyping. They were asked to provide feedback regarding each input/output prototype. The interface design that designed for the users must have these following characteristic

1) User Friendly

Have all the commands, error display, and help service to make the users understand and use the program easily.

2) Object Oriented

The graphic display must be attractive and simple to attract the users to us it.

3) Consistent

All of the operation must be used the same key. The usage of same key will make the users to remember the key easily.

3.2.4 Phase 4: Construction

This phase will be executed after all the system designs had been done. System construction is the development, installation, and testing of system components. The purpose of the construction phase is to develop and test functional system that fulfills business and design requirements and to implement the interfaces between the new system and existing production systems.

Programming is generally recognized as a major aspect of construction phase. In this section we will discuss several tasks involved in the construction phase of this system development project.

1) **Built and test Database**

Building and testing database are unfamiliar tasks for many students, who are accustomed to having and instructor to provide the with the test databases. This task must be immediately preceding other programming activities because databases are resources shared by the computer programs to be written. If new or modified databases are required for the new system, we can now build and test those databases.

2) Install and Test New Software Packages

Some systems solutions may have required the purchase or lease of software packages. If so, once databases for the new system have been built, we can install and test the new software. This new software will subsequently be placed in the software library.

3) Write and test new programs

The primary inputs to this activity are the technical design statement, plan for programming, and test data developed during the system design. This activity also results in program documentation that may need to be approved by the quality assurance group.

Testing is an important skill that is often overlooked in academic course on computer programming, testing should not be deferred until after the entire program has been written. There 3 levels of testing to be perform that are

1) Sub testing

Sub testing is the test performed on individual events or modules of a program. In other words, it is testing of an isolated subset of program.

2) Unit or program testing

Is the test whereby all the events and modules that have been coded and stub tested for a program are tested as an integrated unit; it is the testing of the entire program.

3) Systems testing

Systems testing ensure that application programs written and tested in isolation work properly when they are integrated into the total system.

Just because a single programs works properly doesn't mean that it works properly with others programs. The integrated set of programs should be run through a systems test to make sure one program properly accepts, as input, the output of other programs. Once the system test is complete and determined to be successful, we can proceed to the implementation of the system.

3.2.5 Phase 5: Implementation

The functional system from the construction phase is the key input to the implementation phase. The deliverable of the implementation phase is the operational system that will enter the operation and support stage.

In our information system framework, the implementation phase considers the same building blocks as the construction phase.

3.2.6 Phase 6: System operation and support

System support is the ongoing technical support for users, as well as the maintenance required to fix any errors, omissions, or new requirements that may arise. Before an information system can be supported, it must first be in operation. System operation is the day-to-day, week-to-week, month-to-month, year-to-year execution of information system's business process and application programs.

CHAPTER 4 - SYSTEM ANALYSIS

CHAPTER 4 – SYSTEM ANALYSIS

4.1 INTRODUCTION

System analysis is a problem solving technique that decomposes a system into its component pieces for the purpose of studying how well those component parts work an interact to accomplish their purpose. Some of the aspects that important in system analysis are:

- Services that the system provide.
- The objective of the system that is going to be developed.
- The need of understanding about the software and hardware that are going to be used.

In system analysis phase, we need to:

- Identify the users need.
- Deciding the system concept
- Deciding the tools that are going to be used in developing the system.

4.2 STRUCTURED ANALYSIS

Structured analysis was one of the first formal approaches for systems analysis of information systems. In its various dialects, it is still one of the most widely practiced approaches.

Structured analysis is a model-driven, process-centered technique used to either analyze an existing system, define the requirements for a new system or both. The models are pictures that illustrate the system's component pieces: process and their inputs, outputs, and files.

4.3 PRESENT SYSTEM WEAKNESS

From all the interviews and observations that had been done with doctors and clinic's staffs, the file or manual system that is being used practically have many weakness and giving a lot of problem to the management of the clinic. The computer system needed to overcome this problem. The result from this interview has lead to a few conclusions and solutions. First of all, the doctors need to identify the treatment that been given before to one patient before or the records about past treatment records easier. Secondly, all of the records about payment bills, drugs and medical tools quantity, and others things can be acquired easily and precisely. The last one is the safety of the records that they keep, preventing from an unauthorized person to access it.

4.4 SYSTEM REQUIREMENT

To make sure that the system fully operated in an optimum level, this section will discuss about the tools, hardware, and software required supporting the system.

4.4.1 Functional Specification

This function will explain the interaction between the system and system's environment. Below are the function specifications:

a) Patient Module

- new patient registration
- registered patient
- search patient
- patient's record

b) Pharmacy Module

- new medicine
- list of medicine
- medicine stock
- search medicine

- medicine suppliers

c) Administration Module

- add users
- list of users
- list of panel
- change password
- report:
    - charge
    - income
    - list of supplier
    - medicine and medical kit that need to be order
    - quantity of current medicines

d) Help Module

4.4.2 Specification Non-functional

Reliability

A reliable system is a system that has minimum downtime and high information integrity. It does not produce dangerous or costly failures when it is used in a reasonable manner which a typical user expects it as normal. Reliability is the extents to which a system can be expected to perform its extended function with required precision and accuracy.

Modularity

Modularity is a key factor in good program design. The working of the system was broken into modules so that distinct functions of objects could be isolated from one another. This characteristic makes testing and maintenance easier.

Usability

The system should be developed in such way that is easy to use and user friendly, so that users can interact with the system comfortably and effectively. Visual effects and meaningful images and icons are used to provide the system with a sophisticated and yet simple to use user interface. Meaningful captions and menu options will simplify user communication with the system.

Confirmation message for a non-trivial process such as updating a record should be displayed. Effective error handling and validation procedures will also help the users to use the system. If an error occurs, such as invalid data input and invalid password, the system should display and error message to the user.

Simplicity User-Friendly

Simplicity refers to keeping forms and screens properly uncluttered in a manner that focuses on the user's attention. Features and functions are developed in a self-explanatory method. The screen layouts are tailored to suit the users need and preference. User-friendliness is of utmost importance. Easy understandings are requires less learning time is two factors that optimized the utilization of the system.

Security

The system should be equipped with sufficient security. Each access should be authenticated by the system where users must login with the correct user id password to prevent an authorized access into the system. The system should not show any potential of leakage of information. The password should be encrypted.

Efficiency

Efficiency is a process or procedures that can be called or accessed in unlimited number of time to produce similar outcomes or outputs of creditable pace or speed.

Understandability

It is a degree of self-descriptiveness. The system should contain enough information for the users to determine its objectives, assumption, constrain, inputs, outputs, and states.

Maintainability

A product is maintainability if the system programs are easily modified and tested in the case of updating process to meet the new requirement, correcting errors or move to a different computer system.

4.4.3 Hardware Requirement

Hardware Requirement	Minimum	Suggestion
CPU	Pentium 1	Pentium 3
RAM	8MB	32MB/64MB
Hard disk's space for data	10MB	10MB
Hard disk's space for software	10MB	10MB
Monitor	VGA	SVGA
Input Device	Mouse and keyboard	Mouse and keyboard
Output Device	Dot matrix printer	Bubble jet printer
Hard Disk	2GB	10GB
Color Display	16 Bit	32 Bit
Operating System	Windows 95	Windows XP

Table 4.1: System Hardware Requirement

4.4.4 Software Requirement

Microsoft Visual Basic 6.0

After examine the capability of the system and all of their complex function, good development software has to be chosen. For that, the main programming software that is going to be used to develop the system is Visual Basic 6.0. This software is using

Microsoft Access for its database. The reason of choosing this software is because it is easy to use and suitable for developing the system that has an attractive user interface. Others than that, it also allowed control function such as buttons, checkboxes, edit boxes and others controls function that help the application development and usage.

Active Server Page (ASP)

ASP is a server-side scripting environment for creating dynamic WEB pages or building other interactive Web applications. ASP pages are files that contain HTML tags, text, and script commands. They can call Active X components to perform tasks such as connecting to a database or performing calculation. ASP lets developers add interactive content to web pages or build entire web applications that use HTML pages as the user interface. ASP has four important features that make it unique.

1) It contains server-side scripts. By including the server-side scripts in it, we can create web pages with dynamic content.
2) It provides a number of built in objects. These objects allow us to both retrieve information from and send information to the browser.
3) It bundled with a number of standard server-side Active X components. These components allow us to do such things as determine the capabilities of different web browser or include a page counter on a web page.

4) It can interact with a database such as Microsoft SQL server. By creating Active server Pages that can interact with a database, we can create very advanced web sites.

Microsoft package

The Microsoft packages that are going to be used are Microsoft Office XP and Microsoft Paint. Microsoft Office XP is on of the office application package that includes a few applications such as word processor, spreadsheet, presentation package, scheduling task, and others. The application that included in Microsoft Office XP is

1) Access
2) Binder
3) Excel
4) PowerPoint
5) Scheduler
6) Word

Microsoft Access

Microsoft Access is being used to develop and design the database for this system. The database that had been developed will be imported to Microsoft Visual Basic application to create electronic forms. In this environment, system users will access, display and print the data from this database.

Microsoft Excel

Microsoft Excel is being used to create and draw tables.

Microsoft PowerPoint

Used as a medium to help the changing process of graphic in GIF or JPEG format to others form other that Bitmap. It is being used to create icons or displays.

Microsoft Word

Microsoft Word is used to create help index for users in this system. Files that being documented are going to be compiled in Visual Basic Tool to generate a help system to the users.

Microsoft Paint

Same as Microsoft PowerPoint, Microsoft Paint is used to manipulate the graphic image to create images in Bitmap format. This package support types of images for 256 Bitmap, 16 color Bitmap, 24-bit Bitmap and Monochrome Bitmap.

CHAPTER 5 - SYSTEM DESIGN

CHAPTER 5 - SYTEM DESIGN

5.1 INTRODUCTION

Design is a meaningful engineering representation of something that is to be built. Design focuses on four major areas of concern; architecture, data interfaces, and components. The main aim of this phase is to transform all the requirements analyzed in previous phase – The system Analysis phase, into system characteristics.

5.2 ARCHITECTURAL DESIGN

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

For this project, the architectural design is based on modular decomposition approach. Decomposition is a structured system approach. It is a top-down approach that based on assigning functions of components.

The designer begins with a high-level description 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 approach, the system development begins from high level description and then moves down to a low level description.

As shown below, the clinic information system consisted of four main modules,

- Patient module
- Administration module
- Pharmacy module
- Help module

5.3 DATABASE DESIGN

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

Thus, the design of the database is very important because it can affect greatly on the performance of data retrieval, updating, and query as well in the run-time period of the system.

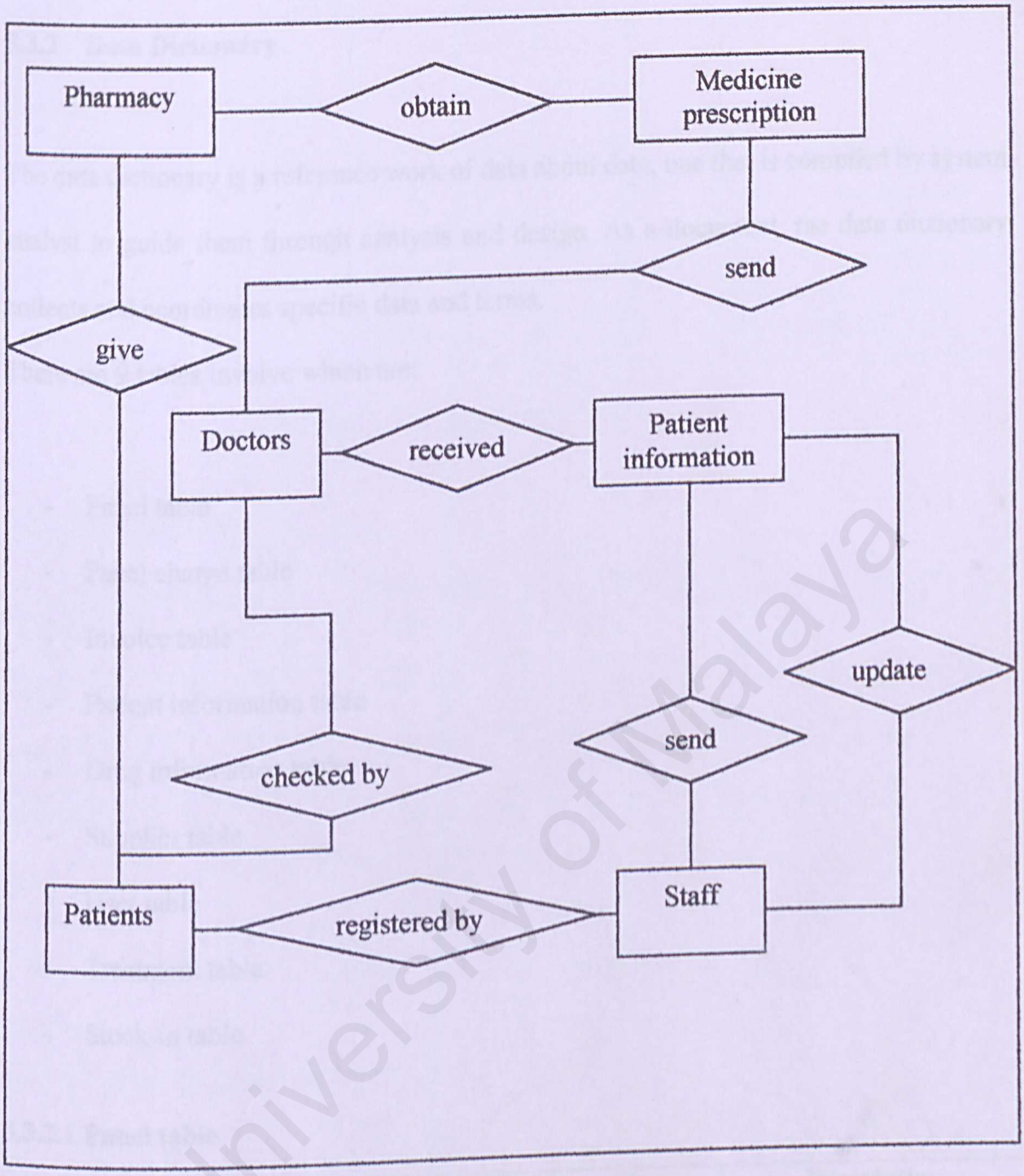


Figure 5.1: Entity-Relationship of Clinic Information System

5.3.2 Data Dictionary

The data dictionary is a reference work of data about data, one that is compiled by system analyst to guide them through analysis and design. As a document, the data dictionary collects and coordinates specific data and terms.

There are 9 tables involve which are:

- Panel table
- Panel charge table
- Invoice table
- Patient information table
- Drug information table
- Supplier table
- User table
- Treatment table
- Stock-in table

5.3.2.1 Panel table

Field Name	Data type	Size	Description
Company	text	30	Name of the company
Address	text	60	Company's address
Telephone	number	10	Company's phone number
Contact_p	text	30	Company's contact person

Table 5.2: Panel Information

5.3.2.2 Panel Charge Table

Field Name	Data type	Size	Description
Company	text	30	Name of the company
Date	number	10	Date of the charge that been made
Charge	currency	10	Total payment

Table 5.3: Panel Charge Information

5.3.2.3 Invoice Table

Field Name	Data type	Size	Description
NoInvoice	number	10	Invoice number for patients
Date	number	10	Date of receipt received
Name	Text	30	Patient's name
Description	Text	20	Drug's name
Price	currency	10	Drug's price

Table 5.4: Invoice Information

5.3.2.4 Patient Information Table

Field Name	Data type	Size	Description
PatientNo	Text	15	Patient's reference number
Panel	Text	30	Patient's company
Name	Text	30	Patient's name
IC	number	14	Patient's IC number (new)
Address	Text	60	Patient's address
Telephone	number	10	Patient's phone number

Sex	Text	6	Patient's sex
Occupation	Text	30	Patient's occupation
Name2	Text	30	Patient's nearest contact person's name
IC2	number	14	Patient's nearest contact person's IC
Relationship	number	10	Patient's relationship wit the contact person

Table 5.5: Patient Information

5.3.2.5 Medicine Information Table

Field Name	Data type	Size	Description
DrugCode	Text	15	Code of the drug
Description	Text	60	Drug's information
Dosage	Number	10	Drug's dosage
Warn	Text	50	Drug's usage warning
Quantity	Number	5	Drug's quantity in stock
MinQuantity	Number	5	Drug's minimum quantity
Supplier	Text	30	Drug's supplier's name
CostPrice	Currency	10	Drug's cost price
SellPrice	Currency	10	Drug's selling price

Table 5.6: Medicine Information

5.3.2.6 Supplier Table

Field Name	Data Type	Size	Description
Supplier	Text	30	Supplier's name
Address	Text	60	Supplier's address
Telephone	number	10	Supplier's phone number

Table 5.7: Supplier Information

5.3.2.7 User Table

Field Name	Data Type	Size	Description
User	Text	30	User's name
UserID	number	10	User's ID number
Unit	text	20	User's unit
Password	text	10	User's password

Table 5.8: User Information

5.3.2.8 Treatment Table

Field Name	Data Type	Size	Description
PatientNo	number	15	Patient's reference number
InvoiceNo	number	10	Patient's invoice number
Disease	text	100	Patient's disease
Treatment	text	100	Treatment for the disease
Drug	text	30	Drugs that used for the treatment

Table 5.9: Treatment Information

5.3.2.9 Stock-In Table

Field Name	Data Type	Size	Description
Date	text	10	Supplier's name
DrugCode	text	15	Code of the drug
Quantity	number	5	Drug's quantity
Cost	currency	10	Drug's cost price

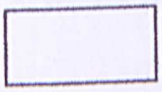
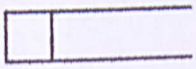
Table 5.10: Stock-In Information

5.4 PROCESS DESIGN

5.4.1 Data Flow diagram

Data flow diagram (DFD) is a technique used to show the graphical characterization of the data process and flows in system. The DFD gives an overview system inputs and outputs, process and the flow of data through each process.

Here are the basics symbols of a data flow diagram

Symbol	Definition
	Transformation of data into other data
	Sources and destination of data
	Data in static storage
	Data on the move

T

Table 5.11: DFD Object

5.4.1.1 Context Diagram

This is the context diagram for the clinic information system.

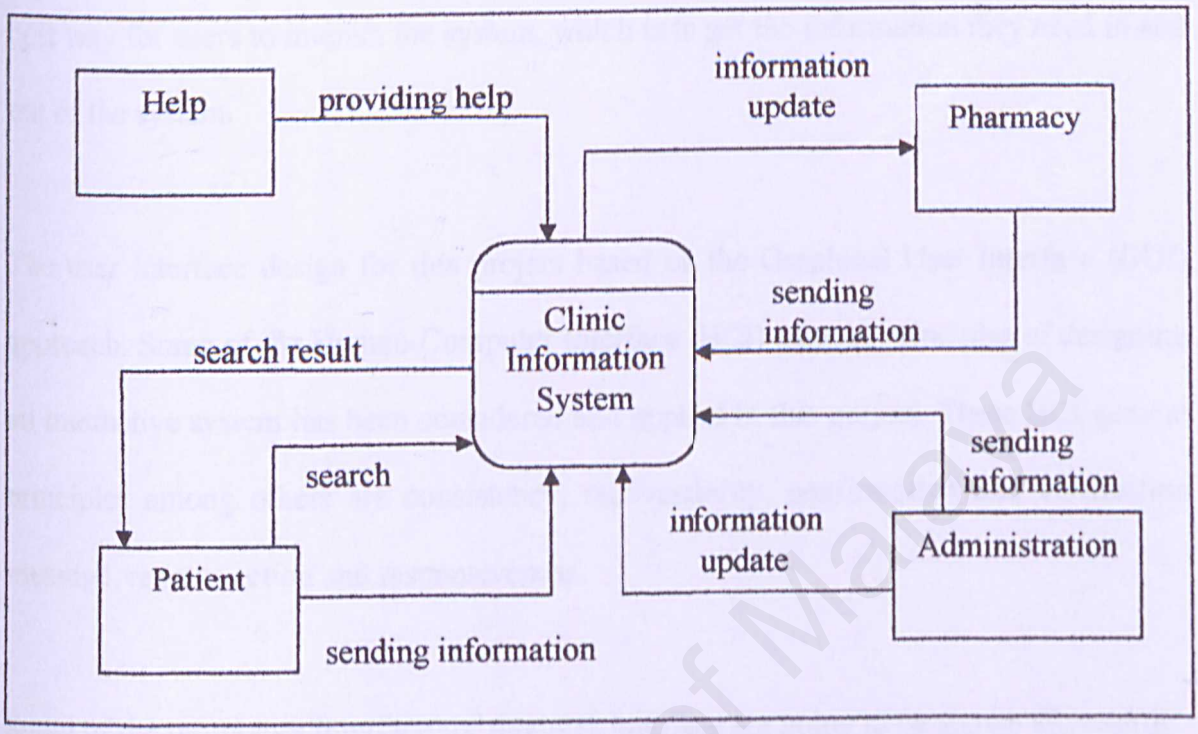


Figure 5.2: Context Diagram for Clinic Information System

5.5 User Interface Design

The interface is the system for most users. The goal of interface design is to provide the best way for users to interact the system, which is to get the information they need in and out of the system.

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

Some of the prototypes interface for modules interface are going to be shown afterwards.

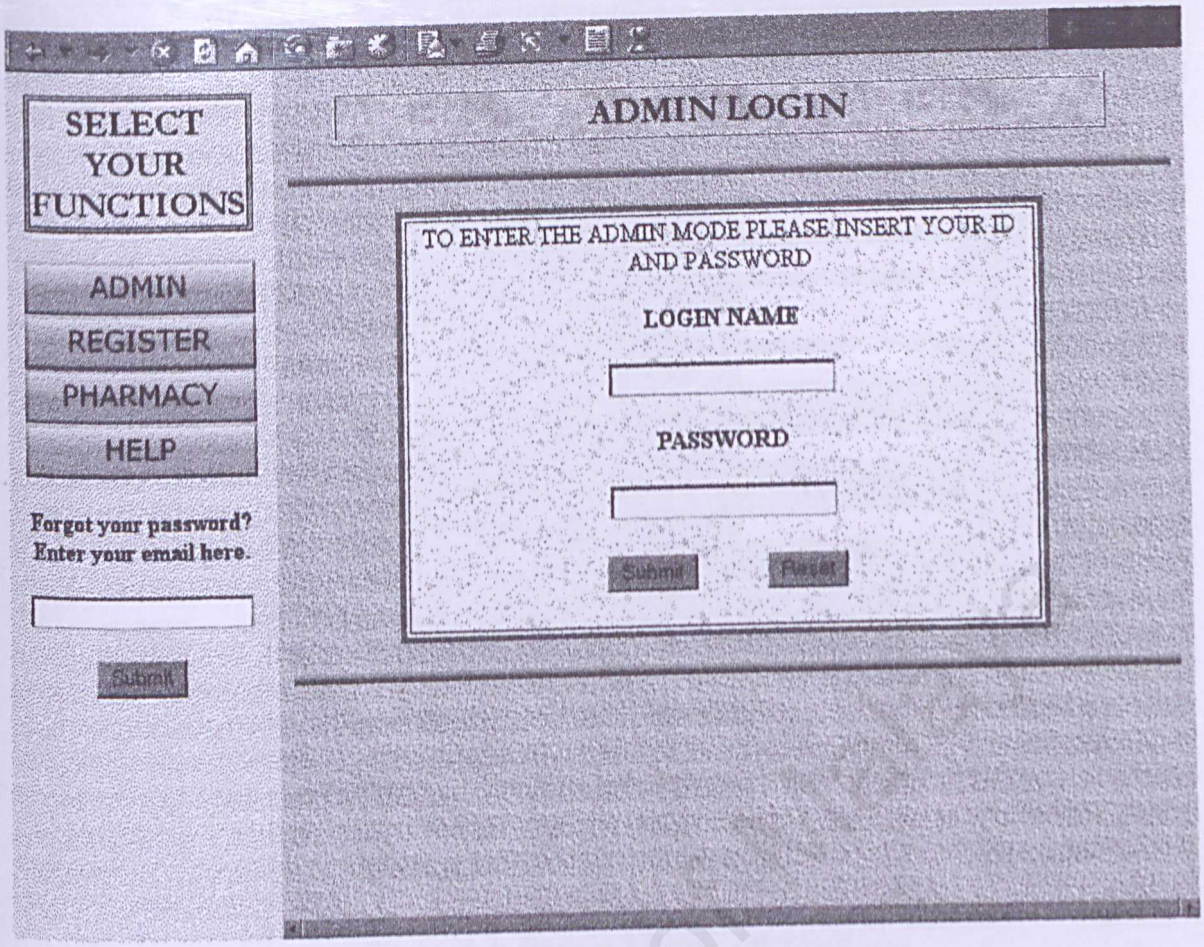


Figure 5.3: Administration Module Interface

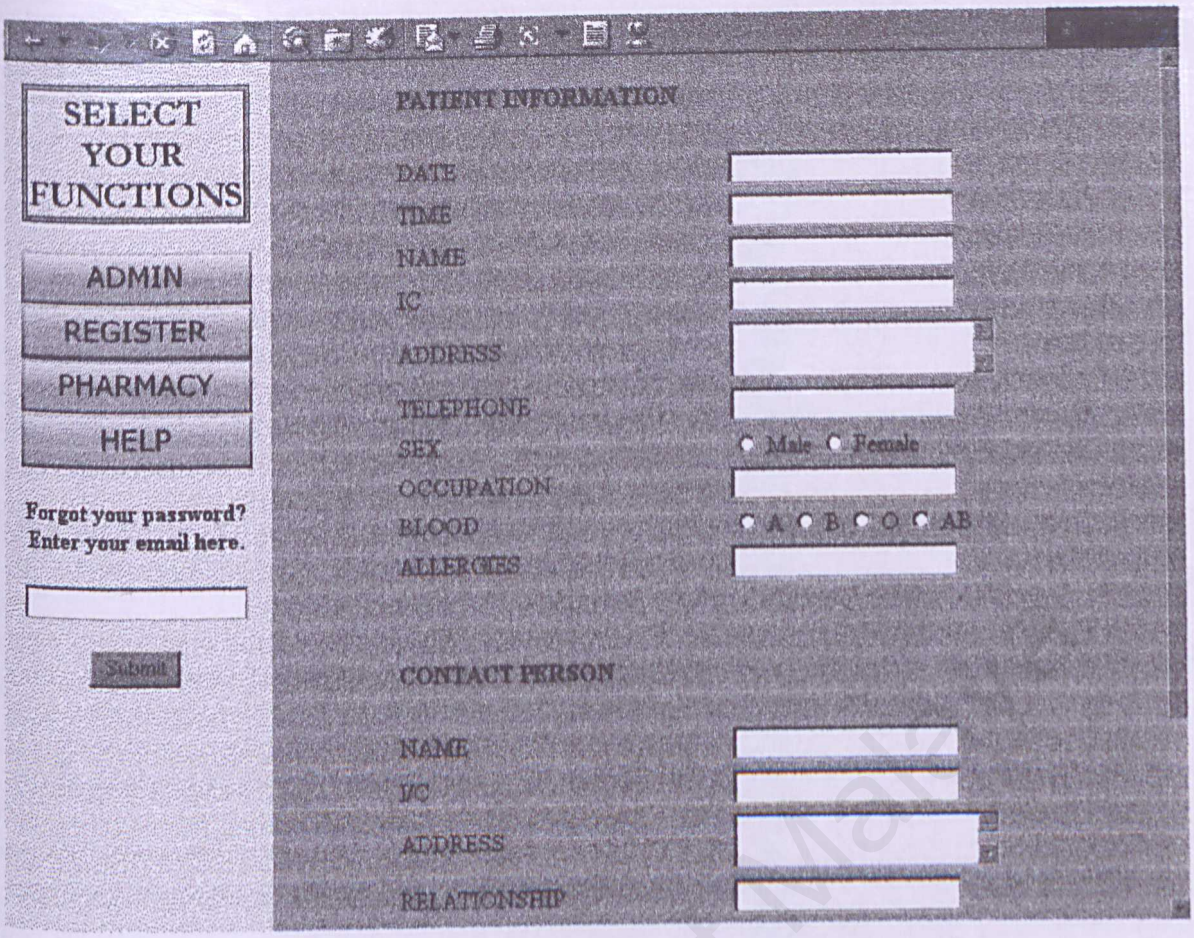


Figure 5.4: Patient Module Interface

CHAPTER 6

SYSTEM IMPLEMENTATION

CHAPTER 6 –SYSTEM IMLEMENTATION

6.1 INTRODUCTION

System implementation is a process that converts the system design and requirement into a programming code. It involved coding step that translates a detail design representation of software into a program language realization.

6.2 DEVELOPMENT ENVIROMENT

The development environment has certain impact on the development of system. Using a suitable hardware and software not only to speed up the system development but also determine the success of the project. The hardware and software tools used to develop the entire system are as shown at the next page.

6.2.1 Hardware Requirements

Server	Workstations
- Running on Windows 2000 Server, Microsoft SQL Server 7.0, and Internet Information Server 5.0. - Consist of 256MB RAM with two hard disk. - Alongside a Pentium 4 processor 2.5 GHz and NIC (Network Interface Card) of Ethernet 10/100Mbps speed.	- Running on Windows XP or Windows 2000 Professional, Visual InterDev 6.0, Internet Explorer 5.0 and Internet Information Server 5.0. - Consists of 256MB RAM with 20 GB hard disk. - Alongside a Pentium 4 processor 2.0 GHz and a NIC of Ethernet 10/100Mbps speed. - Other standard PC component.

6.2.2 Software Tools/Components Requirements

In the development of Clinic Information System, the software applied basically consisted of components and tools. The components included all the technology used to support the functionality of the system such as....

6.2.2.1 Descriptions of Development Applications/Tools

The application and tools categories that been used to develop are as listed below:

1. **Application and coding tools**

   - Ms Visual InterDev 6.0

     Creates and refines web pages for the whole system.

   - Ms Internet Information Server 5.0

     Maps local directory to virtual directory. Creating a local web site.

2. **Database Implementation Tools**

   - SQL Query Analyzer

     Generate Database tables using SQL statements.

   - SQL 7.0 Enterprise Manager

     View and edit tables created in the database. View relationships between the tables.

3. **Graphic/Interface Modeling Tool**

   - Microsoft Paint

     Create a simple image/graphic

   - Macromedia DreamWeaver, FrontPage

     Ease the task creating and editing the web page interface design.

6.3 INSTALLATION AND SETUP

Server and development tools installations are the very first step before starting off with any development work. When using Microsoft's products, it is essential to know the sequence of products installations to ensure smooth execution without system errors.

The sequences of the installation process on the server are:

1. Windows XP
2. Microsoft SQL Server 7.0

And for the workstation are:

1. Windows XP.
2. Microsoft SQL Server 7.0
3. Microsoft Visual Studio including Visual Interdev 6.0
4. Macromedia DreamWeaver.
5. Microsoft Office XP

6.3.1 Create Virtual Server

Personal Web Server (PWS) provides a feature that allows web content to be organized, by using virtual servers. All the asp files, and other relevant files, have to be stored in a directory that is mapped to the server. The following are the steps to create virtual servers.

- Open Microsoft Management Console, and then right click on the physical PWS server (small computer icon).
- From the shortcut menu, click New Web Site.
- Enter a name to describe the web site.
- Next, specify a location of the home directory, which is virtual server's main directory and contains the web site's home page.
- Finally specify the access, permissions for the virtual server.

6.3.2 Create Database

To create a new database by using the Microsoft Access, simply click at MA button. Then choose create new database. There is also a database wizard to help user to create the database. Name the database (for this project the name is clinic).

After that we can start creating the table for the database. There are 2 ways to create the table, which are using the Table Wizard or create it manually by using the Design View.

6.3.3 Create Data Source Name (DSN)

Before connecting to data source like Ms Access, A DSN is needed in order to open connection to the database server. DSN is a method of standardizing database connectivity. There are two types of DSN, System DSN and User DSN. A System DSN tends to be more convenient but less secure then a User DSN. If multiple users are using the same client computer and there is a need to hide the existence of the data source from some of the users, User DSN will do the job right.

Below is a summary of steps taken to create a system/user DSN:

1. On the start menu, click Settings, then click control panel.
2. Double click ODBC to open the driver manager's user interface, the ODBC Data Source Administrator dialog box.
3. Click the System DSN/User DSN tab.
4. Click Add.
5. From the installed ODBC drivers list, select Microsoft Access Driver and click ok to bring up the ODBC Server Setup dialog box.
6. Give the user data source a name, for an example clinic database for this project. This can be either the name of the server to which the user is connecting, or the name of the database to which the user are connecting or some other name that is meaningful to the user.
7. The Description field is optional.

8. Enter the name of the Access Server in Server box. The Access Sever should have the same name as the Window XP-based computer on which it is running.
9. Click finish at the last screen, the user do not has to change the default setting. Then click on Test Data Sources in the dialog box that pop-up to ensure thet the data sources that the user created is functioning.
10. Click OK to return to the ODBC Data source Administrator Dialog box, and click OK again.

6.3.4 Objects Coding

For the objects, only those which are essential and vital to the project's smoothness were chosen and explained as detailed as possible, as not all components in the system can be explained in full details in this concise report. Some of the codes will be appended to Appendix A.

By default, VBScript is the chosen language for scripting ASP objects. But it is still perfectly legal to mix language, as long as they are properly specified for each section of code in the application's page (web page).

There are three unique ways to indicate that blocks of script are to be executed on the server (that is the server-side code). One method is to surround the scripts with the <%

and %> tags. In this case, any text between these tags is treated as server-side scripting commands, based on the language defined at the beginning of ASP.

Several useful built in server objects provided by ASP that will help the application read requests form HTML forms, post results to the web browser, control the server, and others. It is easy to include these server-side objects within the ASP scripts and there is no need to declare them or initialize them. These server-side objects are Application, Request, Response, Server, and Session.

6.3.4.1 Data Connection

As for the first step in adding or retrieving data from database, a connection between the web server and database must be established.

```
<%
' FileName="Connection_odbc_conn_dsn.htm"
' Type="ADO"
' DesigntimeType="ADO"
' HTTP="false"
' Catalog=""
' Schema=""
Dim MM_newpatient_STRING
MM_newpatient_STRING = "dsn=clinic database;"
%>
```

6.3.4.2 Validation Checking

Validation of data is important before adding the data into the database. The validation codes were written in Javascript, so that validation on user input data is done before submission of data to the database. Listed below is the validation checking for new patient information.

```
If (Request.QueryString <> "") Then
 MM_editAction = MM_editAction & "?" & Request.QueryString
End If

' boolean to abort record edit
MM_abortEdit = false

' query string to execute
MM_editQuery = ""
%>
<%
' *** Insert Record: set variables

If (CStr(Request("MM_insert")) = "form1") Then

 MM_editConnection = MM_newpatient_STRING
 MM_editTable = "Patient"
 MM_editRedirectUrl = "successpatient.asp"
 MM_fieldsStr                                                    =
"textfield5|value|textfield4|value|textfield8|value|textfield9|value|textarea3|value|textfield
11|value|radiobutton|value|textfield10|value|blood|value|textfield12|value|textfield|value|t
extfield2|value|textarea|value|textfield3|value"
 MM_columnsStr                                                   =
"Dates|',none,"|Times|',none,"|Name|',none,"|IC|',none,"|Address|',none,"|Telephone|',none,'
'|Sex|',none,"|Occupation|',none,"|BloodGroup|',none,"|Allergies|',none,"|Name2|',none,"|IC
2|',none,"|Address2|',none,"|Relationship|',none,""

 ' create the MM_fields and MM_columns arrays
 MM_fields = Split(MM_fieldsStr, "|")
 MM_columns = Split(MM_columnsStr, "|")

 ' set the form values
```

```
For MM_i = LBound(MM_fields) To UBound(MM_fields) Step 2
 MM_fields(MM_i+1) = CStr(Request.Form(MM_fields(MM_i)))
Next

' append the query string to the redirect URL
If (MM_editRedirectUrl <> "" And Request.QueryString <> "") Then
 If (InStr(1, MM_editRedirectUrl, "?", vbTextCompare) = 0 And Request.QueryString
<> "") Then
  MM_editRedirectUrl = MM_editRedirectUrl & "?" & Request.QueryString
 Else
  MM_editRedirectUrl = MM_editRedirectUrl & "&" & Request.QueryString
 End If
End If

End If
%>
<%
' *** Insert Record: construct a sql insert statement and execute it

Dim MM_tableValues
Dim MM_dbValues

If (CStr(Request("MM_insert")) <> "") Then

 ' create the sql insert statement
 MM_tableValues = ""
 MM_dbValues = ""
 For MM_i = LBound(MM_fields) To UBound(MM_fields) Step 2
  MM_formVal = MM_fields(MM_i+1)
  MM_typeArray = Split(MM_columns(MM_i+1),",")
  MM_delim = MM_typeArray(0)
  If (MM_delim = "none") Then MM_delim = ""
  MM_altVal = MM_typeArray(1)
  If (MM_altVal = "none") Then MM_altVal = ""
  MM_emptyVal = MM_typeArray(2)
  If (MM_emptyVal = "none") Then MM_emptyVal = ""
  If (MM_formVal = "") Then
   MM_formVal = MM_emptyVal
  Else
   If (MM_altVal <> "") Then
    MM_formVal = MM_altVal
   ElseIf (MM_delim = "'") Then  ' escape quotes
    MM_formVal = "'" & Replace(MM_formVal,"'","''") & "'"
   Else
    MM_formVal = MM_delim + MM_formVal + MM_delim
   End If
```

```
  End If
  If (MM_i <> LBound(MM_fields)) Then
   MM_tableValues = MM_tableValues & ","
   MM_dbValues = MM_dbValues & ","
  End If
  MM_tableValues = MM_tableValues & MM_columns(MM_i)
  MM_dbValues = MM_dbValues & MM_formVal
 Next
 MM_editQuery = "insert into " & MM_editTable & " (" & MM_tableValues & ")
values (" & MM_dbValues & ")"

 If (Not MM_abortEdit) Then
  ' execute the insert
  Set MM_editCmd = Server.CreateObject("ADODB.Command")
  MM_editCmd.ActiveConnection = MM_editConnection
  MM_editCmd.CommandText = MM_editQuery
  MM_editCmd.Execute
  MM_editCmd.ActiveConnection.Close

  If (MM_editRedirectUrl <> "") Then
   Response.Redirect(MM_editRedirectUrl)
  End If
 End If

End If
```

6.3.4.3 Session Checking

This security feature is included in every password-protected page. If the user is not properly log, they cannot enter certain mode in the system.

```
' *** Validate request to log in to this site.
MM_LoginAction = Request.ServerVariables("URL")
If Request.QueryString<>"" Then MM_LoginAction = MM_LoginAction + "?" +
Request.QueryString
MM_valUsername=CStr(Request.Form("textfield"))
If MM_valUsername <> "" Then
 MM_fldUserAuthorization=""
 MM_redirectLoginSuccess="start1.asp"
```

```
 MM_redirectLoginFailed="fails.asp"
 MM_flag="ADODB.Recordset"
 set MM_rsUser = Server.CreateObject(MM_flag)
 MM_rsUser.ActiveConnection = MM_admin_STRING
 MM_rsUser.Source = "SELECT User, Password"
 If MM_fldUserAuthorization <> "" Then MM_rsUser.Source = MM_rsUser.Source &
"," & MM_fldUserAuthorization
 MM_rsUser.Source = MM_rsUser.Source & " FROM Admin WHERE User='" &
Replace(MM_valUsername,"'","''") &"' AND Password='" &
Replace(Request.Form("textfield2"),"'","''") & "'"
 MM_rsUser.CursorType = 0
 MM_rsUser.CursorLocation = 2
 MM_rsUser.LockType = 3
 MM_rsUser.Open
 If Not MM_rsUser.EOF Or Not MM_rsUser.BOF Then
  ' username and password match - this is a valid user
  Session("MM_Username") = MM_valUsername
  If (MM_fldUserAuthorization <> "") Then
   Session("MM_UserAuthorization") =
CStr(MM_rsUser.Fields.Item(MM_fldUserAuthorization).Value)
  Else
   Session("MM_UserAuthorization") = ""
  End If
  if CStr(Request.QueryString("accessdenied")) <> "" And false Then
   MM_redirectLoginSuccess = Request.QueryString("accessdenied")
  End If
  MM_rsUser.Close
  Response.Redirect(MM_redirectLoginSuccess)
 End If
 MM_rsUser.Close
 Response.Redirect(MM_redirectLoginFailed)
End If
%>
```

CHAPTER 7 - SYSTEM TESTING

CHAPTER 7 – SYSTEM TESTING

7.1 INTRODUCTION

Testing is a critical element in uncovering logical error and to test the system either it is reliability or not. The goal of the system testing is to design tests that will uncover the greatest number of errors with the minimum amount of time effort.

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

The primary goal of unit testing is to confirm that the unit is correctly coded and that it carries out the functions it suppose to carry out. This stage of testing verifies that the component function properly with types of input and output expected from studying the component's design. After each component 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 series of different tests designed to fully exercise the system to uncover its limitations and measure its capabilities. The objective is to test and integrated system and verifies that it meets specification requirements. System testing takes place at a higher level, the testing focuses on behavior rather than function or functional structure.

7.2 UNIT TESTING

For the Clinic Information System's unit level testing, the author applied the following three categories types of testing:

7.2.1 Ad Hoc Testing

Ad Hoc testing or ad lib testing means that the author simply play with the functioning unit, trying whatever comes to mind, in attempt to make it fail. One shortcoming of Ad Hoc testing is that while the author usually fined many errors, the author can never be sure what was or was not tested. Nevertheless Ad Hoc testing was fast and efficient way of debugging code errors during early development stage.

7.2.2 White Box Testing

White box testing basically involve looking at structure of the code. It focuses on the idea of coverage. The main objective would be to check the missing function.

The author performed branch coverage or node testing for those IF…ELSE…THEN statements where every branch/decision is tested at least once. And, compound condition coverage for multiple condition statements, and example of such would be the time checking statement in the Clinic Information System Schedule.

7.2.3 Black Box Testing

Black Box testing focuses on the functionality of the code. The main objective is to uncover those wrong functions programmed correctly, by feeding input to the black box and take notes on what output is produced.

During black box testing, the author use equivalence class partitioning. In equivalence class partitioning the author's runs one test for each class of input to the module and then run an additional test using invalid data to make sure the routines is working correctly. This test was done on the system user input forms.

The author also did a boundary value analysis on those user input forms, since many errors tend to occur on the boundaries of equivalence classes. The test includes test scenarios where the value sets is inside, on and outside boundaries.

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

1) opening and closing of connection to the database,
2) insertion of new data into database,
3) updating or modification of existing data in the database,
4) retrieving data from database (currently unable)
5) validation of user input before entering the whole system,
6) validation of user before entering certain mode (Admin Mode),
7) deleting data from the form,

8) updating schedule.

7.3 INTERGRATION TESTING

After the unit test done, the modules are integrated into a working system. For this system, incremental approach was used. In the incremental approach, the units are added one by one to the set of integrated units.

During integration testing, two or more units in which either unit that use output data from or provide input data for another unit were tested in collection. These units have related characteristics to perform a common goal or function such as delete function, displaying results.

Multiple values of test data were entered through the interface to ensure that the values along the interface are correct and that the specific calls in the calling modules are in the right sequence and of the right type and the values were inserted correctly into the database.

7.4 SYSTEM TESTING

Here is a list of system tests performed is as below:

- Error Message and Acknowledgement Message Testing

  Error message which can be generated by the system during invalid data entry are checked for spelling, appropriateness and consistence. The same test was also applied to the acknowledgement messages in the system. Acknowledgement message is the message that informs the user the state of a user requested process, for examples data submission, deletion of data, and others.

- Security testing

  This system is tested for improper penetration an unauthorized access, to ensure that the implementation of user login and the valid user checking procedures included in every authorized page are functioning accordingly. The test result shown that the security functions are working properly.

- Transaction Tracking

  A list of possible transaction, are tracked through the system to ascertain that they function correctly from "input" to "output". For example, each time a screen is reached which requires input or generates output, the appropriate

functions are processed and lead to subsystem for processing and then the right output is received.

This test was performed, and the functions behaved according to the requirements. The accuracy of data retrieval was high, effective navigation between screens and the speed of data retrieval was acceptable.

CHAPTER 8 - SYSTEM EVALUATION

CHAPTER 8 – SYSTEM EVALUATION

8.1 INTRODUCTION

At all phases of the system approach, evaluation is a process that occurs continuously, drawing on a variety of sources and information.

The role of this evaluation phase was to determine:

1. The extent to which the expected outcomes have been realized,
2. The prescriptive value of the process where extraneous factors were taken consideration.

8.2 SYSTEM STRENGTHS

- The system provides multiple access to the same pieces of data, for example in the case of view existing patient, the registration staff and the doctors can view it, but only doctors can make any changes for that data.

- Error messages reporting to user after user ignores following certain rules. This will also serve as a guide to users in case they might be accidentally did it.

- The system is designed for easy navigation, links are provided to help user to browses the pages. The left navigator bar always provides a link to the main page in case the user is "lost".

- If the users accidentally entered the wrong password, there will be error messages prompted. Error handling functions are essential so that the system will be able to roll back to its original function.

8.3 SYSTEM LIMITATIONS

- The patient can only try to make an appointment with the doctors through the doctor's email. There is no guarantee that the appointment can be fulfilled.

- There is no search engine for this system. The search engine is replaced with the view function that retrieves data from the database. The view function plays the main role in the system especially during the operation.

- The admin and the pharmacy module do not have a batch delete function for the patient and drugs data. They are required to delete the database one by one.

- Due to the certain problem (development software tool error), the system can not generate the data according to field that they are meant to fill.

- This system is browser dependent because VBScript is used to write the ASP codes, and VBScript is not supported by Netscape browses.

8.4 PROBLEMS AND SOLUTIONS

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

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 be completed within the given time frame. However, this was overcome by analyzing and studying all the capabilities that Visual InterDev and ASP technology can do before determining the scope of the system.

During the design phase, on of the major obstacles is to apply the theoretical information gather in the previous phase into practice use. The author found it difficult and hard to apply and produce the best solution of design in the early semester. Mainly, this was due to lack of experience and insufficient knowledge of designing a system. Reading through some of the related document and revising some of the textbook from previous subject especially on Software Engineering and System Analysis Design.

And during the implementation phase, because there is no prior experience in ASP scripting, there was a bit of learning curve in understanding how the ASP objects work. Scripting in a new environment such as ASP requires some knowledge of what the ASP objects do and how to use the objects to build the required functionality of the web application. These objects are the server components made available by PWS.

Technical problems that were not familiar have to be solved through various ways, reading developer's reference books, getting help from the supervisor, friends and through the internet.

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

8.5 FURTHER ENHANCMENTS

For the further enhancement the author plan to edit the search function to make sure the system is really dynamic and easy to use. The search page result will contain all the function that is needed by the users to done their task.

It would be an attractive feature for future enhancement, if the user allows the jump from the first page to the last page or to whatever page of their linking. This would definitely speed up web browsing and database maintenance as well.

The security level will be upgraded for the sake of the safety of the information about the entire patient of the clinics. The system will go to be more secure so that it can never easily been hacked by the hackers.

The system interface is going to be upgraded so the system looks more attractive and easy to use. Due to time constraint, at the moment, all of the maneuver menus are using the buttons.

8.6 CONCLUSION

On the whole of this project has achieved to deliver the system in specified time frame and fulfilled about 90% of the objectives and requirements as determined during system analysis phase.

During the literature survey phase, the information and data on the current Web technologies such as CGI, ASP concepts, client server architecture, and security issues enlighten the author on how Internet technologies work.

At the development phase of this project, there was a lot of valuable lessons learned and plenty of hand-on experience gained. Learning to program in HTML, ASP, and VBscript will be added valuable skills to the author further taking.

While programming skills are essential, good practice on software engineering techniques must also be applied efficiently. This project has provided an excellent opportunity to apply the theoretical knowledge obtained in subjects such as System Analysis and Design and Software engineering courses on system development life cycle (SLDC), database design, data modeling, design architectural, testing techniques and such into good use.

Apart from technical knowledge, the project also helps polished the author non-technical knowledge, such as communication skills, organizing skills, and problem solving skills.

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

APPENDICES

A - Sample Codes

B - User Manual

C - References

D - Glossary

E - Acronyms

APPENDIX A – SAMPLE CODES

Operation.asp

This function is used by the doctors during their operation time. The patient data are sorted by date and time.

```
<%@ Language=VBScript %>
<!-- #include file = "support.inc" -->
<%Response.Buffer=true%>

<html>
<head>
<style type="text/css">
td{
font-family: verdana;
font-style: normal;
font-size: .6em;
}
<!--a{text-decoration:none}-->
<!--a:hover{color:red;text-decoration:underline;}-->
</style>
<title>Sistem Pengurusan Akaun Famili 2003</title><meta http-equiv="Content-Type"
content="text/html; charset=iso-8859-1"></head>
<body bgcolor="#FFFFFF" background="../icon&background/PPBK094.JPG">
<table width="550" border="0" cellspacing="0" cellpadding="0" align="center">
 <tr>
  <td colspan="2"> <div align="center"><br>
    <br>
   </div></td>
 </tr>
 <tr bgcolor="#336699">
  <td colspan="2" bordercolor="#FFFFFF"
background="../icon&background/LROCK009.JPG" bgcolor="#999999">
   <div align="center"><b><font color="#000000" size="5"
face="Garamond">ADMIN
    OPERATION MODE</font></b></div></td>
 </tr>
 <tr>
  <td><div align="center">
```

```
      <p><br>
      </p>
      <p><strong><font size="2">23 September, 2003 </font></strong></p>
      <p> </p>
      <p><font color="#000000" size="2"><strong>Todays Patient(sort by
time):</strong></font>
      </p>
    </div></td>
  </tr>
  <tr>
    <td colspan="2"> <%
sql= "Select * From Patient"
set rs=Conn.Execute(sql)

Do While Not RS.eof
%> <br> <br> <table width="550" border=1 cellspacing=2 cellpadding=2 align=center
bgcolor=eff7ff>
      <tr bgcolor=b7ceec>
       <td align=center><strong>DATE</strong></td>
       <td align=center><strong>TIME</strong></td>
       <td align=center><strong>NAME</strong></td>
       <td align=center><strong>IC</strong></td>
       <td align=center><strong>ADDRESS</strong></td>
       <td align=center><strong>TELEPHONE </strong></td>
       <td align=center><strong>SEX</strong></td>
       <td align=center><strong>OCCUPATION</strong></td>
       <td align=center><strong>BLOOD GROUPS</strong></td>
       <td align=center><strong>ALLERGIES</strong></td>
       <td align=center><strong>NAME2</strong></td>
       <td align=center><strong>IC2</strong></td>
       <td align=center><strong>ADDRESS2</strong></td>
       <td align=center><strong>RELATIONSHIP</strong></td>
       <% do until rs.eof%>
     <tr>
       <TD align=top> <div align="center">
         <% = RS("Dates")%>
        </div></TD>
       <TD align=top> <div align="center">
         <% = RS("Times")%>
        </div></TD>
       <TD align=top> <div align="center">
         <% = RS("Name")%>
        </div></TD>
       <TD align=top> <div align="center">
         <% = RS("IC")%>
        </div></TD>
```

```
        <TD align=top> <div align="center">
          <% = RS("Address")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("Telephone")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("Sex")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("Occupation")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("BloodGroup")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("Allergies")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("Name2")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("IC2")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("Address2")%>
         </div></TD>
        <TD align=top> <div align="center">
          <% = RS("Relationship")%>
         </div></TD>
      </TR>
      <%
RS.movenext
loop
loop
%>
    </table>
    <p>
     <%
RS.close
Conn.close
%>
    </p>
    <p>  </p>
    <table width="100%" border="0">
     <tr>
```

```
        <td background="../icon&background/PPBK094.JPG" bgcolor="#FFFFFF"><div
align="center"><font color="#FFFFFF" size="2" face="Courier New, Courier,
mono"><strong>
            <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="22">
              <param name="BGCOLOR" value="">
              <param name="movie" value="button36.swf">
              <param name="quality" value="high">
              <param name="base" value=".">
              <embed src="button36.swf" base="."  quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" width="100" height="22"
></embed>
            </object>
            <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="22">
              <param name="BGCOLOR" value="">
              <param name="BASE" value=".">
              <param name="movie" value="treatmentp.swf">
              <param name="quality" value="high">
              <embed src="treatmentp.swf" width="100" height="22" quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" base="." ></embed>
            </object>
            <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="22">
              <param name="BGCOLOR" value="">
              <param name="movie" value="button37.swf">
              <param name="quality" value="high">
              <param name="base" value=".">
              <embed src="button37.swf" base="."  quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" width="100" height="22"
></embed>
            </object>
            </strong></font></div></td>
      </tr>
      <tr>
        <td background="../icon&background/PPBK094.JPG"
bgcolor="#FFFFFF"> </td>
      </tr>
      <tr>
```

```
      <td background="../icon&background/PPBK094.JPG"
bgcolor="#FFFFFF"> </td>
    </tr>
    <tr>
      <td bgcolor="#FFFFFF"> <div align="center">
        <table width="100%" border="0" align="center"
background="../icon&background/PPBK001.JPG">
          <tr>
            <td width="24%" height="39"><div align="center">
              <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
                <param name="BASE" value=".">
                <param name="BGCOLOR" value="">
                <param name="movie" value="button31.swf">
                <param name="quality" value="high">
                <embed src="button31.swf" width="100" height="23" quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" base="." ></embed>
              </object>
            </div></td>
            <td width="24%"><div align="center">
              <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
                <param name="BGCOLOR" value="">
                <param name="movie" value="formback.swf">
                <param name="quality" value="high">
                <param name="base" value=".">
                <embed src="formback.swf" base="."  quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" width="100" height="23"
></embed>
              </object>
            </div></td>
            <td width="24%"><div align="center">
              <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
                <param name="BGCOLOR" value="">
                <param name="BASE" value=".">
                <param name="movie" value="button32.swf">
                <param name="quality" value="high">
                <embed src="button32.swf" width="100" height="23" quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" base="." ></embed>
```

```
            </object>
          </div></td>
        <td width="28%"><div align="center">
          <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
            <param name="BGCOLOR" value="">
            <param name="BASE" value=".">
            <param name="movie" value="button33.swf">
            <param name="quality" value="high">
            <embed src="button33.swf" width="100" height="23" quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" base="." ></embed>
          </object>
        </div></td>
      </tr>
    </table>
    <font color="#FFFFFF" size="2" face="Courier New, Courier, mono"><strong>
    <reatment></strong></font></div></td>
  </tr>
 </table></td>
</tr>
</table>
<p> </p>
<p><br>
</p>
</body>
</html>
```

Login.asp

This function is used by the staffs of the clinic from the main page of the clinic's web site.

```
<%@LANGUAGE="VBSCRIPT" CODEPAGE="1252"%>
<!--#include file="../Connections/login.asp" -->
<%
Dim login
Dim login_numRows

Set login = Server.CreateObject("ADODB.Recordset")
login.ActiveConnection = MM_login_STRING
login.Source = "SELECT * FROM User"
login.CursorType = 0
login.CursorLocation = 2
login.LockType = 1
login.Open()

login_numRows = 0
%>
<%
' *** Validate request to log in to this site.
MM_LoginAction = Request.ServerVariables("URL")
If Request.QueryString<>"" Then MM_LoginAction = MM_LoginAction + "?" +
Request.QueryString
MM_valUsername=CStr(Request.Form("textfield"))
If MM_valUsername <> "" Then
 MM_fldUserAuthorization=""
 MM_redirectLoginSuccess="systemstart.asp"
 MM_redirectLoginFailed="fails1.asp"
 MM_flag="ADODB.Recordset"
 set MM_rsUser = Server.CreateObject(MM_flag)
 MM_rsUser.ActiveConnection = MM_login_STRING
 MM_rsUser.Source = "SELECT UserID, Password"
 If MM_fldUserAuthorization <> "" Then MM_rsUser.Source = MM_rsUser.Source &
"," & MM_fldUserAuthorization
 MM_rsUser.Source = MM_rsUser.Source & " FROM User WHERE UserID='" &
Replace(MM_valUsername,"'","''") &"' AND Password='" &
Replace(Request.Form("textfield2"),"'","''") & "'"
 MM_rsUser.CursorType = 0
 MM_rsUser.CursorLocation = 2
 MM_rsUser.LockType = 3
 MM_rsUser.Open
 If Not MM_rsUser.EOF Or Not MM_rsUser.BOF Then
```

```
    ' username and password match - this is a valid user
    Session("MM_Username") = MM_valUsername
    If (MM_fldUserAuthorization <> "") Then
      Session("MM_UserAuthorization")                                    =
CStr(MM_rsUser.Fields.Item(MM_fldUserAuthorization).Value)
    Else
      Session("MM_UserAuthorization") = ""
    End If
    if CStr(Request.QueryString("accessdenied")) <> "" And false Then
      MM_redirectLoginSuccess = Request.QueryString("accessdenied")
    End If
    MM_rsUser.Close
    Response.Redirect(MM_redirectLoginSuccess)
  End If
  MM_rsUser.Close
  Response.Redirect(MM_redirectLoginFailed)
End If
%>
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body background="../icon&background/grainblue.jpg">
<form name="form2" method="post" action="">
  <div                align="center"><font                color="#000066"                size="5"
face="Garamond"><strong>WELCOME
    TO PUSRAWI CLYSIS</strong></font> </div>
</form>
<p   align="center"><img   src="../icon&background/lines_blue_003.gif"   width="596"
height="7"></p>
<p align="center"><strong>TO ENTER THE SYSTEM PLEASE INSERT YOUR ID
AND PASSWORD</strong></p>
<table width="75%" border="4" align="center" bordercolor="#0000FF">
  <tr>
    <td                          background="../icon&background/LROCK029.JPG"><form
action="<%=MM_LoginAction%>" method="POST" name="form1" target="_self">
      <p align="center"><strong>LOGIN NAME</strong></p>
      <p align="center">
        <input type="text" name="textfield">
      </p>
      <p align="center"><strong>PASSWORD</strong></p>
      <p align="center">
        <input type="password" name="textfield2">
      </p>
```

```
    <table width="50%" border="0" align="center">
      <tr>
        <td> <div align="center">
            <input type="submit" target="_parent" name="Submit" value="Submit">
          </div></td>
        <td> <div align="center">
            <input type="reset" name="Submit2" value="Reset">
          </div></td>
      </tr>
    </table>
    <p align="center">  </p>
  </form></td>
 </tr>
</table>
<p  align="center"><img  src="../icon&background/lines_blue_003.gif"  width="596"
height="7"></p>
</body>
</html>
<%
login.Close()
Set login = Nothing
%>
```

Deletepatient.asp

For the ease of use this function, the doctor only have to enter the patient's IC number to delete the patient's record. The record that has been deleted cannot be restored back once it has been deleted.

```
<%@LANGUAGE="VBSCRIPT" CODEPAGE="1252"%>
<!--#include file="../Connections/deletepatient.asp" -->
<%
' *** Edit Operations: declare variables

Dim MM_editAction
Dim MM_abortEdit
Dim MM_editQuery
Dim MM_editCmd

Dim MM_editConnection
Dim MM_editTable
Dim MM_editRedirectUrl
Dim MM_editColumn
Dim MM_recordId

Dim MM_fieldsStr
Dim MM_columnsStr
Dim MM_fields
Dim MM_columns
Dim MM_typeArray
Dim MM_formVal
Dim MM_delim
Dim MM_altVal
Dim MM_emptyVal
Dim MM_i

MM_editAction = CStr(Request.ServerVariables("SCRIPT_NAME"))
If (Request.QueryString <> "") Then
  MM_editAction = MM_editAction & "?" & Request.QueryString
End If

' boolean to abort record edit
MM_abortEdit = false

' query string to execute
MM_editQuery = ""
```

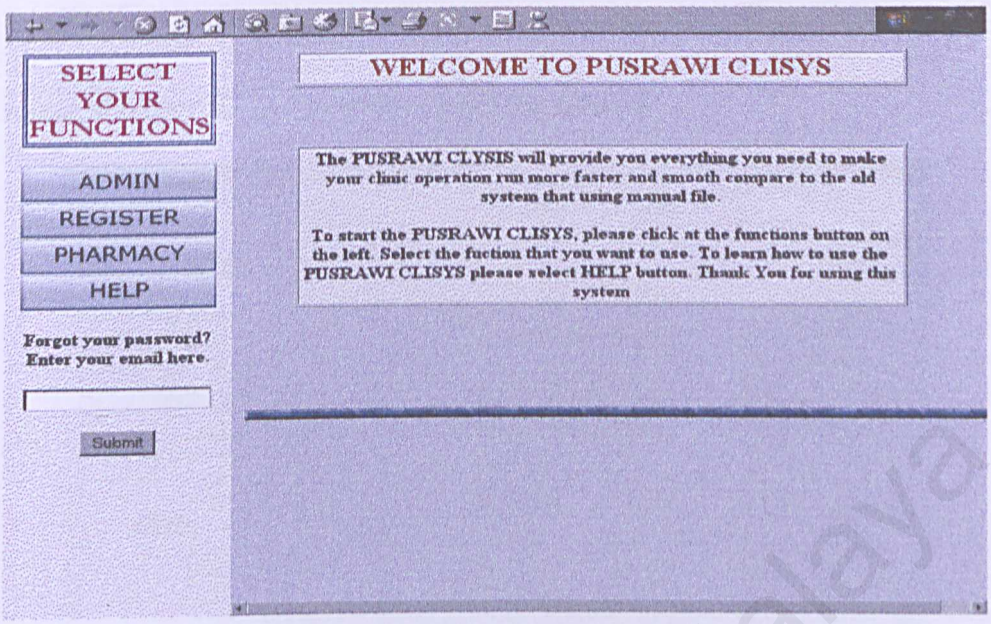
```
%>
<%
' *** Delete Record: declare variables

if (CStr(Request("MM_delete")) = "form1" And CStr(Request("MM_recordId")) <> "")
Then

  MM_editConnection = MM_deletepatient_STRING
  MM_editTable = "Patient"
  MM_editColumn = "IC"
  MM_recordId = "'" + Request.Form("MM_recordId") + "'"
  MM_editRedirectUrl = "deletesuccess.asp"

  ' append the query string to the redirect URL
  If (MM_editRedirectUrl <> "" And Request.QueryString <> "") Then
    If (InStr(1, MM_editRedirectUrl, "?", vbTextCompare) = 0 And Request.QueryString
<> "") Then
      MM_editRedirectUrl = MM_editRedirectUrl & "?" & Request.QueryString
    Else
      MM_editRedirectUrl = MM_editRedirectUrl & "&" & Request.QueryString
    End If
  End If

End If
%>
<%
' *** Delete Record: construct a sql delete statement and execute it

If (CStr(Request("MM_delete")) <> "" And CStr(Request("MM_recordId")) <> "") Then

  ' create the sql delete statement
  MM_editQuery = "delete from " & MM_editTable & " where " & MM_editColumn & "
= " & MM_recordId

  If (Not MM_abortEdit) Then
    ' execute the delete
    Set MM_editCmd = Server.CreateObject("ADODB.Command")
    MM_editCmd.ActiveConnection = MM_editConnection
    MM_editCmd.CommandText = MM_editQuery
    MM_editCmd.Execute
    MM_editCmd.ActiveConnection.Close

    If (MM_editRedirectUrl <> "") Then
      Response.Redirect(MM_editRedirectUrl)
    End If
  End If
```

```
End If
%>
<%
Dim deletepatient
Dim deletepatient_numRows

Set deletepatient = Server.CreateObject("ADODB.Recordset")
deletepatient.ActiveConnection = MM_deletepatient_STRING
deletepatient.Source = "SELECT * FROM Patient"
deletepatient.CursorType = 0
deletepatient.CursorLocation = 2
deletepatient.LockType = 1
deletepatient.Open()

deletepatient_numRows = 0
%>
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
</head>

<body background="../icon&background/PPBK094.JPG">
<form ACTION="<%=MM_editAction%>" METHOD="POST" name="form1">
 <p align="center"><strong>DELETE PATIENT</strong></p>
 <p align="center"> </p>
 <p align="center">Please enter the patient's IC that you want to delete.</p>
 <p align="center"><strong>(Note!!. You cannot undo this process)</strong></p>
 <p align="center"> </p>
 <table width="75%" border="0" align="center">
  <tr>
   <td width="52%"><div align="center"><strong>IC Number:</strong></div></td>
   <td width="48%"> <div align="left">
     <input type="text" name="textfield9">
    </div></td>
  </tr>
 </table>
 <p align="center">  </p>
 <table width="75%" border="0" align="center">
  <tr>
   <td><div align="center">
     <input type="submit" name="Submit" value="Delete">
    </div></td>
   <td><div align="center">
     <input type="button" name="Submit3" value="Save">
```

```
      </div></td>
     <td><div align="center">
       <input type="reset" name="Submit2" value="Cancel">
      </div></td>
    </tr>
  </table>
  <p align="center"> </p>
  <input type="hidden" name="MM_delete" value="form1">
  <input type="hidden" name="MM_recordId" value="<%=
deletepatient.Fields.Item("IC").Value %>">
  <table width="100%" border="0">
   <tr>
    <td bgcolor="#FFFFFF"> <div align="center">
      <table width="100%" border="0" align="center"
background="../icon&background/PPBK001.JPG">
       <tr>
        <td width="24%" height="39"><div align="center">
          <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
            <param name="BASE" value=".">
            <param name="BGCOLOR" value="">
            <param name="movie" value="button31.swf">
            <param name="quality" value="high">
            <embed src="button31.swf" width="100" height="23" quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" base="." ></embed>
          </object>
         </div></td>
        <td width="24%"><div align="center">
          <object classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
            <param name="BGCOLOR" value="">
            <param name="movie" value="formback.swf">
            <param name="quality" value="high">
            <param name="base" value=".">
            <embed src="formback.swf" base="." quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" width="100" height="23"
></embed>
          </object>
         </div></td>
        <td width="24%"><div align="center">
```

```
            <object            classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
              <param name="BGCOLOR" value="">
              <param name="BASE" value=".">
              <param name="movie" value="button32.swf">
              <param name="quality" value="high">
              <embed  src="button32.swf"  width="100"  height="23"  quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" base="." ></embed>
            </object>
           </div></td>
          <td width="28%"><div align="center">
            <object            classid="clsid:D27CDB6E-AE6D-11cf-96B8-444553540000"
codebase="http://download.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#ver
sion=5,0,0,0" width="100" height="23">
              <param name="BGCOLOR" value="">
              <param name="BASE" value=".">
              <param name="movie" value="button33.swf">
              <param name="quality" value="high">
              <embed  src="button33.swf"  width="100"  height="23"  quality="high"
pluginspage="http://www.macromedia.com/shockwave/download/index.cgi?P1_Prod_Ve
rsion=ShockwaveFlash" type="application/x-shockwave-flash" base="." ></embed>
            </object>
           </div></td>
        </tr>
      </table>
      <font color="#FFFFFF" size="2" face="Courier New, Courier, mono"><strong>
      <reatment></strong></font></div></td>
   </tr>
 </table>
</form>
</body>
</html>
<%
deletepatient.Close()
Set deletepatient = Nothing
%>
```

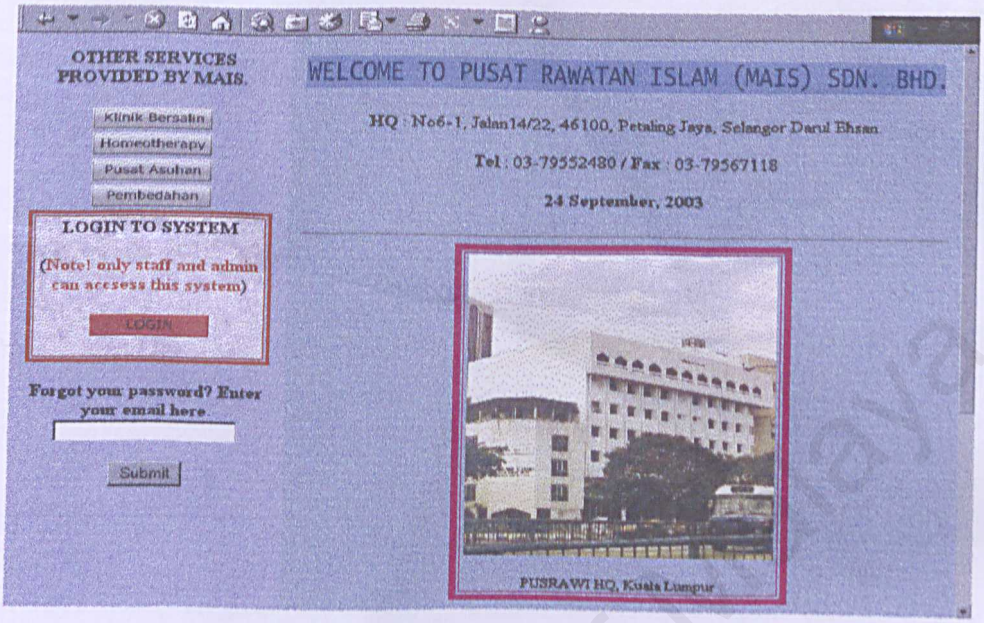
System Main Screen



1. For the administrator (doctors) click at the Admin button to start the operation.
2. For the registration counter, click at the register button.
3. For the pharmacy counter, click at the Pharmacy button.
4. To learn about the Clinic Information System, the user can click at the Help button. A simple instruction will guide the user how to use the system easily.
5. If the user forgot his/her login ID and password (for the doctors), they can acquired it by sending her/his email at the small column on the left side of the page.

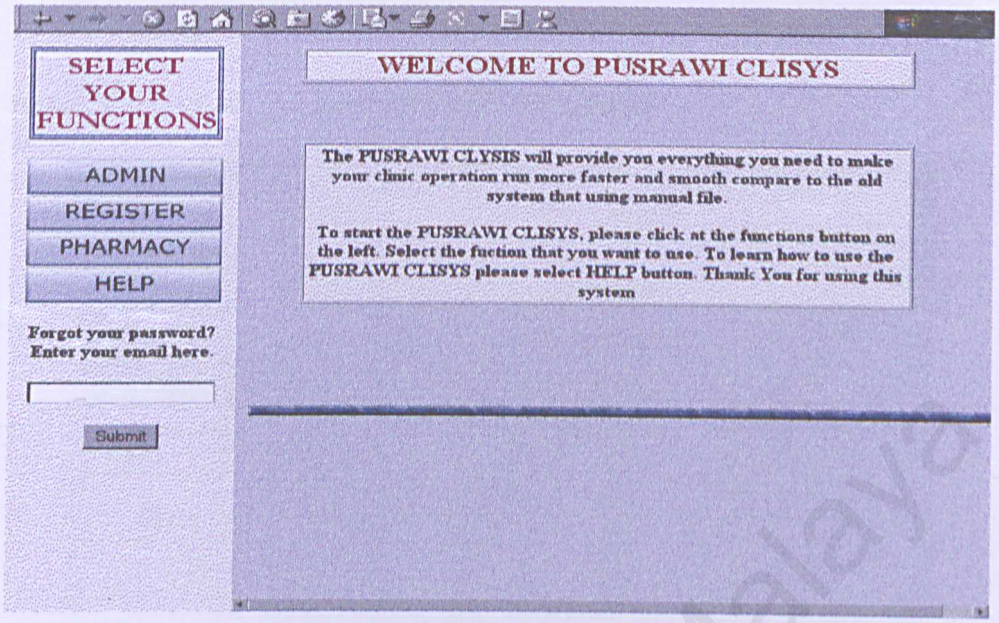
APPENDIX B – USER MANUAL

Entering Clinic Information System



1. For the visitor, they can view the latest information about the clinic by clicking at the About Us button.
2. The patient can view the doctor's schedule by clicking the Schedule button. At the Schedule page, they also can make with by clicking at the doctor's email.
3. To enter the Clinic Information System, click at the login button on the left side of the page. The login screen will appear after clicking the button
4. Enter the login ID and password to proceed to the system.

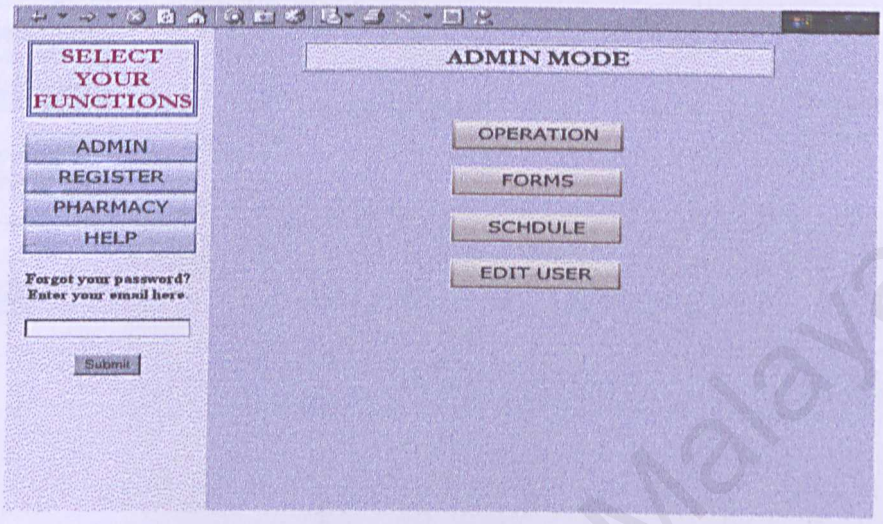
System Main Screen



1. For the administrator (doctors) click at the Admin button to start the operation.
2. For the registration counter, click at the register button.
3. For the pharmacy counter, click at the Pharmacy button.
4. To learn about the Clinic Information System, the user can click at the Help button. A simple instruction will guide the user how to use the system easily.
5. If the user forgot his/her login ID and password (for the doctors), they can acquired it by sending her/his email at the small column on the left side of the page.

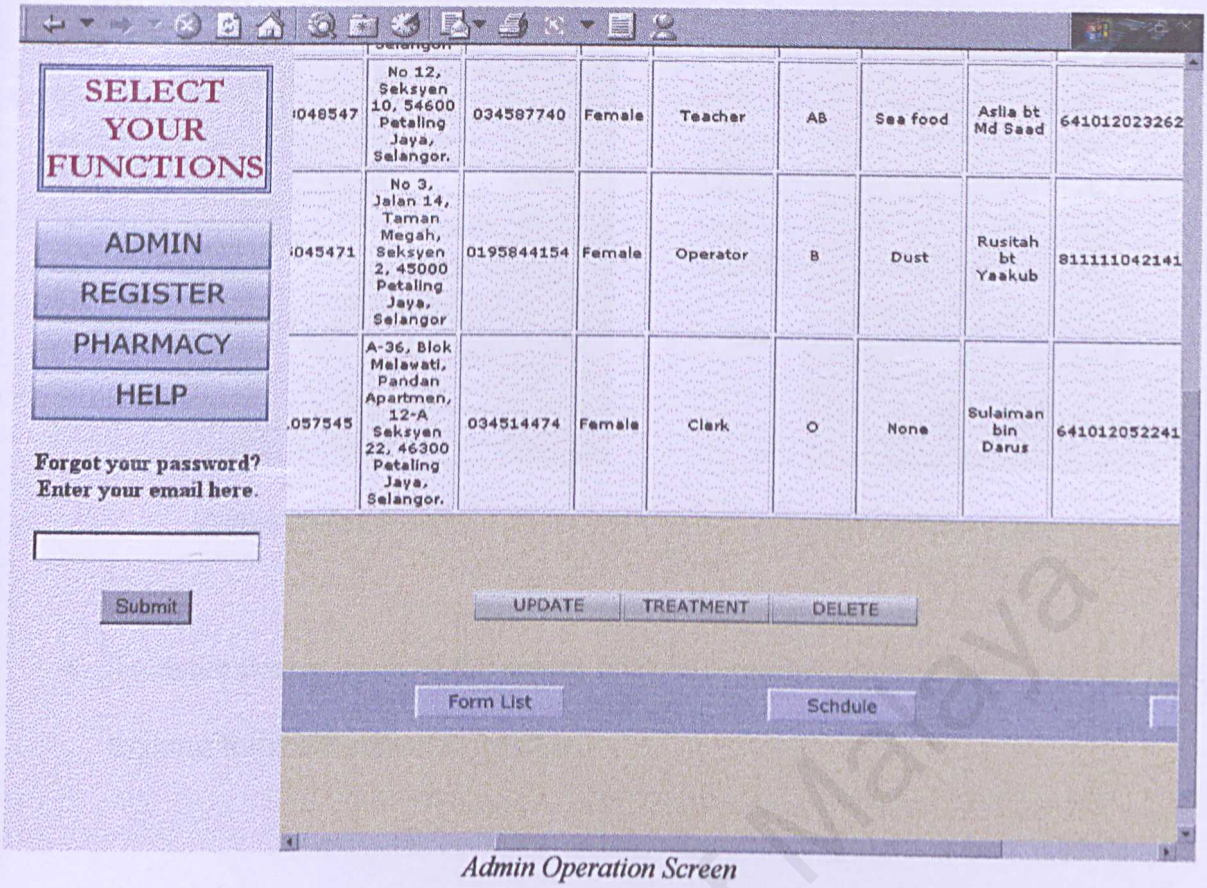
Admin Mode

1. After clicking at the Admin button, a login screen will appear. To enter the admin mode, the user ID and password must be entered.



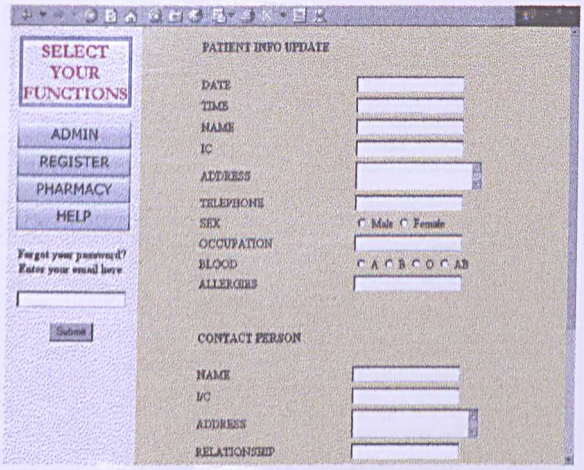
Admin Mode Screen

2. Click at the Operation button to start the operation.

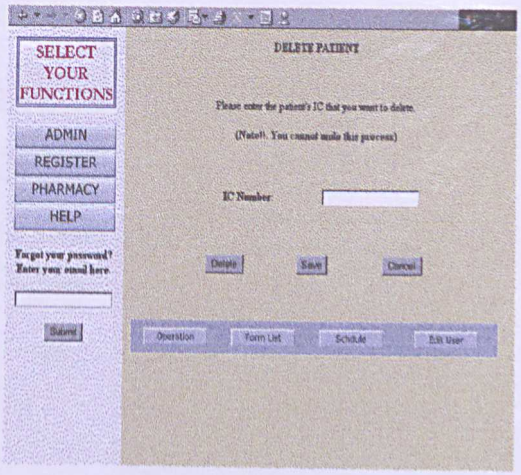


Admin Operation Screen

3. From the operation screen they can view the entire patient that been registered. The patient list will be sorted by date and time. From here the doctors can continue to the treatment process by clicking at the treatment button. The doctors can also update and delete the patient data from here.



Patient Update Screen

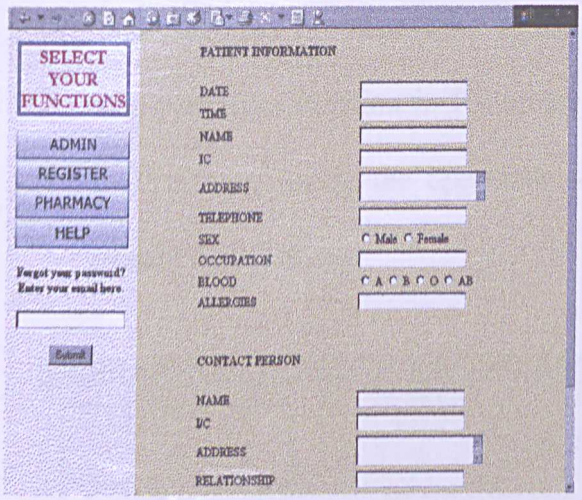


Patient Delete Screen

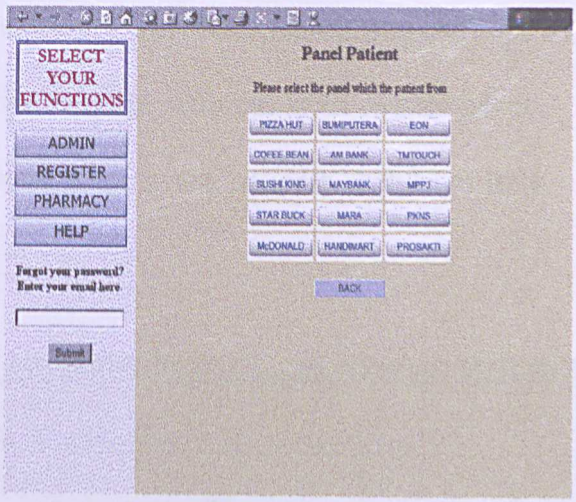
4. To make it easy, each of the screen contain a navigation bar so it is easy for user to jump from one screen to another.

Registration Mode

1. Simply click at the Register button. The Register Mode menu will pop up at the main page.
2. The Register Mode contains 3 functions that are New Patient, Existing Patient, and Panel Patient.
3. To insert new patient data, click at the New Patient button. The new patient form will pop-up after you clicked it.



New patient form



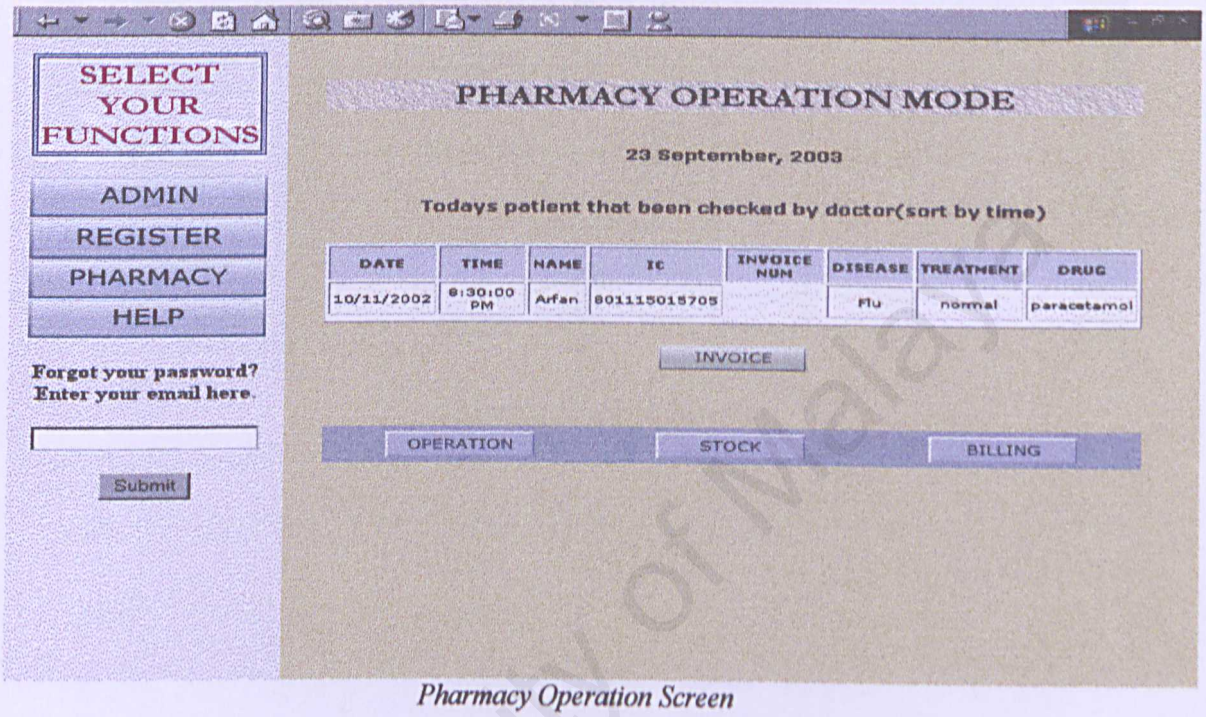
Panel patient list

4. For the patient come from the panel company, you can click at Panel Patient button. The list of the panel company will be displayed at the main page. Select the company name where the patient comes from.
5. To view the existing patient, please click at Existing Patient button. Note!! The registration counter can only view the patient data only. They cannot make any changes in the patient data. Only doctors and Admin have that authority to do that.

Admin Mode

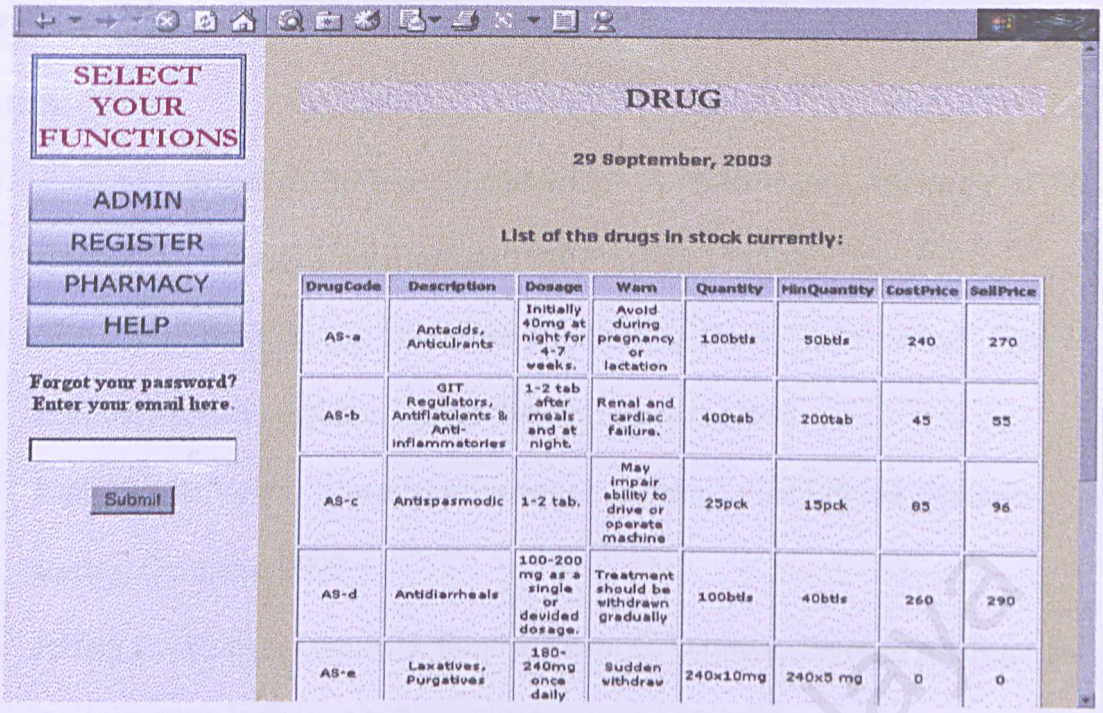
1. To enter the Pharmacy Mode, click Pharmacy button ant the Function Menu. Like Register mode, you don't have to enter the login name and password to enter this mode.

2. The Pharmacy Mode contains 3 functions that are Operation, Stock, and Billing.
3. The operation function in Pharmacy Mode is quite different from Admin Mode. In Pharmacy Mode the operation windows display the patients that have been treated by the doctor.



Pharmacy Operation Screen

4. The stock function is divided into two, New Stock and Existing Stock. The New Stock function is to enter the new stock into the database, while to check the existing stock; you can click at Existing Stock button.



DrugCode	Description	Dosage	Warn	Quantity	MinQuantity	CostPrice	SellPrice
AS-a	Antacids, Anticulrants	Initially 40mg at night for 4-7 weeks.	Avoid during pregnancy or lactation	100btls	50btls	240	270
AS-b	GIT Regulators, Antiflatulents & Anti-inflammatories	1-2 tab after meals and at night.	Renal and cardiac failure.	400tab	200tab	45	55
AS-c	Antispasmodic	1-2 tab.	May impair ability to drive or operate machine	25pck	15pck	85	96
AS-d	Antidiarrheals	100-200 mg as a single or devided dosage.	Treatment should be withdrawn gradually	100btls	40btls	260	290
AS-e	Laxatives, Purgatives	180-240mg once daily	Sudden withdraw	240x10mg	240x5 mg	0	0

Existing Stock Screen

APPENDIX C: REFERENCE

BOOKS

1. Pressman, Roger, S "Software Engineering – A Practitioner's Approach", Fifth Edition, McGraw-Hill International, 2001
2. Shari Lawrence Pfleeger, "Software Engineering Theory and Practice", First Edition, Prentice-Hall Inc, United State of America, 1998
3. Rudi Van de Velde, "Hospital Information System", First Edition, Springer-Verlag, Germany, 1992.
4. Jeffery L. Whitten, Lonnie D. Bently, Kevin C. Dittman, "System Analysis and Design", Fifth Edition, McGraw-Hill International, 2002.
5. Buser, D., Kauffman, J., Ullman, C., Francis, B., Suusman, D., " Beginning Active Server Page 3.0", First edition, Wrox Press Ltd, 1999.
6. Sellapan, P, " Software Engineering", First Edition, 2000
7. Salemi, Joe, "Guide To Client/Server Database", Second Edition, Ziff-Davis Press, California.

WEB SITE URL

1. http://www.mgh.org/
2. http://www.microsoft.com
3. http://www.asp101.com
4. http://www.pcmag.com
5. http://asp.superexpert.com
6. http://www.pmedic.com
7. http://pusrawi.com.my
8. http://www.planetsourcecode.com

APPENDIX D: GLOSSARY

ASP

ASP (Active Server Page) is a HTML page that includes scripts both for client-side and server-side that are processed (only server-side scripts) on the web server before they are sent to the user. It is somewhat similar to a Common Gateway Interface (CGI) application.

ADO

ActiveX Data Objects. A Microsoft data access technology that is the preferred way to provide data access capabilities to any kind of data store, such as relational databases, message stores, etc.

Client Browser

The Client Browser also known as *web browser*. It does the requesting connection to the web server. For example, user of web browser surfing a web site effectively making requests for web pages from the web server over the WWW.

Client/Sever Architecture

A design model for applications running over a network in which the task of the backend processing such as performing a physical search of a database, takes place in the server computer. The front-end processing, which involves communicating with user, is handled by programs distributed in the client workstation.

Database

A collection of related information stored in a structured and organized way. Using this structured collection, standard methods of retrieval like using Structured Query Language (SQL) can be used to define and manipulate the data.

DSN

Abbreviation for Data Source Name. It stores the necessary information for an application to make connection to a database.

Graphical User Interface (GUI)

An interface between user and the computer. GUIs, generally, requires a mouse – a type of pointing device. All GUI-based programs usually look similar, with pull down menus, scroll bars, etc.

Home Page

The first page of a web site, also called as main page.

HTML

Abbreviation for Hypertext Markup Language, on type of language used to create conventional Web pages.

HTTP

Abbreviation for Hypertext Transfer protocol, the native communication scheme of the World Wide Web (WWW), initially used to transfer hypertext documents.

Internet

A worldwide system of linked computer networks for data communication services such as World Wide Web and electronic mail. The Internet is a way of connecting existing computer networks that greatly extends the reach of each participating system.

Java

A platform-independent development language that allows system developers to create applications that runs in Internet or Intranet environment. Applications written in Java can be executed on any platform such as Macintosh, Windows or UNIX.

Open Database Connectivity (ODBC)

A standard protocol for database servers. If a database has an ODBC driver which is used to connect to ODBC –compliant data sources.

Remote Data Service

A high performance web based technology by Microsoft that brigs database connectivity and corporate data publishing capabilities in faster and easier manner.

TCP/IP

Abbreviation for Transmission Control/Internet Protocol. It is a standard set of communications protocols for computers to communicate with each other across the Internet and Intranet.

Web Server

A computer that provides web services and we pages to the Internet and Intranet users. This is where the server side business objects are typically deployed here.

World Wide Web (WWW)

A popular hypertext-based system of transmitting textual and multimedia based information through the Internet.

APPENDIX E: ACRONYMS

ADO	ActiveX Data Objects
ASP	Active Server Page
CDO	Collaboration Data Objects
CGI	Common Gateway Interface
DBMS	Database Management System
DFD	Data Flow Diagram
DLL	Dynamic Link Library
DSN	Data Source Name
ERD	Entity Relationship Diagram
GIF	Graphic Interchange Format
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IE	Internet Explorer
IS	Information System
ISAPI	Internet Server Application Programming Service
JPEG	Joint Photographic Experts Group
MS	Microsoft
ODBC	Open Database Connectivity
PWS	Personal Web Server
SQL	Structured Query Language
URL	Uniform Resource Locator

VBscript	Visual Basic Scripting Edition
WWW	World Wide Web