Penang Medical Centre System (PMCS)

MALA a/p ARUMUGAM

WET000089

2002/2003

Supervisor: **Assoc. Prof. Dr Diljit Singh**

Moderator: **Pn Norizan Mohd Yasin**

                      En. Mustaffa Kamal

Abstract

Penang Medical Centre System (PMCS) is a web-based registration system aimed to help hospital/clinic to achieve a balance between providing the best possible care for patient and minimizing risk to organize all the procedures at the same time. Users have the possibility to access information from the database from any location at any time as long as they connect to the Internet. The target users of this PMCS system are normally the Physician, Nurse/Staff and Pharmacist. PMCS mainly focuses on the Registration, which is part of the Health Information System.

PMCS is divided into four modules: the Registration module, the Scheduling module, the Patient History module and Admission and Discharge module. A combination of waterfall model and prototyping approach was selected for the development process because it supports the rapid application development and reduces the risk involved. Besides that, Active Server Pages are also selected to develop this application on the Windows XP Professional; moreover they integrated well with Microsoft Access 2000 for database application.

Extensive research and background surveys on all topics involved have been conducted to determine the feasibility of the system and what is required from this proposal. This survey is done to ensure a higher level of accuracy in the designing process, along with a better knowledge base to begin with when the actual system development begins. All these activities are aimed at achieving the best outcome; which is to create a comprehensive Health Information System to cater the needs and requirement of the users.

Acknowledgement

First and foremost I would like to extend my greatest gratitude and thanks to my supervisors, Profesor Madya Dr. Diljit Singh, for his invaluable assistance, idea, advice and utmost guidance throughout all the planning stages of this project. I wish to express my special thanks to Encik Mustafa Kamal, for his valuable comments and ideas on improving my proposed system.

I would also like to thank my family especially my mum, Madam Saraswathy and my friends: Sivashanker and Meera Jaganathan for their untiring help and support. And not to forget to my senior Mr. Selvakumar, who were very helpful in assisting me throughout this project.

Finally, I would like to share my gratitude to all the people who had kindly lead me with a helping hand whenever the occasion call for it and for their useful advice and expertise during this project.

Thank you very much….

Mala Arumugam.

Table of Contents

List of Figure & Tables

List of Figure

List of Tables

1.1 Background to Project

As medicine today contains many specialties, communication requirements are growing in experiential related to the rate of sub-specialization in health care. One of the greatest challenges that hospitals and clinics faces today is how to achieve a balance system between providing the best possible care for patients, managing resource efficiently and also minimizing risk to organize. Through the introduction of the Penang Medical Centre System (PMCS), patient's registration, patient details, appointment procedure and other important function that can be easily managed on an Information Technology System.

Accurate and complete patient information is essential for scheduling appointments, providing medical care, and running reports. This information also needs to be available to all the users. The registration information system will be designed to be the standard interface of entering patient registration data. Once you've entered the patient's registration information, it's available throughout the network whether you're scheduling an appointment, sending a letter, reviewing medical records, or running a report.

ENTITY Registration system will give you the ability to gather and store complete, accurate registration data which is the foundation of an effective patient management system. With complete Appointment Scheduling, Patient History and Admission, you finally have a comprehensive system which includes the automation and integration you need especially to remain competitive in today's changing health care environment. Therefore, the PMCS system creates a systematic flow of information for the user especially in managing patient records, it also creates customer/patient satisfaction and increases the standards of health management.

1.1 Background to Project

As medicine today contains many specialties, communication requirements are growing in experiential related to the rate of sub-specialization in health care. One of the greatest challenges that hospitals and clinics faces today is how to achieve a balance system between providing the best possible care for patients, managing resource efficiently and also minimizing risk to organize. Through the introduction of the Penang Medical Centre System (PMCS), patient's registration, patient details, appointment procedure and other important function that can be easily managed on an Information Technology System.

Accurate and complete patient information is essential for scheduling appointments, providing medical care, and running reports. This information also needs to be available at multiple sites. That's why the Registration system will be designed to be the standard component of the ENTITY system. Once you've entered the patient's registration information, it's available throughout the network whether you're scheduling an appointment, sending a letter, reviewing medical records, or running a report.

ENTITY Registration system will give you the ability to gather and store complete, accurate registration data which is the foundation of an effective patient management system. With complete Appointment Scheduling, Patient History and Admission, you finally have a comprehensive system which includes the automation and integration you need especially to remain competitive in today's changing health care environment. Therefore, the PMCS system creates a systematic flow of information for the user especially in managing patient records; it also creates customer/patient satisfaction and increases the standards of health management.

1.2 Project Overview

The proposed system which is basically the Penang Medical Center System (PMCS) is actually a web-based information system with a centralized database, which is designed based on the client server architecture. The primary concern of PMCS is to develop a web-based information system in a hospital/clinic merely for information sharing. This information is neither administrative nor medical. This system can give access to all of the demographics, patient's progress, physicians and medical center records.

Considering the Penang Medical Center System (PMCS) and what it's going to develop are basically systems that mainly focus on Patient Management Systems. The areas (sub-system) encompassed can be divided into few different modules, which are *Registration, Scheduling, Patient History and Admission/Discharge*. Each module consists of their areas as follows:

- ❑ **Registration**

  Inpatient & Outpatient

- ❑ **Scheduling**

  Doctor Schedule & Outpatient Appointment

- ❑ **Patient History**

  Prescription, Treatment, Laboratory and Personal Information

- ❑ **Admission/Discharge**

  Ward Management, Inpatient Management, Admission/Discharge

1.3 Objectives of Project

The goal of PMCS is more to the usage of computers and communication equipment to collect, to store, to process, to retrieve, and mainly to communicate patient care and administrative information for all hospital/clinic-affiliated activities and to satisfy the functional requirement of all authorized users. Besides that, the project's objectives is to develop and to produce an appropriate, smart, robust, flexible, scalable and efficient registration system for Penang Medical Center System mainly for hospitals and clinics in Malaysia. The other objectives include are:

- ❧ To speed the admission processing
- ❧ To help patient by reducing their waiting time-during registration
- ❧ To avoid time overlapping for different patients-during appointment
- ❧ To view patient records which are stored in one place – patient's medical History
- ❧ To develop a database system to organize and store patient records and data related to the system.

The long term objectives of this project are as mention below:

- ❧ To minimize repeat work processing
- ❧ To reduce paper management
- ❧ To have the accessibility of information from any location
- ❧ To have user friendly system

4

1.4 Scope of Project

The Penang Medical Center System will be developed according to the following parameters:

    ❦ Remove multiple recording of patient registration data through a central registration database.

    ❦ The attainment of the system is limited with the usage of the users name and password and therefore assuring only a number of user to access.

    ❦ The PMCS would comprise four modules, which is: *Registration, Scheduling, Patient History and Admission/Discharge.*

```
                        ┌─────────────┐
                        │    PMCS     │
                        └──────┬──────┘
         ┌───────────────┬─────┴─────────┬───────────────┐
  ┌─────────────┐ ┌─────────────┐ ┌─────────────┐ ┌─────────────┐
  │ REGISTRATION│ │ SCHEDULING  │ │   PATIENT   │ │  ADMISSION/ │
  │             │ │             │ │   HISTORY   │ │  DISCHARGE  │
  └─────────────┘ └─────────────┘ └─────────────┘ └─────────────┘
```
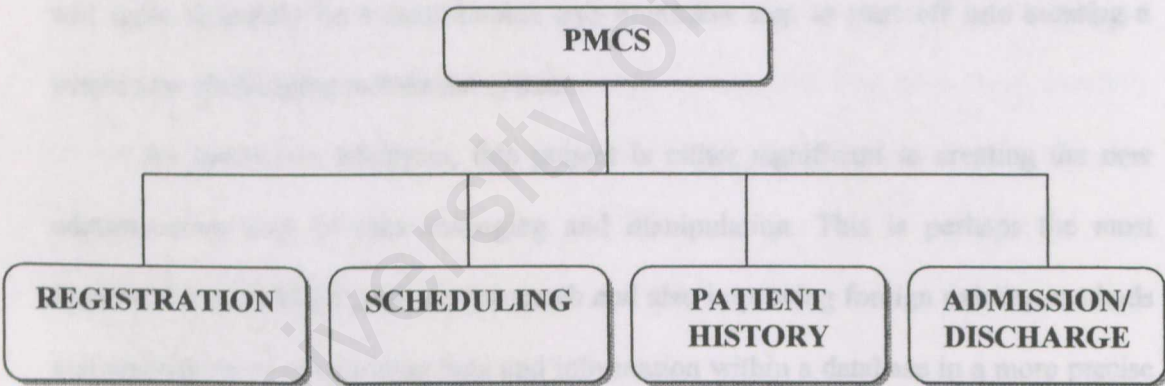
Figure 1.1: Project Scope

1.5 Importance of Project

In medical development, we are continuously moving towards an electronic environment in which user will have the possibility to access in a transparent and integrated way of all medical data including numeric data, imagine data and text data. Significance of the project is the transformation of patient records into a completely electronic form. Besides that, through this gigantic project, not only administrators and patients have an overview of all the medical data, but they will also be able to reduce their time during registration process.

The quality and outcome of the public users can be improved through the web-based system by ensuring the accurate, relevant, structured and timely information which is made accessible to the appropriate personal at all levels. Therefore, the PMCS will quite definitely be a considerable and important step to start off into creating a whole new challenging web-based system.

As pioneer in Malaysia, this project is rather significant in creating the new administrative way of data managing and manipulation. This is perhaps the most important time to begin with new research and also improving foreign existing methods and algorithms of maintaining data and information within a database in a more precise and meaningful manner. This will not only help to boost but maybe totally changing the way of Malaysian medical center registration process from its previous strategy into a better and a more competitive one.

This is definitely going to be a gigantic outbreak in the Malaysian Medical History with the hope of some level of telemedicine that is taking place widely although room for improvement still presents.

1.6 Definitions

Health is a state of complete physical, mental and social well-being, and not merely the absence of disease or infirmity. This is a positively oriented definition because it centers on well-being. It is also a broad definition because it includes mental and social well-being and does not define health only in term of diseases or infirmities.

What is managed *healthcare*? Any simple answer would miss the very real forces at work in the health care industry today that have led to evolution of management health care and would likewise miss the substantial positive contribution managed health care has made to the American health care system as we struggle to balance the three key elements which are access, cost, and quality. Managed healthcare, or managed care, is an approach to manage both the quality and the cost of medical care.

System is a collection of element or components that are organized for a common purpose or a system that comprises hardware components that have been carefully chosen so that they work well together and not forgetting software components or programs that run in the computer.

1.7 Project Schedule

A project schedule that consists of the whole development's activities is essential as it acted as a time management and control to the developer. A carefully planned out project will achieve a systematic progress and ensure on-time delivery of the product. The draft of the proposed schedule is drawn using a Gantt Chart for a good look of the timeline proposed. Below Table 1.1 shows a systematic timeline of the proposed Penang Medical Centre System.

Key Activities	2002							2003	
	Jun	July	Aug	Sep	Oct	Nov	Dec	Jan	Feb
Research & Reading	▓	▓							
Literature Review		▓	▓						
System Analysis									
Proposal Finalization			▓	▓					
System Design					▓				
Module Coding						▓			
Module Testing						▓			
System Integration & Testing							▓		
Report Documentation				▓	▓	▓	▓	▓	▓

Table 1.1: Project Schedule

1.8 Project Report Organization

This report will be prepared and arranged according to the chapters as below:

- **Chapter 1- Introduction**
  - ❖ Brief descriptions on project, the ideas, objective, scope, schedule and report organization

- **Chapter 2- Literature Review**
  - ❖ Research on existing health information system clinic and hospital
  - ❖ The health information system development in Malaysia
  - ❖ The existing system and comparisons

- **Chapter 3- Methodology and System Analysis**
  - ❖ Model system development methodology
  - ❖ Functional and non-functional requirements of the system are identified
  - ❖ System requirements- hardware and software
  - ❖ Development tools

- **Chapter 4- System Design**
  - ❖ System Architecture- system flowchart
  - ❖ Database design-screen design
  - ❖ System logical design- DFD diagram

- **Chapter 5- System Implementation**
  - ❖ Processes involved in System Implementation
  - ❖ Coding approach and style

- **Chapter 6- System Testing**
  - ❖ Purposes of System Testing

❖ Testing techniques

❖ Testing analysis

✦ **Chapter 7- System Evaluation**

❖ Evaluation techniques

❖ Strengths and weakness of the system

❖ Future enhancement and expansions of the system

❖ Problem encountered and solution

✦ **Conclusion**

❖ Project's conclusion

1.9 Chapter Summary

PMCS is a web-based application that makes the life of nurse/staff, physician and the user easier. It is designed to overcome those shortcomings in the current health information system like problems of time-consuming, work redundancy, etc. With the realization of its constraints or limitations, PMCS is designed to achieve its objectives of providing an easy-to-use, reliable, and maintainable system for health information users. Its development process will be guided by the project planning as scheduled in project Gantt chart.

After a brief overall introduction on the system, the next chapter will bring more in-depth explanation of the system. Literature review is carried out whereby current system. Literature review is carried out whereby current systems are surveyed to better understand how it is implemented, together with comparisons between different development tools, database and others.

Chapter 2
Literature Review

2.1 Role of Literature Review

To develop a system, you need to gather a lot of information regarding the system itself. A lot of research and analysis have to be done before the development phase of the proposed system can be initiated. In search of useful, informative and firm requirements, the findings of all the information should cover broad areas of the various fields. It also must have a strong relevance in what it is being mainly discussed about. It is important enough to avoid any information repetition and eventually putting on redundancy.

The main objective is to acquire the essential knowledge in giving the correct techniques and methods in implementing and designing a stage. Therefore the finding of this information comes from various sources for instance books, internet and surveys that are well thought and comprehensive.

2.2 Approaches to Literature Review

Basically, each source will yield different information and facts and it depends on how the researches are being done. The approaches that are being used in this project are as follows:-

- ↓ Basically while doing this project I've done a lot of reading on various sources and initiatively summarizing the written materials.

- ↓ Rather than that I've also have seek advice and guidance from my supervisor and also having group discussions with my team members.

- ↓ Besides that I've also access to some resources for instance the internet for more up-to-date information.

- ↓ I've done some general researches on previous projects, thesis and reference books in the library and from the document room.

- ↓ Finally to support my project and to visible the facts that I've gathered I personally visited a clinic.

2.3 Findings

It is always been wise to create a good solution by looking around for existing ones. The objectives of this finding are mainly focused on the user interface, flexibility and the features provided. Below are the lists of some of the web site that were observed during this project and also include a short summary of the visit to one of the local private clinic which is listed as part of the findings.

- **Sokka Clinic**

  Sokka Clinic is located in Prai, Penang. The system that's been used here is a database system and it is considered a small system. This database system is mainly for physician diagnosis and used by the physicians to keep patient's records. This system does not include registration and billing. The registration process in this clinic is still in manual system which still uses card for each patient's medical record.

- **University Malaya Medical Center**

  This web site gives information about the hospital and vacancies that are available in the hospital, besides the basic information given. We can't view or had a look directly to the system because our appointment was after submit this proposal. No much can said about the system unless about their web. But the requirement will take as a guideline to develop this PMCS.

- **Malacca Hospitals**

  This web site provides user with information about the hospital, facilities, bed categories (wards class) and other information that a user should know about the hospital.

15

↓ Hospital Fatimah, Ipoh

This web side is consisting or provides information about the hospital to the Internet user. Beside the information it also provides 2 different services to patient and the user that is "Ask the Expert" and "Email Patient". Email patient provide service to the user to send the mail to their relative or friend who admitted. This message will printout or inform to the patient y the staff who incharge, this message will delivery within 24 hours. Beside that Ask the Expert it kind of questions and answers (Q&A) about the disease or any subject about health or need any further information.

↓ Cape Fear Valley Health System

This web site provides health care providers referred services, but only those health care providers that are taking in new patients can be found. The search criteria is limited to the usage of "by specialty" and "by last name" keywords. Furthermore, the layout for this site is too simple and plain.

↓ Baylor Health Care System

This web site is quite a good site and it provides almost all of the specialties that enable users to obtain the diseases information that they would like to know about.

BHCS provides powerful search engines that enable users to search a doctor by locations, health information by disease and condition, services by specialties and even job opportunities are also available. However, BHCS does not provide an online Health Care Management of Information System that allows physicians, medical administrators or patients to update with the relevant data through the net.

16

Providence Health System

This web site is a simple site with a simple layout. The drawbacks that can be found in this site are the registration forms at the online pre-registration. Filling out these forms online will take about 20 minutes and the form consists of six pages. The patient must submit all six pages successfully to complete the online pre-registration.

Figure 2.1: Example of the existing system (Providence Health System)

Figure 2.2: Example of the existing system (Medina Health Care System)
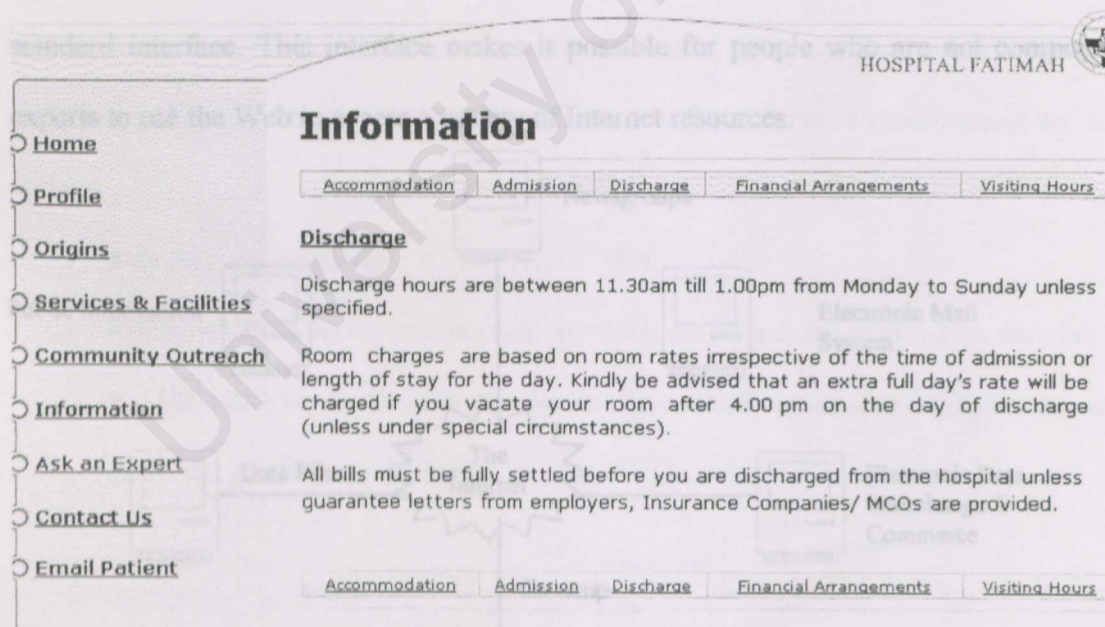
Figure 2.3: Example of the existing system (Hospital Fatimah)

2.3.1 Overview of Internet

The Internet is a large system of interconnected computer network that spans the globe. Using the Internet you can communicate with other people throughout the world by means of electronic mail; read online versions of newspapers, magazines, academic journals, and books; by joining the discussion group on almost any conceivable topic; participate in games and simulations; and obtain free computer software. Today, all kinds of businesses provide information about their products and services on the Internet itself. Many of these businesses use the Internet to market and sell their products and services. Part of the Internet is known as the World Wide Web, or, more simply, the Web is actually a subset of the computers on the Internet that are connected to each other in a specific way that makes those computers and their contents easily accessible to each other. The most important thing about the Web is that it includes an easy-to-use standard interface. This interface makes it possible for people who are not computer experts to use the Web to access a variety of Internet resources.
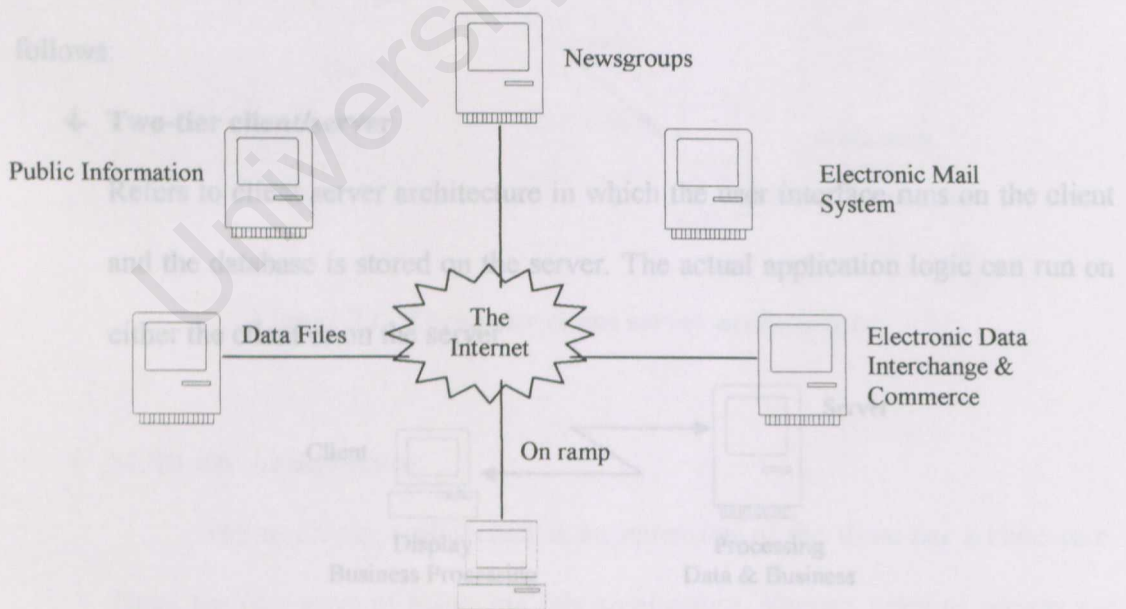
Figure 2.4: Concept of World Wide Web

19

2.3.2 Client/ Server Architecture Review

A client can be defined as a single-user workstation that provides presentation services and the appropriate computing, connectively, and database services and interface relevant to the business need.

Meanwhile a server is known as one or more multi-user processor with shared memory providing computing, connectively, and database services and interface which is actually relevant to the business needs.

Client server computing is an environment that satisfies the business need by appropriately allocating the application processes between the client and the server processors. The client requests services from the server; the server processes the request and returns the result to the client. The communication mechanism is a message passing inter-process communication (IPC) that enables to distribute the placement of the client and server processes. Client server is a software model of computing, not a hardware definitions. The most widely used architectures in today's network environment are as follows:

＋ **Two-tier client/server**

Refers to client server architecture in which the user interface runs on the client and the database is stored on the server. The actual application logic can run on either the client or on the server.
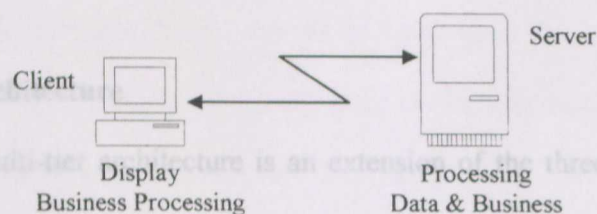
Figure 2.5: Two-tier client server architecture

Three-tier Architecture

A special type of client server architecture consisting of three well-defined and separate processes, each running on different platform: (1) The user interface; (2) The middle tier or often called the application server which is actually a process data; (3) A database management system (DBMS) that stores the data required by the application server. This tier runs on a second server called the database server.

The three-tier design has many advantages over traditional two-tier or single tier designs, due to the fact that the added modularity makes it easier to modify or replace one tier without affecting the other tiers and separating the application functions from the database functions makes it easier to implement load balancing.
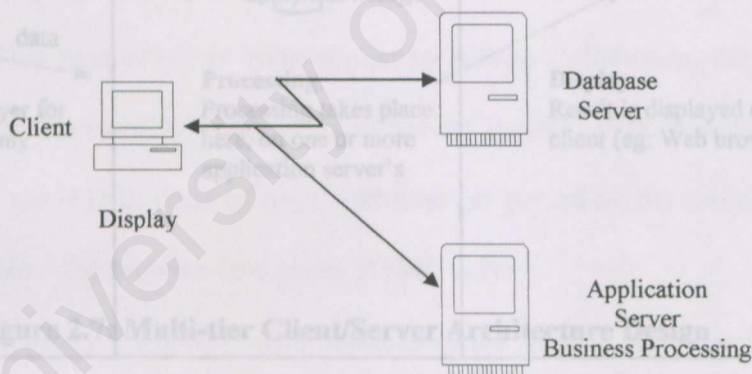
Figure 2.6: Three-tier client server architecture

Multi-tier Architecture

The multi-tier architecture is an extension of the three-tier architecture. There are two ways of achieving this architecture. Various types of servers are connected to the middle tier. These services are integrated and coupled, and to be

used only by the client. The other method would be partitioning the application logic among various hosts. Encapsulation of distributed functionality in such a manner provides significance advantages such as reusability and reliability. Application servers that contain the business rules and processing logic can access other application servers in order to supply services to the client application and servers.

Data Server Layer	Application Server Layer	Client Layer
	apply knowledge ←	← query
data server	network	network PC client
	apply knowledge	
data		
Storage: Data server layer for data storage only	Processing: Processing takes place here, on one or more application server's	Display: Result is displayed on thin client (eg: Web browser)

Figure 2.7: Multi-tier Client/Server Architecture Design

2.3.3 Overview of Scripting Language

Scripting language is an intermediate stage between HTML and programming languages such as Java, C++, and Visual Basic. The primary difference between scripting languages and programming languages is that the syntax and rules of scripting languages are less rigid and intricate than those of the programming languages. There are two types of scripting languages which are:

2.3.3.1 Server- Side Scripting

A server-side script is a script that is specially interpreted by the web server. There are several server-side technologies available in the market today. Just to name a few of them which are: PHP, CGI and ASP.

↓ **PHP**

PHP is a programming language being used on the server to create dynamic web pages. This technology is quite similar to Allaire ColdFusion, the Mod_Perl, Sun's JSP or Microsoft's ASP (Active Server Pages). PHP scripts are often embedded in the HTML code of page, and then get parsed on the server side. This would only allow the browser sees plains HTML only.

PHP is actually the most popular module for the Apache web server, according to an E-Soft survey. Anyway, studies have found that PHP is only in use on over 6% of all web domains in the world. That meant its market competitiveness is still very low compared to other web programming language. The knowledge of C programming language (with the complexity such as memory management, pointers, and strong typing taken out) is essential for the coding of PHP.

In the aspect of performance, a dynamic web page written using PHP could only execute in an intermediate speed as all the logic the application are integrated into the HTML. This makes web pages quite large in its size and takes some time to load and need optimizers to enhance the speed. PHP developed applications are portable in the condition of using PHP4.0 and above. The previous versions of PHP could only be integrated into the Apache web server or runs as a separated CGI Program.

♦ Common Gateway Interface (CGI)

CGI is a mechanism for creating scripting on the server, which can then be used to create dynamic web applications. CGI allows the user to invoke another program (such as a Perl script) on the web server to create the dynamic web page and the role of CGI is basically to pass the user-supplied data to the program for processing. CGI scripting programming is a method of creating a program that follows this standard of communications. A CGI script is just simply a program that communicates with the web document. Web documents are any kind of files used on the web. They could be HTML document, text files, images, or any other file formats. The existence of this gateway between the programs programmers have to write and the web document, allows them to create much more dynamic and interactive web pages than with the HTML alone. Figure 2.8 shows an example of client's request and retrieving of data.

24

HTTP Protocol   CGI

Web browser	Web server	CGI programs	Database
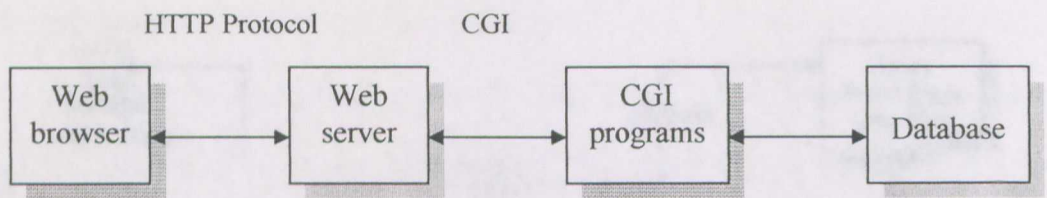
Figure 2.8 Example of the client's request and retrieving data using CGI

↓ **Active Server Pages**

Active Server Pages provides the efficiencies of application programming languages (API) access without the complexity and programming. In an ASP system, the code is usually written using a scripting language within the HTML page along with the tags themselves. When a web server locates an ASP page, the code is then executed on the server itself, generating in effect a pure HTML page. An often quoted security advantage is that the ASP code is stripped out before the page is sent to a browser preventing access to the source of proprietary application.

ASP is actually a feature of the Microsoft Internet Information Server (IIS), but the pages can be delivered to both Microsoft Internet explorer and Netscape Navigator. It is important to note that ASP is different from the client-side scripting.

With ASP scripting, the user can use any scripting language for which he/she provides the appropriate scripting engine. The majority of ASP pages are created using either VBScript or Jscript.
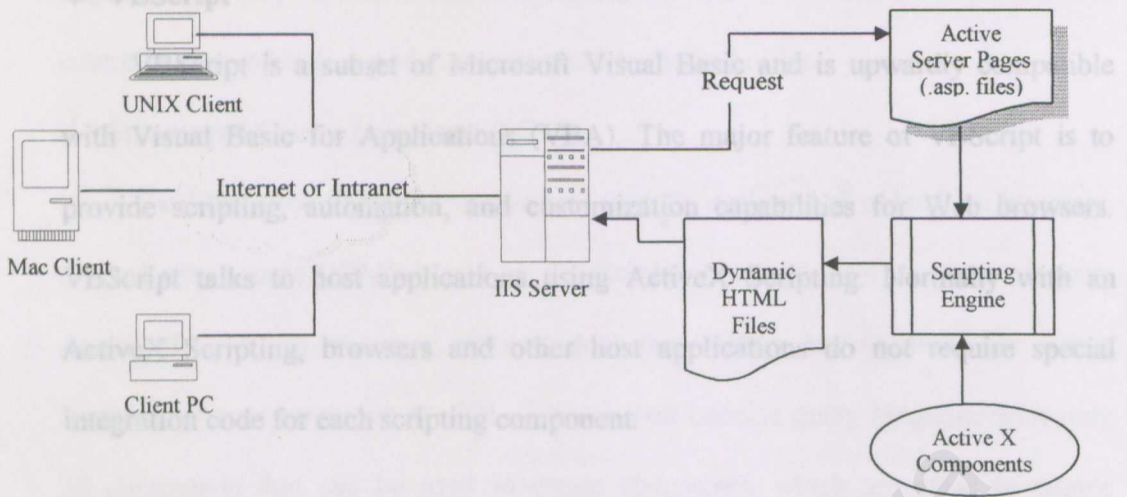
Figure 2.9: How Active Server Processes ASP Files

2.3.3.2 Client-Side Scripting

A client-side script is a script that is interpreted by the browser; Java, JavaScript and VBScript which are the most popular ones at the moment.

↓ Java

Java is a cross-platform for developing applications. Java offer better functionality and capabilities in areas such as graphic functions and file handling than scripting language.

↓ JavaScript

JavaScript is a scripting language mainly developed by Netscape. JavaScript is relatively simpler and less sophisticated than a real programming language, yet still powerful enough to be truly useful and it is also ideal for validating user input and it makes sure that the data entered by users is valid. JavaScript can easily manipulate various browser objects and it is used intelligently to conserve bandwidth.

26

- **VBScript**

VBScript is a subset of Microsoft Visual Basic and is upwardly compatible with Visual Basic for Applications (VBA). The major feature of VBScript is to provide scripting, automation, and customization capabilities for Web browsers. VBScript talks to host applications using ActiveX Scripting. Normally with an ActiveX Scripting, browsers and other host applications do not require special integration code for each scripting component.

2.3.4 Database Application Programming Language

- **Microsoft Access 2000**

Microsoft Access 2000 is a relational DBMS (Database Management System) which is designed for small office or home user to use for storing data's in relational format. With data access interface paradigm such as Remote Data Object (RDO) and Data Access Object (DAO), Access not only can be used as a database in a client/server but also as an n-tier architecture system. It provides intuitive and user-friendly interface to create a database easily, but it is quite slow in processing transaction to SQL 7.0.

- **Oracle**

Oracle supports not only structured alphanumeric data placed in traditional database tables and rows, but also unstructured text, images, audio and video. It is used to manage everything from personal information to corporate data centers and even to giant multimedia system that serve up information which is currently on demand. It runs on almost every popular computer, from the smallest laptop, to the

largest supercomputer, to the network computer that will bring interactive news, entertainment, education and commerce into home and office.

↓ **Structured Query Language (SQL)**

The Structured Query Language was initially designed as a database language to explicitly access DBMS based on the Relational model. SQL is more properly described as a sub-language, since it does not contain any facilities for screen handling or user input/output. SQL is an elegant concise query language with only 30 commands that can be used to create statements, which are close to simple English natural language statements.

2.3.5 Other Software Development Tools

↓ **Dreamweaver UltraDev 4**

Dreamweaver UltraDev 4 is software created to help the developer, programmer and designer to ceate and change the web application data in visual with multiple server platform. Among the server that support by amweaver UltraDev 4 are Microsoft ASP, Sun Microsystem JSP and Allaire ColdFusian Markup Language.

↓ **Microsoft FrontPage 2000**

Microsoft FrontPage 2000 is software that to develop web pages and management tools which we can use to desing anything in the web page. It is easy to because its function is same as other Microsoft Office Premium.

The differences between Macromedia Dreamweaver Ultradev 4 and Microsoft
FrontPage 2000 is shown in the Table 2.1.

Software	Advantage	Disadvantage
Dreamweaver UltraDev 4	i) Allow multiple server platform ii) Create database content straight to web pages iii) Developer feel easy to manage the data iv) Easy to integrate with other multimedia softwares	i) Need good understanding ii) This software need to buy separately iii) Expensive sotfware
Microsoft FrontPage 2000	i) Easy to understand ii) Can buy together with other Microsoft office software iii)Suitable for novice user	i) Limited platform ii) Cannot integrate with the database

**Table 2.1: The differences between Macromedia Dreamweaver Ultradev 4
and Microsoft FrontPage 2000**

2.4 Relationship to Proposed Project

The proposed Penang Medical Center (PMCS) system would incorporate several techniques used in the reviewed system. As the main objectives of the proposed system would be to provide user with a good registration system, a user-friendly web-based system, the system has to be simple and easy-to-use. The literature reviews has helped to decide on the most suitable tools and to determine the important elements of a Penang Medical Center System (PMCS).

Though there are many elements comprised on each of the system the PMCS would only include the necessary elements to keep it simple. The proposed system would be a free web-based system that contains similar elements akin to many of the reviewed systems. As a web-based system, the system would be developed on a client-server computing model. There are various client/server architectures but only a few suits the proposed system. The review on the client/server architecture provides a great deal of Information on client/server computing model and also helped to identify the most suitable architecture for the proposed system.

2.5 Chapter Summary

The overview of the existing system is to determine the advantages and the weaknesses of the system. Therefore, it can become the guideline to develop a more advanced new system by taking the advantages and avoiding the weaknesses. Advantages of the existing Computerized System are basically a lot and one of it is a user can access the web site anytime (24 -hours per day) and this method is used for the stand-alone system. The security in the Internet is not a problem, because there are wide ranges of security such as firewall and digital signatory existed in it. Besides that the users can also analyze patient's medical history because every patients report is up to date because it is systematically recorded in the database.

The weakness of the existing system contains too much information because the web pages are full of the text information; it is too complicated and difficult to understand. Rather than that the poor interface design which is full of text and subsequently makes the background color too bright and causes the user to feel inconvenienced and uncomfortable. Moreover a user also needs to fill in many fields and it takes a lot of their time. This PMCS system although does not have powerful features to some extent, still has some strength of its own when compared to some existing system (computerized system whether Ms DOS or stand alone).

Chapter 3
Methodology

3.1 Objectives of Project

The goal of PMCS is to use computers and communication equipments to collect, store, process, retrieve, and communicate patient care and administrative information for all hospital/clinic-affiliated activities and to satisfy the functional requirement of all authorized users. Besides that the project's objectives are developing and producing an appropriate, smart, robust, flexible, scalable and efficient system for Penang Medical Center System mainly for hospitals and clinics in Malaysia. The other objectives include:

- To speed the admission processing
- To help patient by reducing their waiting time-during registration
- To avoid time overlapping for different patients-during appointment
- To view patient records which are stored in one place
- To develop a database system to organize and store patient records and data related to the system.

The long term objectives of this project are as mention below:

- To minimize repeat work processing
- To reduce paper management
- To have the accessibility of information from any location
- To have user friendly system

3.2 Development Methodology

In this project, the system development methodology proposed is actually a combination of waterfall model and prototyping. The waterfall model is very useful especially in presenting a very high-level view of system development.

In this model, all stages are depicted as cascading from one to another implies that one development stage should be completed before the next begins. This model also includes the activities and sub-processes called prototype that enhance the understanding of system requirements. Prototyping helps to improve or to correct an error, thus the major kinks in the requirement, are addressed and fixed before the requirements are officially validated when a system testing is taking place. Validation ensures that the system has implemented the entire requirement, so each of the functions can be trace back to a particular requirement in the specification. System testing also verifies the requirement as the verification ensures that each function works correctly.

The principal stages of the model onto the fundamental development activities are:

1. *Requirements analysis and definition.* In this stage the requirements of the "to be developed" software are established. These are usually the services it will provide, its constraints and the goals of the software. Once these are established they have to be defined in such a way that they are usable in the next stage. This stage is often preceded by a feasibility study or a feasibility study is included in this stage.

2. *System and software design.* In this stage the established requirements, flowing from the first stage, are identified as software and hardware requirements. The software requirements are then transformed ... so that they can be readily transformed into computer programs.

3. *Implementation and unit testing.* This is the stage where the computer programs are created. Each program is called a unit, and unit testing is the verification that every unit meets its specification.

4. *System testing.* All the units are combined and now the whole is tested. When the combined programs are successfully tested the software product is finished.

5. *Operation and maintenance.* Most software products include this stage of the development. It involves correcting errors that have gone undetected before, improvement and other forms of support. This stage is part of the life cycle of a software product, and not of the strict development, although improvements and fixes can still be considered as "development".

Figure 3.1: Waterfall model with prototyping

3.3 Principal Stage of Waterfall Model

The principal stages of the model onto the fundamental development activities are:

1. *Requirements analysis and definition.* In this stage the requirements of the "to be developed software" are established. These are usually the services it will provide, its constraints and the goals of the software. Once these are established they have to be defined in such a way that they are usable in the next stage. This stage is often precluded by a feasibility study or a feasibility study is included in this stage.

2. *System and software design.* In this stage the established requirements, flowing from the first stage, are identified as software or hardware requirements. The software requirements are then translated in such a way that they can be readily transformed into computer programs.

3. *Implementation and unit testing.* This is the stage where the computer programs are created. Each program is called a unit, and unit testing is the verification that every unit meets its specification.

4. *System testing.* All the units are combined and now the whole is tested. When the combined programs are successfully tested the software product is finished.

5. *Operation and maintenance.* Most software products include this stage of the development. It involves correcting errors that have gone undetected before, improvement and other forms of support. This stage is part of the life cycle of a software product, and not of the strict development, although improvements and fixes can still be considered as "development".

3.4 Rationale for Proposed Methodology

This model is chosen because waterfall model can suggest to the developer the sequence of activities that they should expect to encounter. Besides that, the developer also can gauge how close the project is to completion and mainly to give point of time. The Waterfall Model also enables the developer to make necessary preparation for the next phase. This is because variety of techniques should be used to determine the user's need and Prototyping is used with waterfall model because it basically helps the developer to enhance their understanding about the system. The user requirement will be identified and documented in the prototyping section in the waterfall model. This information will be used to develop user interface and will be known as the prototype. In other words, prototyping also enables the user to interact with the system to have a better understanding about what the new system will actually be. All of the feedback from the user will be used to re-adapt the prototype in order to fulfill the user needs. The prototype is then used again and re-adapt until it satisfy the developers and users. The prototyping is added to waterfall model because the users do not know exactly what they want until they actually have a chance to see and work with the system or part of the system. Then, the system developers build system using feedbacks supplied by the users.

3.5 Requirement Analysis

A requirement is a feature of the system or a description of something that the system is capable of doing in order to fulfill the system's purpose [Pfleeger, 2001]. A requirement does not only describe the flow of into and from the system but also the constraints on the system's performance. Requirement elicitation us the critical part of the process. This is because variety of techniques should be used to determine the user's need and customer's want. Basically the requirements identify the (what) of the system where else the design identify the (how) of the system.

Requirements definition is a complete listing of everything the customer expects the proposed system to do. It represents an understanding concept between a customer and developer of what the customer actually needs or wants and it is usually written jointly with the developer. On the other hand, the requirement specification restates the requirement definition in technical terms appropriate for the development of a system design *Figure 3.2* shows the process of Determining Requirement.
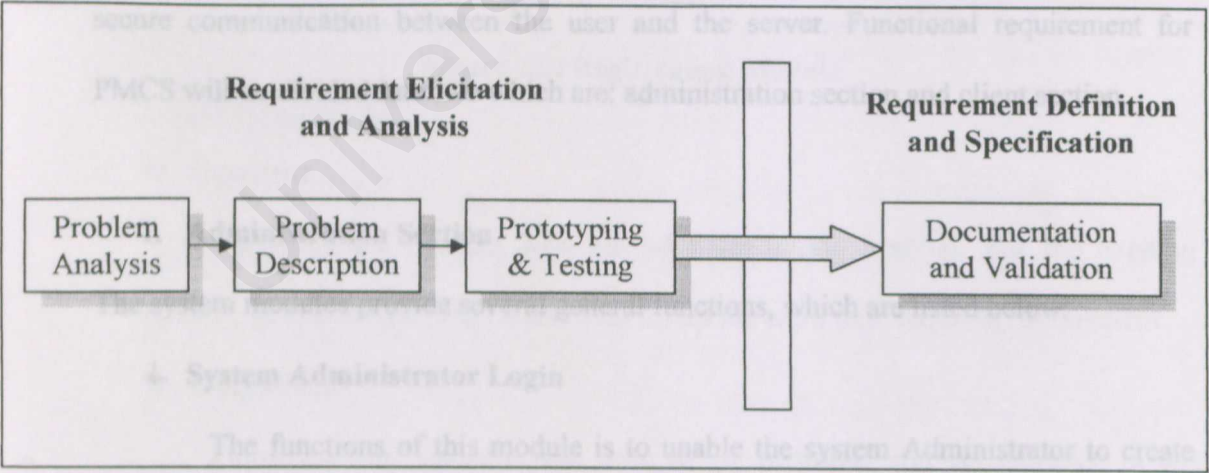
Figure 3.2: The process of Determining Requirements

3.5.1 Requirement Specifications

The requirements of PMCS can be divided into two categories. They are functional requirements and non-functional requirements. Functional requirement describes how PMCS applications interact with its operating environment. On the other hand, non-functional requirement are the limitations on the system that narrow the choices that can be made for implementation.

3.5.1.1 Functional Requirements

A functional requirement describes an interaction between the system and its environment. Since the requirement describes a system's behavior, hence the functional requirement also explains how the system should behave when given a contain stimuli. An authentication and authorization process is vital to this system to protect its database from the non-authorized user. In order to access patient's data in the administration section, a user identity and password are required. The process is necessary to establish a secure communication between the user and the server. Functional requirement for PMCS will be divided into two which are: administration section and client section.

1. Administration Section

The system modules provide several general functions, which are listed below:

✦ System Administrator Login

The functions of this module is to unable the system Administrator to create user groups and creates the administrator ID to access the system.

Add/ Delete User

An administration can add new user or delete a current user from the system. This function enables the administrator mainly to keep track of the current users of the system.

Change Password

An administrator is able to change the password if he/she wants to. To enhance security aspects, the password has to be confirmed.

Link

This module has several links that links to another interface. The following page lists down the functions and advantages of each modules and sub modules in the system.

Registration

Figure 3.3: Registration Module

○ **Inpatient**

Inpatients are patients who are admitted in the hospital. For the existing records, search for the records and then fill up the extra field required for inpatient.

○ **Outpatient**

Outpatients are patients who want to get treatment and not admitted in the hospital. The registration for outpatient will be accessed by the staffs and nurses who are on duty.

❖ **Advantages**

During an emergency case, staff/nurse or medical administrator can access the patient records immediately when a patient is in the emergency condition. Searching for the patient's record will be easier and no more repeating the work (manual system) for a same purpose.

Scheduling
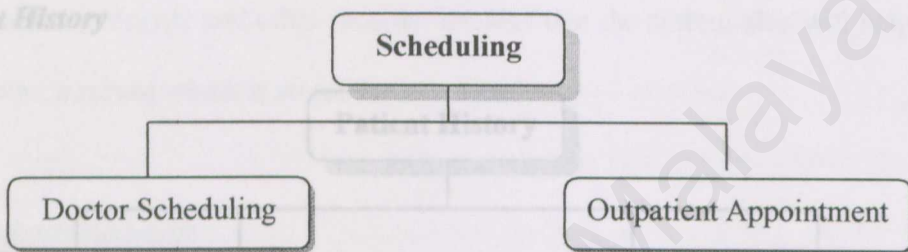
Figure 3.4: Scheduling Module

◗ **Doctor Schedule**

This interface will be access mainly by the doctors and it will give information's or a reminder to the physician on how many patients they are suppose to visit for each day.

Besides that it also provides patient schedule for the next appointment to the doctor, and it will prevent time overlapping with other patient's appointment time.

◗ **Outpatient Appointment**

This interface will be access by the nurse/staff, to view time and how many patients are scheduled for physician for each day.

❖ **Advantages**

This system will help to avoid time overlapping for different outpatient at the same time. When a Physician has to attend emergency operation, their patient will be arranged to other physicians, it will be done by nurse and they will fix the new time to the new physician. Besides that it is also will reduce the waiting time for outpatient before starting with their treatment.

Patient History
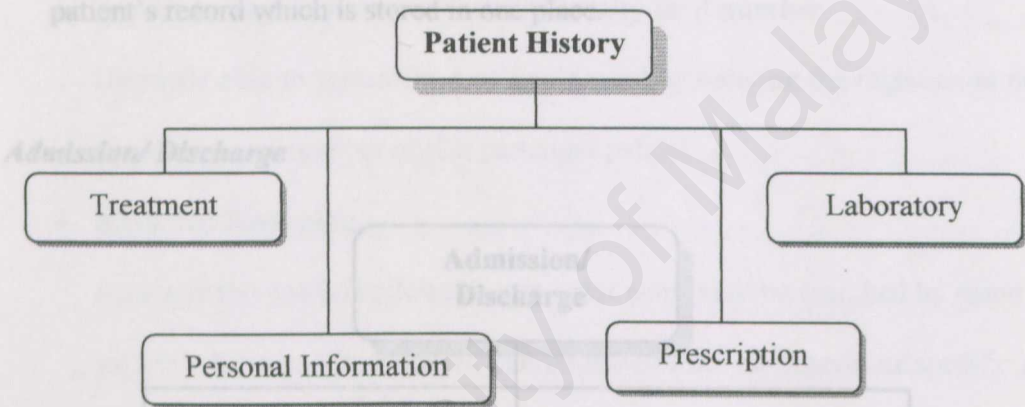
Figure 3.5: Patient History Module

◗ **Personal Information**

Keep track of all the patient's information records

◗ **Treatment**

The interface is actually what the Physician will record about the patient's treatment on that particular treatment day. This interface links to the laboratory by choosing the option button to view if the patient has done laboratory test to view the results.

❖ **Prescription**

This interface will be access by the pharmacist

❖ **Laboratory**

❖ This interface is for the lab assistant to key in the result of the test they have done

to the patient.

❖ **Advantages**

This system can minimize the time used in searching for the patient's records,

laboratory records and other records. Besides that the system also will help to view

patient's record which is stored in one place.

Admission/ Discharge
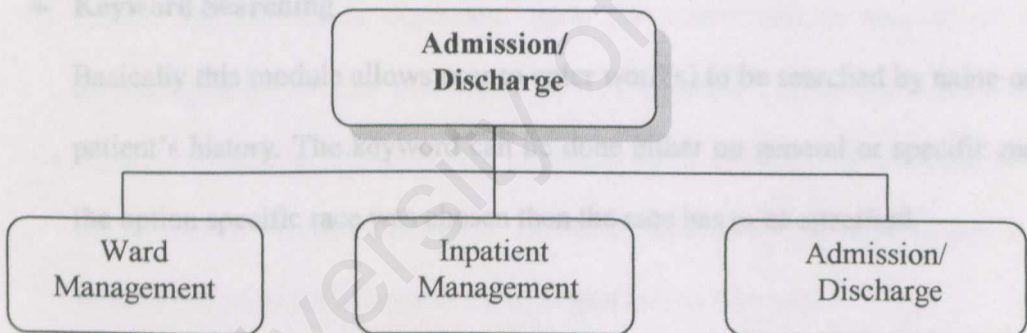
Figure 3.6: Admission/ Discharge Module

❖ **Ward Management**

This is the information about the ward and the person who is on duty.

❖ **Inpatient Management**

This interface consist of patient treatments, extra treatments such as test with

result, medicines that was given when the patient was admitted, and all about the

patient as it was link to the module Patient History.

- ○ **Admission/Discharge**

  This interface consist of patient admission, discharge process, patient location, schedule of discharge patient to assign vacate beds to the waiting patients.

- ❖ **Advantage**

  Inpatient managements are better managed compared to the previous time.

2. Client Section

The client section consists of 2 different main modules as shown below:

- ↓ **Search by registration number or identity card number**

  Users are able to search for a patient's ward by entering the registration number or identity card number of that particular patient.

- ↓ **Keyword Searching**

  Basically this module allows user to enter word(s) to be searched by name or the patient's history. The keyword can be done either on general or specific race if the option specific race was chosen then the race has to be specified.

The Functional requirement for integration of all these modules is that, all these modules are expected to provide their services in an integrated environment. The integration covers the flow of data among the systems used and front-end integration. The front-end integration includes the user interface integration. Integration should also allow modules to be added in future in order to increase the usefulness of the system.

3.5.1.2 Non-Functional Requirement

A non-functional requirement or constraint describes a restriction on the system that limits the choices for constructing a solution to the problems that arise. The constraint usually narrows the selection which is made at the design stage, after the requirements have been specified. The following are described as the keys non-functional requirement for the PMCS.

↓ **Portability**

Portability refers to the capability of the system to operate on various platforms regardless at manufacturer or operating system. The implementation of PMCS involves integration of many tools and basically uses component-based technologies provided primarily by Microsoft. Therefore, the portability of the system applications is dependent upon the restrictions in support of these technologies by various platforms. At this moment, due to the restrictions in support of these technologies, the implementation of PMCS is restricted only to the Microsoft Windows platform. However, this situation might change in the future as support for Microsoft technologies are on demand.

↓ **Maintability**

Maintainability is the ease with which a program can be corrected if an error is encountered, adopted if its environment changes, or enhanced if the customer desires a change in the requirements [Pressman 1997]. Effort should be taken to locate and fix an error that occurs in a program.

45

Reliability

Software reliability is the probability that a program will operate successfully. The application system which includes software and hardware shall be reliable and shall not cause unnecessary and unplanned down-time of the overall environment. Each time an error is defected, it should be successfully rectified in order to improve the reliability of the system.

Flexibility

Flexibility refers to the system's ability to adopt now technologies and resources as well as implementation in changing environments. As the projects implementation is based on client server architecture, it will be easy to modify the operational program.

Usability

Normally the application system shall be easy to use. They shall enhance support rather than limit or restrict business process. Human interfaces shall be intuitive and consistent within themselves for the sake of purpose use.

Scalability

Scalability refers to the ability of the system to migrate to the machine of greater or lesser power, depending upon the need, without much change to the underlying components. Due to the client, server application and changes enhance the system to web-based application the scalability issues can be addressed rather well by separating the key modules. Each module can run on separate machines that can expend or contracted as well as run from within a single machine. Database scalability issues can be resolved using distributed database architecture.

- **User Friendly**

    If a program is not "user friendly" it is often doomed to failure, even if the functions that it performs are valuable. Therefore, the user interfaces of the system should be easy to understand and user friendly.

- **Unity of Environment**

    The PMCS must appear to the user as a single working environment and not an amalgamation of distinctly separate units. As such there must be a standardized interface design as well as standardized method for function invocation by the user.

3.6 Feasibility Studies

An estimation is made of whether the identified user needs may be satisfied using the current software and hardware technologies. The study will actually decide if the proposed system will be cost-effective from the business point of view and if it can be developed given the existing budgetary constraints. A feasibility study should be relatively cheap and quick. The result should inform the decision of whether to go ahead with a more detailed analysis. [Sommerville, 2001]

The PMCS is feasible because there are many systems regarding the health in the market. Even in Malaysia there are a few systems such as in Selayang Hospital which has been developed. So, in my opinion this project can been done successfully provided by certain tools and techniques given. This system is not only efficient but also provides a good way of registration process especially to patients who are in need. The PMCS constraints a patient's time and most of all the system is user-friendly.

3.7 Determination of System Concept

Determination of system concept is done from the early stage of planning to develop this system. This is to avoid the development of system which is differ from its objective and the system purpose. This determination include the service implemented, facilities provided, system constrain, functional and non-functional requirement.

3.8 System Requirement

3.8.1 Software Tools

Information system today is developed using some form of software tools and these tools help to simplify the system development process by automating some of the processes besides reducing cost and time. While doing some surveys on development tools, the tools that have been chosen are most suitable tools for the system. The tools including the Operating System, Web Server, Database System, Web Application, Scripting Language and preferred Web Browser. The following are the tools used in the system after considering their advantage.

↓ **Operating System**

Microsoft Windows XP Professional

Windows XP Professional was chosen as the operating system of choice due to several advantages that are distinct when compare to other operating system. It is because Windows XP Professional has great capability in the behind-the-scenes running of the computer. Its performance is at an all-time high, allowing of using more programs and having them run faster than ever. Windows XP Professional is dependable and

stable, so we can always rely on the performance and effectiveness of the computer. Best of all, it's compatibility with other programs is better than ever. Although Linux also provides a user friendly GUI, it undoubtedly has much lower usage percentage in the market. This prompted us to choose the more popular Window XP Professional over Linux. UNIX was not chosen because it does not provide a user friendly GUI environment. Beside Window XP Professional support for innovative web published features, customizable tools and the new technology make Windows XP Professional is the best operating system available to publish information over the Internet especially in PMCS.

⬇ Database

Microsoft Access 2000

Microsoft Access 2000 is a relational database that was developed by Microsoft. It has currently 10 million users worldwide. The Access package is one of the best-selling relational database packages for Windows in the market. Together with the ODBC driver for Access, data retrieval can be made from database in client-server system architecture.

The reasons of choosing Access 2000 are:

- ❑ It can be easily maintained by the database administrator.
- ❑ It migrate data from spreadsheets and other databases, and is the easy way to find answer, share information over intranet and Internet.
- ❑ It allows generating, analyzing and creating reports fast. It integrates ease of use from the data entry point to printing in HTML.
- ❑ It upsized to Microsoft SQL Server 7.0 for more controls.

- ○ It can be integrated easily with Microsoft Front Page 2000.

- ○ Many simple and user-friendly features in building tables, quires, forms and reports that can be customized to suit project needs.

⊥ Web Application Language

Active Server Pages

ASP is chosen to be the toll for creating dynamic web pages. It is chosen because it is one of the most powerful, fastest and widely used in the industry. In fact because it is fast, we can cut down the processing time. This will be nice to user; they do not waste their time waiting for the downloading process. Because it is widely used, a lot of programmer may know it; this will be increase the ease of maintenance. Nowadays, a lot of end user deals with Microsoft Application; there will be a well combination between ASP and this software because it is a Microsoft technology. Besides, ASP can be executed in any computer, with any modern browsers.

⊥ Scripting Language

VBScript

Microsoft Visual Basic Scripting Edition or better knows as VBScript is a member of the Visual Basic family of programming languages. It brings active scripting to wide variety of environments, including Web client scripting in Microsoft internet Explorer and Web server scripting in Microsoft Personal Web Server.

VBScript does not include functionality that directly to the client machine's operating system or file system so it is safe for the World Wide Web.

51

JScript

It is a powerful scripting language targeted specifically at the Internet and it is a full implementation scripting language that consists of some enhancement that takes advantage of capabilities of Microsoft Internet Explorer. JScript is an interpreted object-based scripting language. JScript scripts can run only in the presence of an interpreter, either in a web server or a web browser. It is a loosely typed language that has little capability for reading or writing files and Jscript is also a case-sensitive, while VBScript is not.

↓ Web Application Development Tools

Microsoft FrontPage 2000

Microsoft FrontPage 2000 provides a comprehensive solution for publishing and managing workgroup Web site. FrontPage 2000 helps users to create the sites they want whether in creating a personal Web page or corporate Internet or Intranet site and it is easy to learn. Besides that it makes updating sites easy, and works well with Office. In addition, Office 2000 users can now save HTML documents directly to FrontPage-based sites, while features such as shared Office menus and toolbars make Office users feel immediately comfortable with FrontPage 2000 by describing new and improved features that allows teams to create and manages intranet sites.

Macromedia Dreamweaver Ultradev 4

Besides Microsoft FrontPage 2000, I also have been using Macromedia Dreamweaver Ultradev 4 in order to create ASP pages. It is easy to use because we can check the coding while we typing the coding. Other than that, Macromedia

Dreamweaver Ultradev 4 also is a user friendly web development tools. It also have interactive buttons to help to develop interactive web pages.

↓ **Web Browser**

In terms of Web Browser, I have chosen Internet Explorer (IE) because of some unique features that the other browser doesn't have. One of them is the IE which interacts well with both the Scripting language VBScript and JScript. Both JScript and VBSript are essentially plug-in scripting language-they interact with internet Explores underlying ActiveX scripting architecture to drive the Web browser. If we choose IE as our browser, then it will not assume that our- user will use IE as their browser. They have their authority to use others that suit to their application.

There are two ways to solve this problem:

◑ *Code for one browser only.*

This is not really a viable solution. By using the Navigator only the JavaScript will limit to our web page's audience.

◑ *Code only to the least common denominator.*

By limiting the JavaScript to that which works for both Navigator and Internet Explore, developers will have to access to most of the functionality they need. In other words, the developers has to test with both the tests will have to occur via the web server as well as run locally; the tests should be repeated several times to ensure that no adverse effect show up over time ; the code should be tested with all possible environments.

Others Tools

Adobe Photoshop 6, Macromedia Flash 5, Swish 2.0 and Background Magic are the tools chosen to develop more attractive web page.

3.8.2 Hardware Requirement

Server Hardware Requirement

- Server with Pentium 2.00GHz Processor.

- 2 GB *hard disk*

- Minimum 32 MB RAM

- NIC and networking connection with 10 Mbps bandwidth

3.9 Chapter Summary

Methodology and system analysis are the most critical process of information system development. By having a good methodology a plan can be drawn up to guide the development towards the project goals. However system analysis is used to determine and clearly to find out what a system does and to analyze the system needs either is functional requirement or non-functional requirement.

The reviewing of the methodology being used, procedures that specify the system requirements in detail, analysis of development technologies will help in gaining the advantages and knowledge about the implementation of the proposed system.

The following chapter will discuss about the design of the system, which includes the system architecture review, description of system flow, description of database and interface design and other system components.

Chapter 4
System Design

4.1 Introduction

"The system design phase is covered with synthesizing or putting all the parts together into a viable, workable system" [Sellapan, 1998]. In other words, system design is concerned with how the system's functionality is to be provided by the different components of the system. The design processes for PMCS are divided into architectural, functional, process, database and user interface.

4.2 Structure Chart

Structure chart is used to depict high level of obstruction of specification system. The use of structure chart is to describe the interaction between independent modules. Major function forms the initial component part of the structure chart, which can be broken in to detail sub component. PMCS is divided into four major components mainly Registration, Scheduling, Patient History, Admission / Discharge. Each of these four components is further divided in to many modules as shown in Figure 4.1.

Login Screen

Registration | Scheduling | Patient History | Ad/ Dis

Registration:
- Search
- Add
- Save

Scheduling:
- Outpatient App
  - View
  - Edit
  - Send
  - Save
- Doc. Schedule
  - Search
  - Delete
  - Save
  - View

Ad/ Dis:
- Search
- Edit/Add
- Delete
- Save
- View

Personal Info | Treatment | Laboratory | Prescription

Personal Info:
- Edit
- Search
- Save
- Delete

Treatment:
- Update
- Search

Laboratory:
- Search
- Update
- Save
- Add

Prescription:
- View
- Add

Figure 4.1 Structure Chart PMCS

58

4.3 Architectural Design

An architectural design is the initial design process of identifying the subsystems and establishing a framework for subsystem control and communication. Large system can be decomposed into subsystems that provide some related set of services [Sommerville, 2001]. Architectural design usually requires the system structuring and modular decomposition activities.
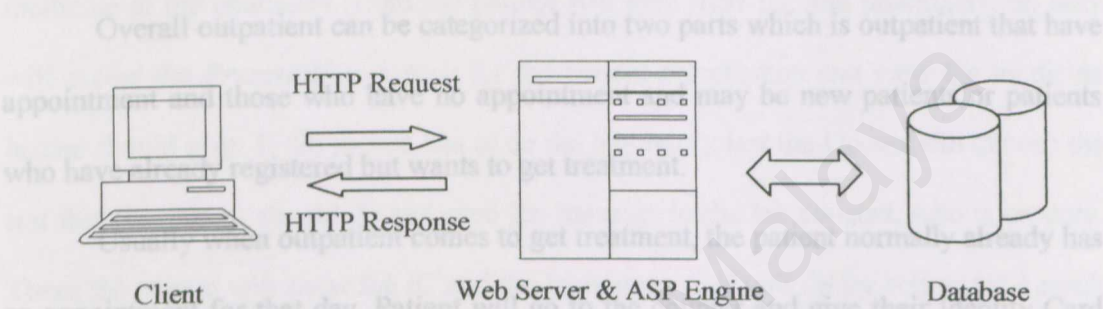
Figure 4.2: System Architecture for PMCS

The system is designed in such a way like 3-tier client server architecture, where in local network every system user will be the client to make request to the server. After receiving requests, the server will response and processes the client's request and presents an active or dynamic content page by retrieving data from backend database. For development stage, the server combines three major functions which are the web sever, Active Server Pages and the Access Database.

4.4 System Functional Design

Designs for this part are same like in 3.5.1.1 Functional Requirements.

4.5 Process Design

The program's design is concerned with translating the design into well-structured program modules.

4.6 System Flow

❖ **Outpatient Treatment**

Overall outpatient can be categorized into two parts which is outpatient that have appointment and those who have no appointment and may be new patients or patients who have already registered but wants to get treatment.

Usually when outpatient comes to get treatment, the patient normally already has an appointment for that day. Patient will go to the counter and give their Identity Card (IC). The Staff on duty will find out whether the patient has a record or not in the hospital. If not the Staff will register the patient record under *Registration module*. The Staff will give the patient a patient number. The staff will inform to the doctor by sending a message that he/she has new patient. However the patient with an appointment will inform the Staff on duty and give their IC, the Staff will confirm their Doctor by viewing the Patient *Appointment Scheduling* and send message to the doctor that their patient is already here. If there is an emergency and the Doctor is not in, his/hers patient will be handed over to another Doctor on duty. This will be done when the patient comes to the counter and the staff will inform them and send them to the new Doctor. All modification will be done under *Patient Appointment Scheduling*. Everything is the same as explains above.

Once the Doctor receives the message, the doctor will search for the patient's previous treatment... By selecting the date, he/she wants to look and select **view** button. This will link with *Treatment;* here the doctor can see the previous treatment. If he/she wants to add any treatment today, the doctor will choose the **add** button. If the patient, in last treatment have done Laboratory test the doctor can view by choosing the test the patient has already done. This will link to the *Laboratory.* If not, the patient can get the medicine at the pharmacy. Here the patient will give their IC, and pharmacist on duty will access the *Prescription,* search for the patient prescription and **view** the medicine he/she should give. If the patient has to do the laboratory test the Doctor will choose the test that the patient should do and send the message to the lab assisted, who is on duty. There the patient will show the IC and the laboratory staff will do the test and the result will be keyed in to the *Laboratory.* The patient who has completed the laboratory test will go to the accounts department to pay the fee for the test they have undergone.

If the patient has to come for a next treatment the doctor will check the doctor's schedule to find a suitable date. This is to ensure that the date will not overlap with other patients.

❖ **Inpatient Treatment**

For inpatient treatment the patient will see the doctor before the patient is admitted, if they do. The nurse will register the patient as inpatient under *Registration.* The nurse will view the *Ward management* module to see the empty ward and the class of ward the patient wants. Once the patient is already registered and have bed, the records will be updated in the *Billing.*

The Doctor will write the treatment or the patient status in the report beside the bed (it is a carbon copy), after that the doctor will update the record. Once the patient is

discharged, the patient will have to go to the account department to settle her/his fees. This will all be done under *Admission /Discharge* and combination with *Billing* module.

Yes

START

No

Cancel

Display Main Screen Menu

No

Choose Registration Module — Yes → Display Registration Submenu

No

Choose Scheduling Module — Yes → Display Scheduling Submenu

No

Choose Patient History Module — Yes → Display Patient History Submenu

No

Choose Ad/Disc Module — Yes → Display Ad/Disc Submenu

No

To Proceed

THE END
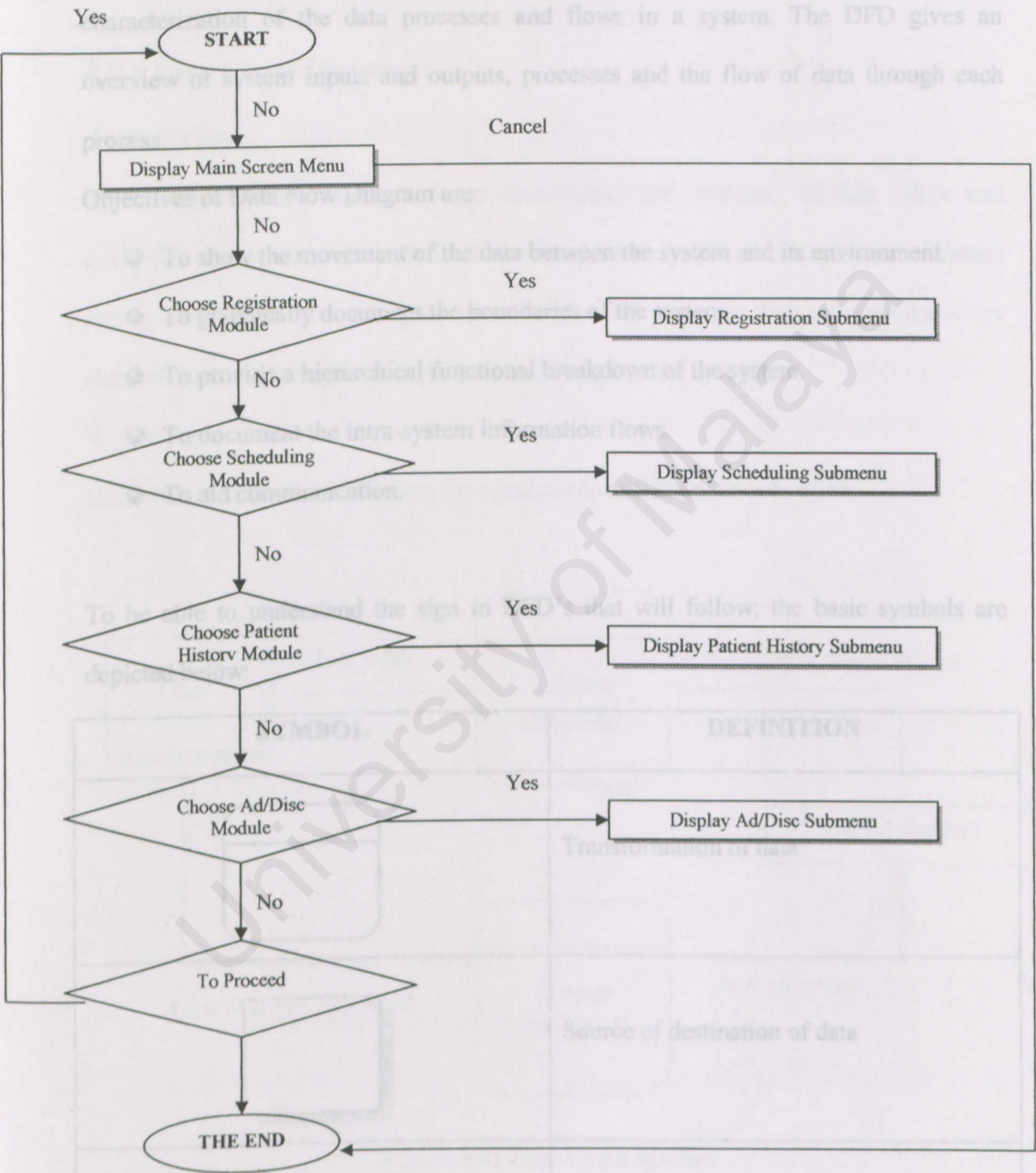
Figure 4.3: System Flow Chart

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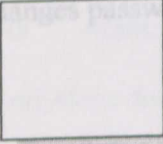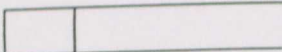
Data Flow Diagrams (DFD) is a technique used to show the graphical characterization of the data processes and flows in a system. The DFD gives an overview of system inputs and outputs, processes and the flow of data through each process.

Objectives of Data Flow Diagram are:

- To show the movement of the data between the system and its environment.
- To graphically document the boundaries of the system.
- To provide a hierarchical functional breakdown of the system.
- To document the intra-system information flows.
- To aid communication.

To be able to understand the sign in DFD's that will follow; the basic symbols are depicted below:

SYMBOL	DEFINITION
	Transformation of data
	Source of destination of data
	Data in static storage

63

	Flow of data

Table 4.1: Data Flow Diagram Symbols

4.7.1 User Login Overview

↓ **User Login**

When the user inputs the login identification and password, the data will be sent to the server. The server will then validate the password with the one stored in database server. Users with the correct password would be able to access their information, where else, unsuccessfully login would display a message on the system.

↓ **Change password**

Users have to provide the old and new passwords to change the passwords.
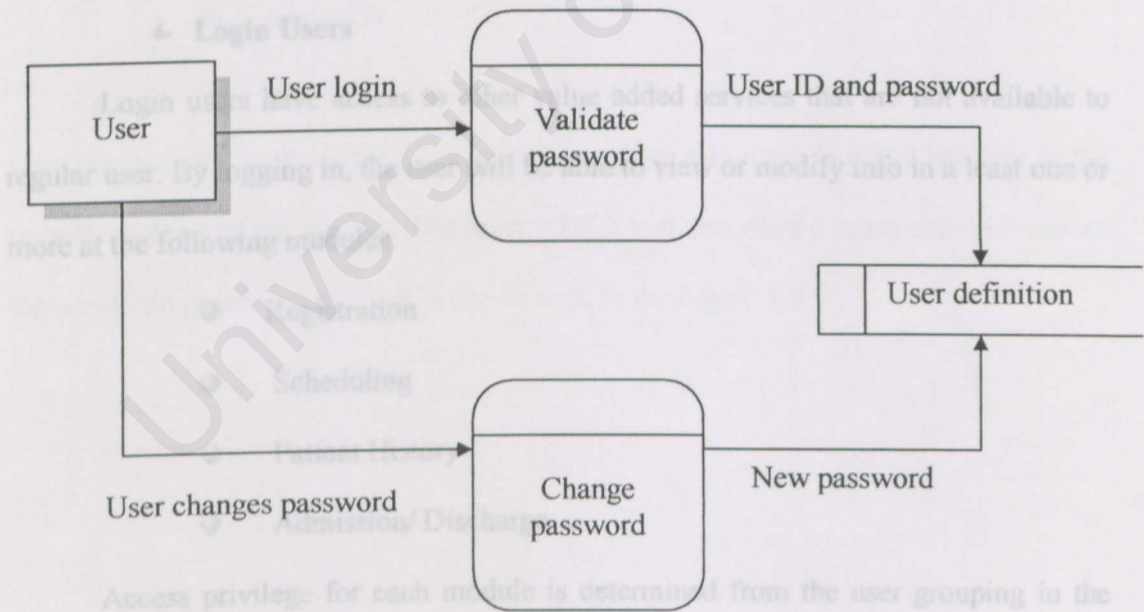
Figure 4.4: User Login Module

64

4.7.2 Overall Design Overview Data Flow Diagram

After much discussion, an overall view of how the different modules will interact with each other was thought up and a Level 1 DFD was drawn out to picture the overall system in graphical manner.

The User Definition table is the only shared data source of this level of the system design. It contains the user names and definition for every user in the system. Each module in the system accesses this table to verify users before they are allowed entry into any particular module.

The main entry point into the system is via the main page. At the main page the user can access more functions by logging into the system using specified username and password which are determined by the system administration.

↓ Login Users

Login users have access to other value added services that are not available to regular user. By logging in, the user will be able to view or modify info in a least one or more at the following modules:

- Registration
- Scheduling
- Patient History
- Admission/ Discharge

Access privilege for each module is determined from the user grouping in the User Definition table. By controlling user privileges access to the system it can be

controlled by the administrator. The purpose of limiting user access is to avoid accidental crashes of the system or corruption of the data sources.

Restricting site access also enables the administrators of the site to determine what went wrong should any user access cause the system to crash or lockup. In this way the cause of error can be determined without having to check the entire system. Besides in the PMCS the Services Center will contact the registered patient to verify their identity. A more detailed description of the how the login portion of the system works can be viewed in the Level 1 DFD in the Figure 4.5.

+ **Non-Login User**

The non-login portion of the site will have access to three of the main module use by the system as well as one more additional module, they are as mention below:

- Outpatient Registration
- About Us
- Services

User interaction for non-login users is kept to a bare minimum to avoid mishaps in the overall working of the system. The level 1 DFD that describes a more detailed view of the non-login portion of the PMCS can be seen in the Figure 4.6.

Figure 4.5: Level Data Flow Diagram Depicting the Penang Medical Centre System Login Module
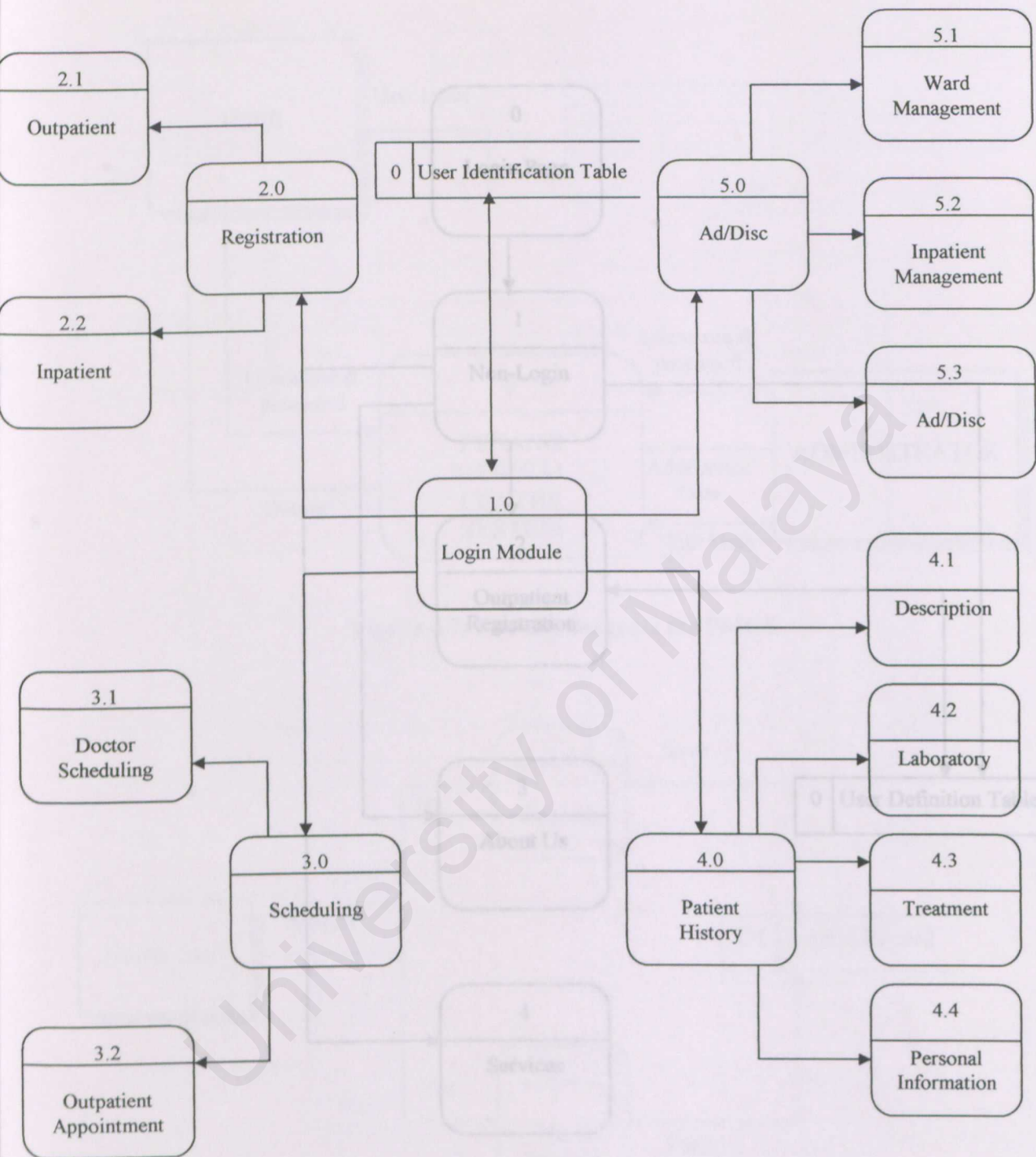
Figure 4.5: Level Data Flow Diagram Depicting the Penang Medical Centre System Login Module
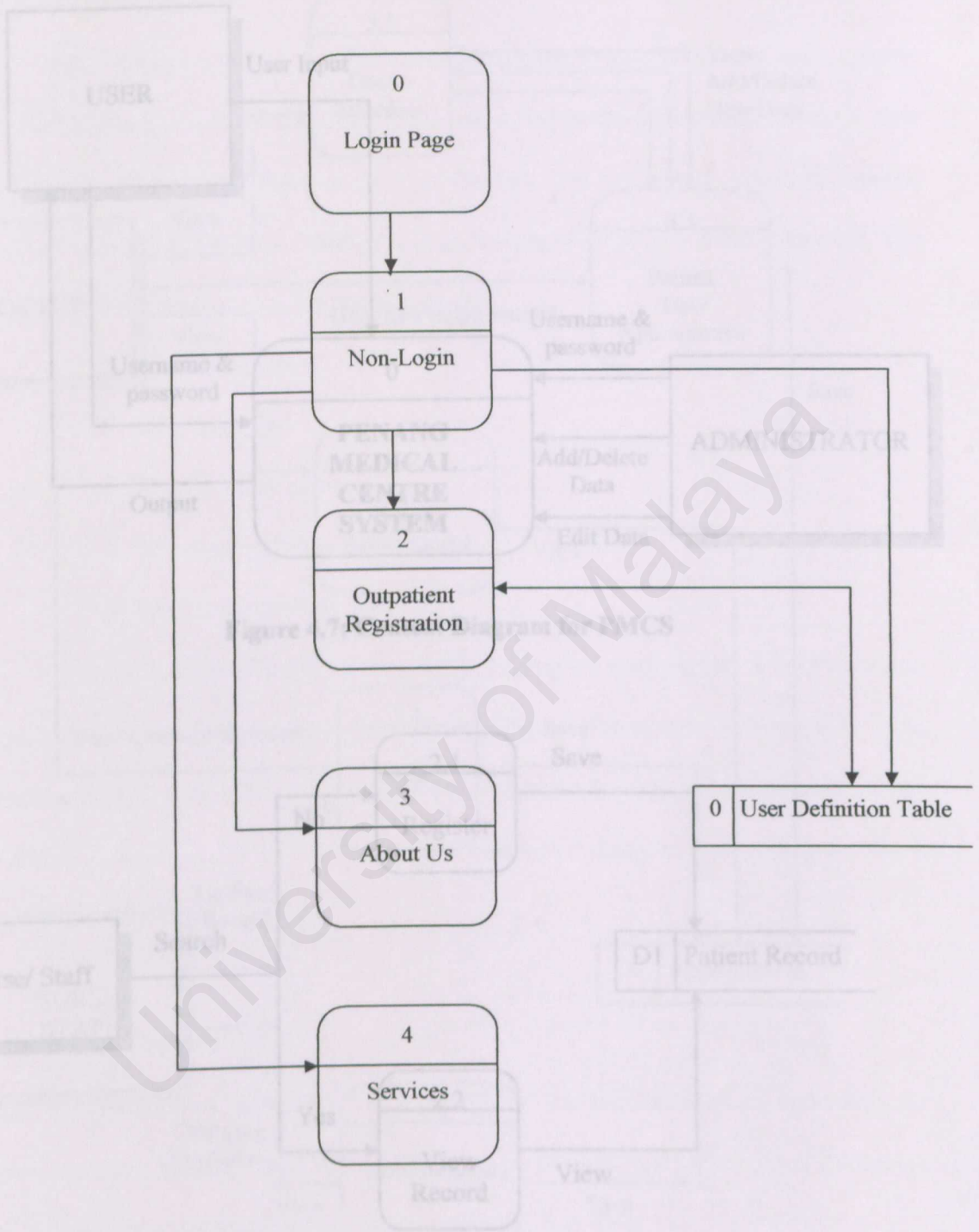
Figure 4.6: Level 1 Data Flow Diagram Depicting the Penang Medical Centre System Non-Login Modules
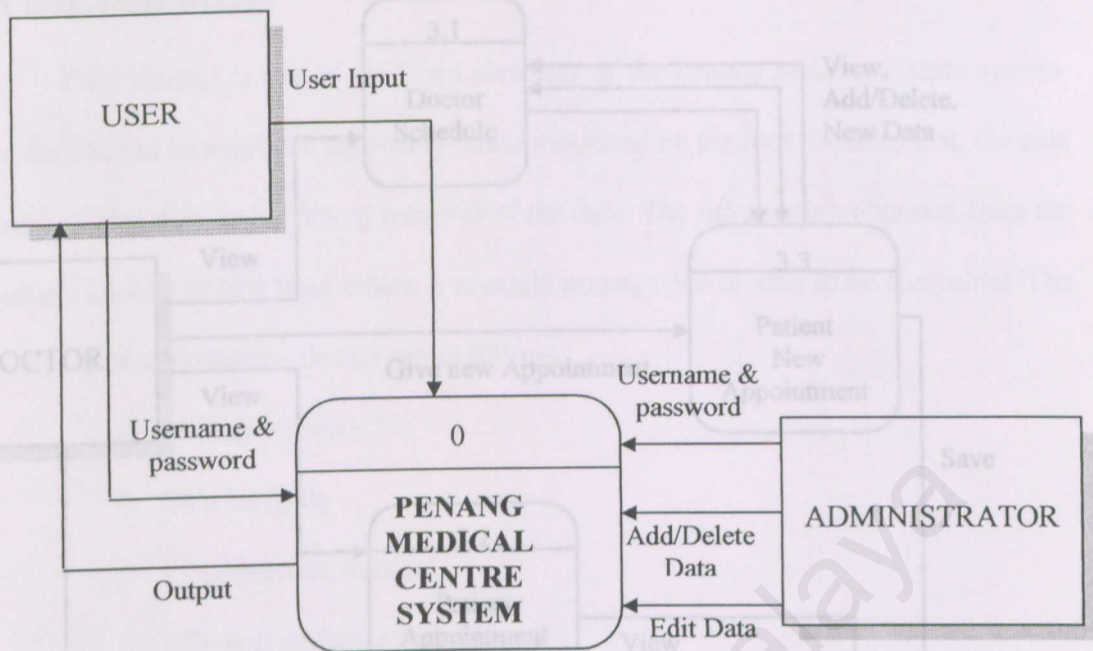
Figure 4.7: Context Diagram for PMCS
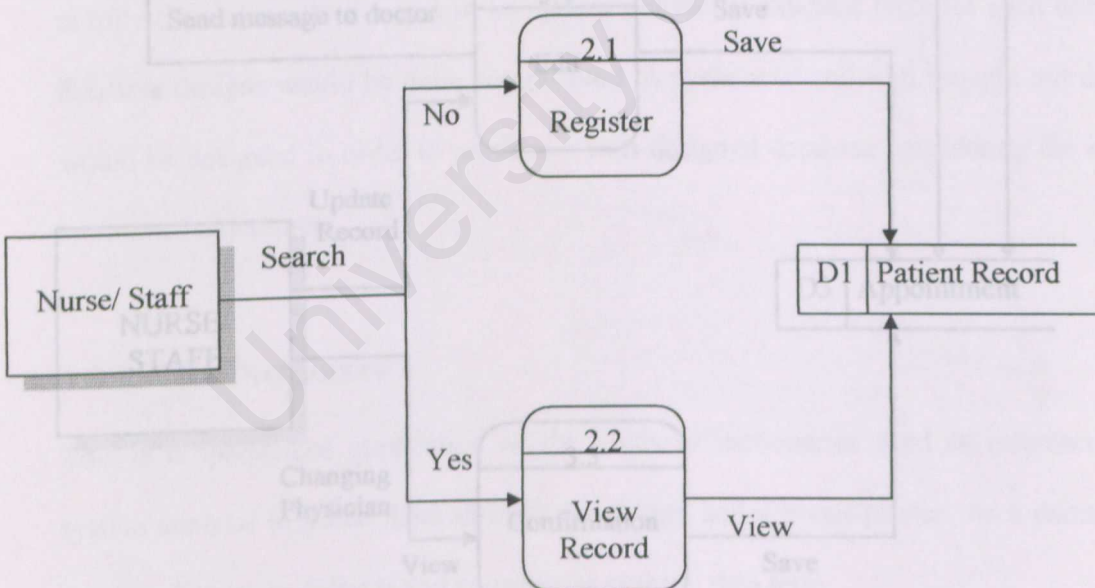
Figure 4.8: Child Diagram for Registration Module
Figure 4.9: Child Diagram for Scheduling Module

69

3.1
Doctor Schedule

View, Add/Delete, New Data

3.3
Patient New Appointment

DOCTOR

View

View

Give new Appointment

Save

3.2
Patient Appointment

View

Send message to doctor

3.4
Edit

Save

Update Record

D3 | Appointment

NURSE/ STAFF

Changing Physician

3.5
Confirmation

View

Save

Figure 4.9: Child Diagram for Scheduling Module

4.8 Database Design

Data storage is one of the main elements of the Penang Medical Centre system. The data has to be available especially when requested by the user. Besides that, the data must be up-to-date and efficient retrieval of the data. The information obtained from the database should be in a form where it is easily manageable or able to be controlled. The objectives of the database design are as follows:

- Data availability
- Data Integrity
- Efficient data storage
- Efficient updating and retrieval
- Useful information retrieval

Normally the detailed database would be designed concurrently when the system is fully developed. As the system comprises individual database table for each user, the database designs would be quite complicated. A systematic and well thought-out design would be designed in order to produce a well-designed database considering the above mentioned objectives.

4.8.1 Data Dictionary

This is a specialized application of the kinds of dictionaries used as references for system analysis to guide them through the design and analysis phases. As a document, the data dictionary collects and coordinates specific data term.

There are 13 tables involved, which are:

- Appointment

- Doctor
- Laboratory
- Login
- Login_admin
- Pnt_personalInfo
- Prescription
- Pnt_registration
- Staff
- Treatment
- Ward
- Wardmgmt
- Cnt

Primary key = †

Appointment

Field Name	Data Type	Description
app_id	AutoNumber	Auto number generated by MS Access for identification
doc_id	Text	Unique doctor Id assign to doctor
pnt_id	Text	Unique patient Id assign to patient
app_date	Date/Time	Date of appointment made
app_time	Date/Time	Time of appointment made

Table 4.2: Database Structure of Appointment Table

Doctor

Field Name	Data Type	Description
sno_ID	Autonumber	Auto number generated by MS Access for identification
doc_id	Text	Unique doctor Id assign to doctor
doc_name	Text	Doctor's name
doc_pass	Text	Doctor's password
doc_Specialist	Text	Doctor's specialization
doc_email	Text	Doctor's Email address
doc_phone	Text	Doctor's phone number

Table 4.3: Database Structure of Doctor Table

Laboratory

Field Name	Data Type	Description
pnt_ID	Text	Unique patient Id assign to patient
date_Test	Date/Time	Date when lab test was done
†TestID	Autonumber	Test Id number
test_Result	Text	Result for lab test
lab_asst	Text	Incharge Lab Asst

Table 4.4: Database Structure of Laboratory Table

Login

Field Name	Data Type	Description
pnt_id	Text	Unique patient Id assign to patient
password	Text	Patient password

Table 4.5: Database Structure of Login Table

Login_admin

Field Name	Data Type	Description
username	Text	Admin login name
password	Text	Admin password

Table 4.6: Database Structure of Login_Admin Table

Pnt_personalInfo

Field Name	Data Type	Description
† pnt_ID	Text	Unique patient Id assign to patient
pnt_bloodgroup	Text	Person's blood group
pnt_height	Text	Person's height
pnt_weight	Text	Person's weight
pnt_smoke	Yes/No	Person's smoking habit

Table 4.7: Database Structure of Pnt_personalInfo Table

Prescription

Field Name	Data Type	Description
Prescription_Id	Autonumber	Auto number generated by MS Access for identification
treatment_id	Number	Treatment Id
date_Pre	Date/Time	Date Prescribed
medicine	Text	Type of medicine
qty_id	Text	Quantity of medicine given
pharma_incharge	Text	Pharmacist incharge

Table 4.8: Database Structure of Prescription Table

Pnt_registration

Field Name	Data Type	Description
pnt_ID	Text	Unique patient Id assign to patient
today_date	Date/time	Today's date
schedule	Text	Inpatient or outpatient
name	Text	Patient name
marital	Text	Marital Status
gender	Text	Gender
dob	Date/Time	Date of birth
address1	Text	Person's address 1
address2	Text	Person's address 2
state	Text	State
email	Text	Email address
hphone	Text	Handphone number
ephone	Text	Emergency contact

Table 4.9: Database Structure of Pnt_registration Table

Staff

Field Name	Data Type	Description
SNO	AutoNumber	Auto number generated by MS Access for identification
STAFF_ID	Text	Unique staff Id assign to staff
STAFF_NAME	Text	Staff name
STAFF_PASS	Text	Staff password
STAFF_TYPE	Teat	Staff positions (type)

Table 4.10: Database Structure of Staff Table

Treatment

Field Name	Data Type	Description
pnt_id	Text	Unique patient Id assign to patient
date_treated	Date/Time	Treatment date
† treatmentid	Autonumber	Auto number generated by MS Access for identification
remarks	Text	Remarks by the doctor
doc_id	Text	Unique doctor Id assign to doctor
prescription	Yes/No	Prescription

Table 4.11: Database Structure of Treatment Table

Ward

Field Name	Data Type	Description
slno	Autonumber	Auto number generated by MS Access for identification
ward_id	Text	Unique ward Id assign to each ward
staff_id	Text	Unique staff Id assign to staff
num_of_bed	Number	Beds number
Remarks	Text	Remarks by the warden

Table 4.12: Database Structure of Ward Table

Wardmgmt

Field Name	Data Type	Description
† wardmgmt_id	Autonumber	Auto number generated by MS Access for identification
ward_id	Text	Unique ward Id assign to each ward

4.9 **User Interface Design**

The user interface design is the primary yardstick by which that system is judged. An interface that is hard to use will, at best, result in a high level of user errors Sommerville, 2006) with the user needs, experience and capabilities of the users should be anticipated in the system as to give a more "conservative" look. Therefore, the colors used through out this system were consistent. Error messages are provided by the system to aid users who make mistakes during the operation.

The user interface adopts the following design and engineering principles:

○ **Consistency**

The interface would be consistent and therefore the comparable operations should be activated in the same way. Consistency plays role in structuring and organizing the content of pages in a tidy, neat and standard method, which will not cause users lost in space where they visited before.

○ **Recoverability**

The interface would be including mechanism to allow users to recover from their errors.

○ **User Familiarity**

The interface would use terms and concepts, which are common and easy to use.

○ **User Guidance**

The interface would incorporate some form of context-sensitive guidance and assistance.

	Text	Unique patient Id assign to patient
pnt_id		
admit_date	Date/Time	Admitted date
admit_time	Date/Time	Admitted time
dis_date	Date/Time	Discharge date
dis_time	Date/Time	Discharge time
bed_no	Number	Bed number
dis	Yes/No	Discharge Process

Table 4.13: Database Structure of Wardmgmt Table

Cnt

Field Name	Data Type	Description
Staff	Number	Count staff
Patient	Number	Count Patient
Doctor	Number	Count Doctor

Table 4.14: Database Structure of Count Table

4.9 User Interface Design

"The user interface of a system is often the yardstick by which that system is judged. An interface that is difficult to use will, at best, result in a high level of user errors" [Sommerville, 2001]. It is with that in mind, the needs, experience and capabilities of the users. The uses of color have been limited for this system as to give a more "conservative" look. Therefore, the colors used through out this system were consistent. Error messages are provided by the system to aid users who make mistakes during the operation.

The user interface also will be designed as following principals:

⊙ **Consistency**

The interface would be consistent and therefore the comparable operations should be activated in the same way. Consistency plays role in structuring and organizing the content of pages in a tidy, neat and standard method, which will not cause users lost in space where they visited before.

⊙ **Recoverability**

The interface would be including mechanism to allow users to recover from their errors.

⊙ **User Familiarity**

The interface would use terms and concepts, which are common and easy to use.

⊙ **User Guidance**

The interface would incorporate some form of context-sensitive guidance and assistance.

77

⚫ Simplicity

There are less typing need, the instructions are simple and easy to be interpreted by the users.

⚫ Attractiveness

It brings the attentions of users using interesting pictures or icons rather than plain and bored textual displays.
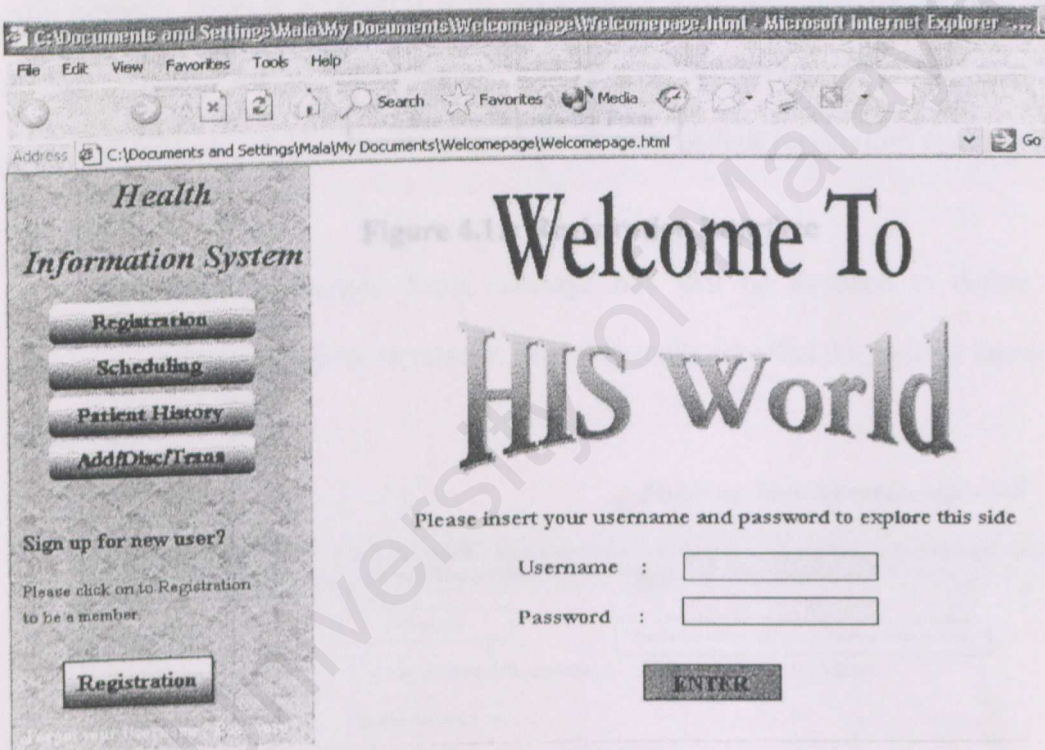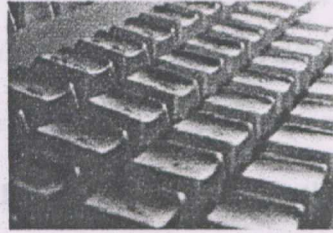

Figure 4.10: Login Interface

Forms play a very vital role as a tool of obtaining or capturing data in any information system. They are the most commonly used dialogue types for data entry.

... screen design where the user will be familiar with the ...

... user ...

... and less easy ... PMC's several sub entry forms were designed for the use of the system. Among the forms designed are patient registration form and physician appointment form. The example of the form for the patient registration can be seen in Appendix.

Home | **Find a Doctor** | **Patient Resources** | **Services** | **Search**

Pre-Registration

No one enjoys paperwork or the time spent filling it out.
But the speed of your registration process is enhanced
when we can obtain complete and accurate information before
you arrive.

To serve you, we've created an online pre-registration form
you can fill out right here and submit before you arrive at one
of our hospitals. The form requires you to provide some standard information.

The registration form is contained within a secure connection. This allows HIS to
protect your personal information. When entering this section, a security alert will appear. Please click OK to proceed. For
security purpose our Service Centre will contact the Registered Patient to verify their identity.

If you have any questions about the form or the online pre-registration process, or if you would like more information, please
contact our Service Centre at 04-3832247 or 04-3832249.

[Online Pre-Registration Form]

Figure 4.11: Registration Interface

In patient registration form, message box will be included to define what inpatient and what outpatient stands for. This is to make sure that the patients know what actually they are doing.

Patient Registration

Registration	
Patient Information	
Today's Date	27/06/02
I am scheduled for*	○ Outpatient Admission ○ Inpatient Admission
Location*	Kuala Lumpur
Name*	
Marital Status*	○ Married ○ Single ○ Widowed ○ Divorced
Sex*	○ Female ○ Male
Patient Birth Date*	
Address* Mailing address	
City*	
State*	

Figure 4.12: Registration Form

4.10 Forms Design

Forms play a very vital role as a tool of obtaining or capturing data in any information system. They are the most commonly used dialogue types for data entry. Forms are designed in system development as it can be used for the basis of data entry screen design where the user will be familiar with the data entry interface layout and most importantly transcription from paper to computer follows a sequence that the user know and the work of data entry into a computer system can be done easily and less error prone. In PMCS, several data entry forms were designed for the use of the system. Among the forms designed are patient registration form and physician appointment form. The example of the form for the patient registration can be seen in Appendix.

In patient registration form, message box will be included to define what inpatient and what outpatient stands for. This is to make sure that the patients know what actually they are doing.

Figure 4.12: Registration Form

4.11 Statement of Expected Outcome

As a conclusion, the expected outcome of this project would be a proper registration system which will reduce the time for filling the fields in the forms. Besides that this project also is a fully web-based and easy-to-use. The project has to accomplish the listed prerequisites and achieve all the objectives. The Penang Medical Centre System is expected to be stable and proficient in handling the requests of multiple users. Each and every user should not find any difficulties in using this system.

4.12 Chapter Summary

System design is a very important phase in developing a system, regardless it is a brand new system or modifying an existing system, because it is going to turn a system on paper to a real one that its target users are going to use, feel and experience it. Thus, every single part must be taken into count and careful consideration in order to avoid losing control in processing of system development. In this chapter, the system design is discussed from broad idea to a detailed definition, which is from system architecture to system process and data flow. Besides, functionality of the system architecture to system process and data flow. Besides, functionality of the system is described here where significant and important modules are identified. Lastly, outline for the structures of database and tables are created and the sample screen layout template design is constructed for future development references.

After the system design phase, the next stage to be discussed is the system implementation phase which is an important phase in translating the concept of the system into software representation.

Chapter 5
System Implementation

5.1   Introduction

After the system designing phase on how the system should be functioning, the next section will discuss the implementation phase. The implementation phase is an important element especially when the initiation is a project developed by a team of people especially in regards of a data-centred and database application.

System implementation is the ultimate completion of database and application design. It is identifies as the design phase which may or may not have involved prototyping. Once ready, both the application have to be implemented. Once the design and implementation of a compiler accepts sources code, in other and translates it in a machine-ready code as output. Compiler output is further utilized. The implementation phase will use the development processes micro-coded logic to the highest extent. This phase discuss the major works involved in the development of the system.

The effectiveness of this phase will greatly determines the success of the system and may like prototyping, programming, debugging, testing, validation, system installation and documentation.

5.2   Development Environment

The development environment has a certain impact on the development of a system. The choice of suitable and tools to get the work done is very important, either in software or hardware. This will influence the success rate of the project a great deal.

The hardware and software tools used in this system are as below in summarized version.

5.1   Introduction

After the system designing phase on how the system should be functioning, the next process will involves the implementation phase. The implementation phase is an important element especially when it involves a project developed by a team of people where integration of system is needed between sub systems.

System implementation is the physical realization of database and application design. On completion of the design stages (which may or may not have involved prototyping), here comes the stages where the database and the application have to be implemented. The transaction process continues when a compiler accepts sources code as input and produces machines- dependent object- code as output. Compiler output is further translated into machine code- the actual instruction that drives micro-coded logic in the central processing unit (CPU). Coding and debugging is the major works involved in the implementation phrase.

The effort spent in this phase will actually determines the success of the system and ease the processes of modification, debugging, testing, validation, system integration and for future enhancement.

5.2   Development Environment

The development environment has a certain impact on the development of a system. The choice of a right set of tools to get the work done is very important, either in software or in hardware. This will influence the success rate of the project a great deal.

The hardware and software tools used in this system are as below in summarized version.

○ **Hardware Used**

The hardware used to develop the system is as listed below:

+ Intel Pentium IV 2.00GHz

+ 256MB Random Access Memory

+ 18.61GB Hard Disc Drive

+ Other standard desktop PC accessories such as keyboard, mouse, monitor, and network card.

○ **Software Used**

The software used to develop the system is as listed below:

+ *Operating System*: Window XP Professional

+ *Database Management System*: Microsoft Access 2000

+ *Programming Tools*: Microsoft Front Page/ Macromedia Dreamweaver Ultradev

+ *Word Processor*: Microsoft Word 2000

5.2.1 Program Development and Coding

Program development is the process of creating the program needed to satisfy an information system's processing requirements. Development and coding is the phase where takes the longest time in the development life cycle. Therefore, using the right tool and the right way to develop the system are crucial in determining the success of the project.

For PMCS, it involves developing using different programming language from Actice Server Pages, JScript, VBScript, and Hyper Text Markup Language (HTML). Before starting on the coding process or any other detailed works on the program, a

review on the program documentation needs to be done followed by design of the program and finally going into the program coding process.

5.2.1.1 Review the Program Documentation

The first and foremost step to be taken in program development phase is to review the program documentation that was prepared during the earlier phases. The program documentation prepared in the System Design of Penang Medical Centre consists of architectural view, functional design, database design, process design, modules flow diagrams, data dictionary, and also the sample layout of the interface. The documentation provides a guide and an understanding of the works that need to be done in the coding phase.

5.2.1.2 Designing the Program

After reviewing the program documentation, designing the program is the next following process after that. For this phase, determining how the program can accomplish the features and functions that are described in the program documentation and developing a logical solution to the program problem is done. The logical solution or the logical of the program is a step-by-step solution to the programming problems.

5.3   System Development

This system development is done with the help of proper software tools. Coding is a important tasks to ensure system development follow the needs of system chracterizations.

5.3.1 Coding

Coding that being used to developed the system is programming language such as ASP,

VBScript and Jscript.

Before the system development is proceeed fully, coding for each application is

identified. The main applications that being used in this system development are as

follows:

    i.   Add new data

    ii.   Delete old data

    iii.   Error check

    iv.   Data Connection

5.3.1.1 Add new data

```
<%
dim RS,RS1 set my_conn= Server.CreateObject("ADODB.Connection")
my_Conn.Open SourceDSN SET
RS=SERVER.CreateObject("ADODB.RECORDSET") SET
RS1=SERVER.CreateObject("ADODB.RECORDSET") RS1.OPEN "select * from
cnt",my_Conn,1,3 RS.Open "select * from staff",my_Conn,1,2
rs1("staff")=rs1("staff")+1 rs1.update RS.AddNew rs(1)="stf"& rs1("staff")
RS(2)=Request.Form("txtStaffName") RS(3)=Request.Form("txtStaffPassword")
RS(4)=Request.Form("cmbPosition")

RS.Update
%>
```

5.3.1.2 Delete old data

```
<%

dim rs, sql, delete
delete = false
set my_conn= Server.CreateObject("ADODB.Connection")
my_Conn.Open SourceDSN
set rs=server.CreateObject("adodb.recordset")
rs.Open "select * from staff",my_conn, 1, 3
while not rs.eof
        if request.form(rs("staff_id"))= "ON" then
                rs.delete
                delete=true
        end if
        rs.movenext
wend
if delete then
        response.redirect "deldocresp.htm"
else
response.redirect "delstaff.asp"
end if

%>
```

5.3.1.3 Error Checking

```
<script LANGUAGE="JavaScript">

function CheckError() {

        if (document.reg.txtStaffName.value==""){
        alert("Insert Staff Name.");
        document.reg.txtStaffName.focus();
        return false;
         }

        if (document.reg.cmbPosition.value=="Choose One"){
        alert("Select Position.");
        document.reg.cmbPosition.focus();
        return false;
         }
```

```
        if (document.reg.txtStaffPassword.value ==""){
        alert("Insert the password.");
        document.reg.txtStaffPassword.focus();
        return false;
        }

        if(document.reg.txtStaffPassword.value!=document.reg.txtStaffConfirm.
        value) {
        alert("Make sure the password and retype password are same.");
        document.reg.txtStaffConfirm.focus();
        return false;
        }
    }
</script>
```

5.3.1.4 Data Connection

```
<%
dim RS,con
set my_conn= Server.CreateObject("ADODB.Connection")
my_Conn.Open SourceDSN
SET RS=SERVER.CreateObject("ADODB.RECORDSET")
RS.Open "select * from STAFF order by sno",my_Conn,1,3 %>

%>
```

5.4 Programming Guidelines

The design is a guide to the function or purpose of each component, but the programmer has great flexbility in implementaing the design as code. The design or requirements specification may suggest a programming language, either directly because it is specified by the designers or customers, or indirectly because of the constructs used. In developing this Penang Medical Centre, ASP programming language is chosen to translate the designs to code.

No matter what language is used, each program component involves at least threec major aspects: control structures, algorithms, and data structures.

Control structures

It is important for program structure to reflect the design's ontrol structure. Readers should not have to jump widly through the code, marking sections to which to return and wondering whether they have followed the right path. They should concentrate on what is being done by the program, not on the control flow. Thus, many guidelines and standards suggest that the code be written so anyone can read a component easily from the top down.

Generality is a virtue in writing code; do not make code more specialized than it needs to be, but do not make the components so general that performance and understandings are affected. Other design characteristics translate to sode omponents, such as coupling and cohesion. When writing programs, using parameter names and comments that exhibit the coupling among components is advisable.

✦ Algorithms

The program design often specifies a class of algorithms to be used in coding the component that is written,however there is a great deal of flexibilty in converting the algorithm to code, subject to the constraints of the implementation language and hardware.

Performance or efficiency of implementation: it is good to make the code run as fast as possible, however, making the code faster may involve hidden costs:

- ❖ The cost to write the faster code, which may be more complex and thus take more time to write
- ❖ The cost of time to test the code, whose complexity requires more test cases or test data
- ❖ The cost of time for users to understand the code
- ❖ The cost of time to modify the code, if neccessary

Thus, execution time is only a small part of the overall cost equation. Execution time must be balanced with design quality, standards, and customer requirements. In particular, sacrifice of clarity and correctness for speed is not advisable.

✦ Data Structures

In writing programs, data should be formatted and stored so that data management and manipulation are straightforward. There are several tecniques

that use the structure of the data to suggest how the program should be organized.

Keeping the program Simple. The program's design may specify some of the data structures to be used in implementing funtions. Often, these structures are chosen because they fit into an overall scheme that promotes information hiding and control of component interfaces.

Using a Data Structure to Determine a Program Structure. In general, data structures can influence the organization and flow of a program. In some cases, the data structures can influence the choice of language, too. For example, LISP is designed to be a lis processor, and it contains structures that make it much more attractive than some other languages for handling lists. Similarly, Ada and Eiffel contain constructs for handling unacceptable states called exceptions.

5.4.1 General Guidelines

Several overall strategies are useful in preserving the design quality in code.

+ **Localizing Input and Output**. Those parts of a program that read input or generate output are highly speacilized and must reflect characteristics of the underlying hardware and software. Because of this dependence, the program sections performing input and output functions are sometimes difficult to test. In fact, they may be the sections most likely to change if the hardware or software is modified. Therfore, it is desirable to localize these sections in generalization of the overall system. Other systemwide functions to be performed on the input (such as formatting or type checking) can be included in the specialized

component, relieving the other components of the burden and thus eliminating repetition. Similarly, putting output functions in one place makes the system easier to understand and change.

✦ **Including Pseudocode**. The design usually lays out a framework for each program component. Since the design is an outline of what is to be done to the code, rather than to translate the design immediately into code. Pseudocode can be used to adapt the design to the chosen language. In this way, code can be rearranged and restructured with a minimum of rewriting. Thus, pseudocode has acted as a framework on which to construct the code.

✦ **Revising and rewriting, not patching.** When writing code, as when preparing a term paper or creating a work of art, a rough draft is often written. Then processes of revising and rewriting is repeated till programmers are satisfied with the result.

✦ **Reuse.** There are two kinds of reuse: producer reuse, where components are created designed to be reused in subsequent application, and consumer reuse, where components that were originally developed for other projects are being used. Being a consumer for a project, there are four characteristics to check about the components that are about to reuse:

1. Does the component perform the function or provide the date you need?

2. If minor modification is required, is it less modification than building the component from scratch?

3. Is the component well-documented, so you can understand it without having to verify its implementation line by line?

4. Is there a complete record of the component's test and revision history, so you can be certain that it contains no faults?

If being a producer of reusable components, several things must be kept in mind:

- Make the component general, using parameters and anticipating similar conditions to the ones in which your system will invoke your components.

- Separate dependencies so sections likely to need change are isolated from those that are likely to remain the same.

- Keep the component interface general and well-defined.

- Include information about any faults found and fixed.

- Use clear naming conventions.

- Documentation the data structures and algorithms.

- Keep the communication and error-handling sections separate and easy to modify.

5.5 Documentation

Many corporate or organizational standards and procedures focus on descriptions accompanying collection of programs. Program documentation is considered to be the set of written descriptions that explain to a reader what the programs do and how they do it. Internal documentation is descriptive material written directly within the code; all other documentation is external documentation.

5.5.1 Internal Documentation

The internal documentation contains information directed at someone who will be reading the source code of the programs. Thus, summary information is provided to identify the program and describe its data structures, algorithms, and ontrol flow. Usually, this information is placed at the beginning of each component in a set of comments called the **header comment block.**

+ **Header Comment Block.** Just as a good newspaper reporter includes the who, what, where, when, how and why of a story, the following information must be included in the header comment block for each component:

  1. What your component is called?

  2. Who wrote the component?

  3. Where the component fits in the general system design?

  4. When the component was written and revised?

  5. Why the component exists?

  6. How the components uses its data stuctures, algorithms and control?

+ **Other Program Comment.** The header comment block acts as an introduction to program, much as the introduction to book explains its purpose. Addtional comments enlighten readers as they move through the program, helping them understand how the intentions programmers describe in the header are implemented in the code.

For example:

If (value=="Y")

{counter=counter+1;}//increase counter by 1 if encounter value equals to 'Y'

- **Meaningful Variable Names and Statement Labels.** Choose names for the variables and statements that reflect their use or meaning. Writing

  weekwage = (hrrate * hours) + (.5) * (hrrate) * (hours - 40.);

  Make more sense to the reader than

  z = (a*b) + (.5) * (a) * (b-40.);

- **Formatting to Enhance Understanding.** The format of comments can help a reader to understand the goal of the code and how the goal is reached. Indentation and spacing of statements can reflect the basic control structure. For example:

  if (age < 55) benefits = minimum;

  else if (age < 65) benefit = minimum + bonus;

  else if (age < 75) benefit = minimum * 1.5 + bonus;

  else benefit = maximum;

  is much more better than

  if (age < 55)

  benefits = minimum;

  else if (age < 65)

  benefit = minimum + bonus;

  else if (age < 75)

  benefit = minimum * 1.5 + bonus;

  else benefit = maximum;

- **Documenting Data.** Program readers will find very difficult to understand the way in which data are structured and used. A data map is very useful in interpreting the code's actions, especially when a system handles many files of

varying types and purposes, coupled with flags and passed parameters. This map should correspond with the data dictionary in the external documentation, so the reader can track data manipulation through the requirements and design to the code.

5.5.2 External Documentation

External documentation is intended to be read by those who may never look at the actual code. Designers may review the external documentation when considering modifications or enhancements. Besides, the external documentation gives programmers a chance to explain things more broadly than might be reasonable within their program's comments. If header comment block is considered as an overview summary of aprogram, then the external documentation is the full-blown reports. It answers the same questions – who, what, why, when, where and how – using a system, rather than a component, perspective.

Because a software system is built from interrelated components,the external doumentation often includes an overview of the system's components, or of several grouping of components (such as the user-interface components, or the database management). Diagrams, data flow and share between components are shown in external documentation.

+ **Describing the Problem.** In the first section of the code's documentation, what problem is being addressed by the component should be explained. It describes what options were considered for solutions and why a particluar solution was chosen.

- **Describing the Algorithms.** Each algorithm used by the components should be explained, including formulas, boundary or special conditions, and even its derivation or reference to the book or paper from which it is derived.

- **Describing the Data.** In the external documentation, the users or programmers should be able to view the data flow at the component level. Data flow diagrams should be accompanied by relevant data dictionary references.

5.6 Data Validation

Data Validation is performed before record is inserted into the database. The purpose of this feature is to make sure invalid data will not insert into the database and cause error.

5.7 Security Management

Security Management that needed in this system is based on access level need by users. Just because this system is not a system which based on Electronic Commerce (*e-commerce*), security in accessing the web is not that critical. But, if the database provide services to multiple group of users, hence emphasis to relevant security should be applied. Generally, user should see/access data related to them only, this to avoid them accidentally delete or destroy the data.

In this system implimentation, users is given a password for accessing the informations in the database. Access level for each user is different from the other users. Session object is used to perpetuate the valid login information and user level in whole sesion.

In the interface that used limited accessing, there is a code for validation of the interface access. If unauthorized user access the system, they will get error message.

5.8 Chapter Summary

As a conclusion this chapter explain about the steps that taken in the procedure valution and the proper way that was used in modelling the programming language. Usage of programming language such as ASP in certain part of the system is explained in detail.

Besides, this chapter also discuss about the main issues that occurs and the way to overcome the problems. After the details explaination of the implementation phase, the next chapter will discuss about the testing phase. This is also a veru important stage whereby testing is essential to ensure quality of the system and implementation assures that the system being developed is operational and then allowing the users to take over its operation to use.

6.1 Introduction

Testing is carried out to discover different classes of errors in software development. Besides that, testing is also a critical element in uncovering logical error and the test the system reliability. The goal is to design tests that will uncover the greater number of errors or classes of errors with the minimum amount of time and effort. It demonstrates that the software functions appear to be working according to the specifications and performance requirements that have been met.

However, testing cannot show the absence of defects, it can only show that software defects are present.

Glen Myers states a number of rules that can serve well as testing objectives:

- Testing is a process of executing a program with the intent of finding an error.
- A good test case is one that has a high probability of finding an error.
- A successful test is one that uncovers a yet undiscovered error.

In developing a system, testing usually involves several stages. First, each program component is tested on its own, isolated from the other components in the system. 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 is carries out all functions it is supposed to carry out. This stage of testing verifies that 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

Chapter 6
System Testing

6.1   Introduction

Testing is carried out to discover different classes of errors in software development. Besides that, testing is also a critical element in uncovering logical error and the test the system reliability. The goal is to design tests that will uncover the greater number of errors of classes of errors with the minimum amount of time and effort. It demonstrates that the software functions appear to be working according to the specifications and performance requirements that have been met.

However, testing cannot show the absence of defects, it can only show that software defects are present.

Glen Myers states a number of rules that can serve well as testing objectives:

- Testing is a process of executing a program with the intent of finding an error.

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

- A successful test is one that uncovers a yet undiscovered error.

In developing a system, testing usually involves several stages. First, each program component is tested on its own, isolated from the other components in the system. 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 is carries out the functions it is supposed to carry out. This stage of testing verifies that 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

group of integrated modules to verify that the system components work together as described in the system and program design specifications.

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

6.2   Test Planning

Test planning helps in designing and organizing tests, so that testing is carried out appropriately and thoroughly. The steps for the test planning are:

a) Establishing test objective

b) Designing test case

c) Writing test case

d) Testing test case

e) Executing test

f) Evaluating test results

Besides, sistem development may also facing bugs problem which are codes that effectuate fault and error in system operasion. Figure 6.1 is the guide of effort to debugging the system.

```
┌─────────────────────────────────────────┐
│   Find the source where the error occurs │
└─────────────────────────────────────────┘
                    │
                    ▼
┌─────────────────────────────────────────┐
│        Identify error/fault type         │
└─────────────────────────────────────────┘
                    │
                    ▼
┌─────────────────────────────────────────┐
│   Creating the program to be easier using│
│              the test case               │
└─────────────────────────────────────────┘
                    │
                    ▼
┌─────────────────────────────────────────┐
│       Creating a test environment        │
└─────────────────────────────────────────┘
```
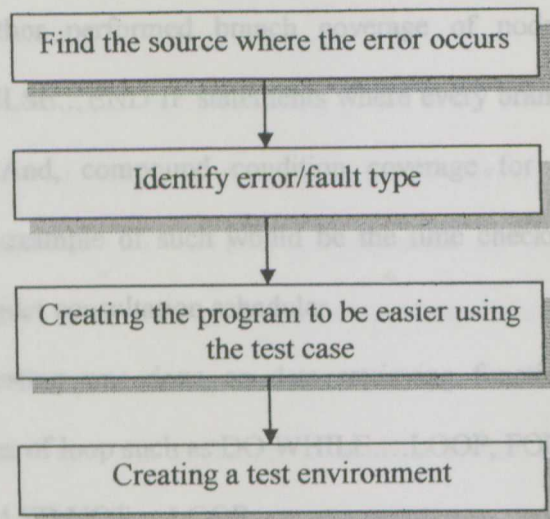
Figure 6.1: System debugging guide

System debugging process is done before testing phase and started before the

implementation stage (together with the source code development).

6.3   Types of Testing

The types of unit testing were deliberated in this project are shown as follows:

- Black Box and White Box Testing

- *Top-down* and *Bottom-up* Testing

6.3.1 Black Box and White Box Testing

1. White Box Testing

White box testing basically involve looking at the structure of the code. It

focuses on the idea of coverage. The main objective would be to check for

missing function.

104

The author performed branch coverage of node testing for those If.....THEN...ELSE...END IF statements where every branch/decision is tested at least one. And, compound condition coverage for multiple conditions statements, an example of such would be the time checking statement in the healthcare provider consultation scheduler.

Loop testing was done on data retrieving functions where there are extensive usages of loop such as DO WHILE....LOOP, FOR...TO...NEXT, and DO WHILE...AND NOT....LOOP. Testing control structures of a procedural design. Can derive test cases to ensure:

- ❏ all independent paths are exercised at least once.
- ❏ all logical decisions are exercised for both true and false paths.
- ❏ all loops are executed at their boundaries and within operational bounds.
- ❏ all internal data structures are exercised to ensure validity.

Why do white box testing when black box testing is used to test conformance to requirements?

- ❏ Logic errors and incorrect assumptions most likely to be made when coding for "special cases". Need to ensure these execution paths are tested.
- ❏ May find assumptions about execution paths incorrect, and so make design errors. White box testing can find these errors.
- ❏ Typographical errors are random. Just as likely to be on an obscure logical path as on a mainstream path.

2. Black Box Testing

Black box testing focused on functionality of the code. The main objectives 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 used equivalence class partitioning. In equivalence class partitioning the author runs one test for each class of input to the module and then run additional tests using invalid data to make sure the error routines are 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 included test scenarios where the value sets is inside, on the outside the boundary.

Below listed the summary of units that were independently unit-tested:

a) Opening and closing of connection to the database

b) Inserting of new data into database

c) Modification of existing data in the database

d) Retrieving data from database

e) Validation of user input data before submission

f) Validation of user identity before granting permission to hospital admin page

g) Returned query results from search module

6.3.2 *Top-Down* and *Bottom-Up* Testing

1. *Top-Down*

Top-down testing is the top level, usually controlling component,and is tested by itself. Then all components called by the tested component(s) are combined and tested as a larger unit. This approach is reapplied until all components are incoporated.

2. *Bottom-up*

Bottom-up is the popular approach for merging components to test the larger system. When this method is used, each component at the lowest level of the system hierarchy is tested individually first. Then, the next componentsto be tested are those that call the previously tested ones. This approach is followed repeatedly until all components are included in the testing.

6.4   Testing Technique Chosen

After analyse all the testing tecnique which was discuss above, I feel that *bottom-up* (Figure 6.2) is more favourable and more applicable to this system. Through this testing, developer can test the module in its lower position first and followed by the module that call the tested component. This testing continue until all module is tested.

Test
E

Test
B, E, F

Test
F

Test
C

Test
A, B, C,
D, E, F,
G

Test
G

Test
D, G

Figure 6.2: *Bottom-Up* Testing

6.5 Testing Levels

There are four testing steps which used in this project. They are as follows:

a) Unit Testing

b) Integration Testing

c) Acceptance Testing

d) Complete System Testing

Figure 6.3 is about the relationship between the tests which is done.

| Unit Testing | System Testing | Integration Testing | Complete System Testing | Acceptance Testing |
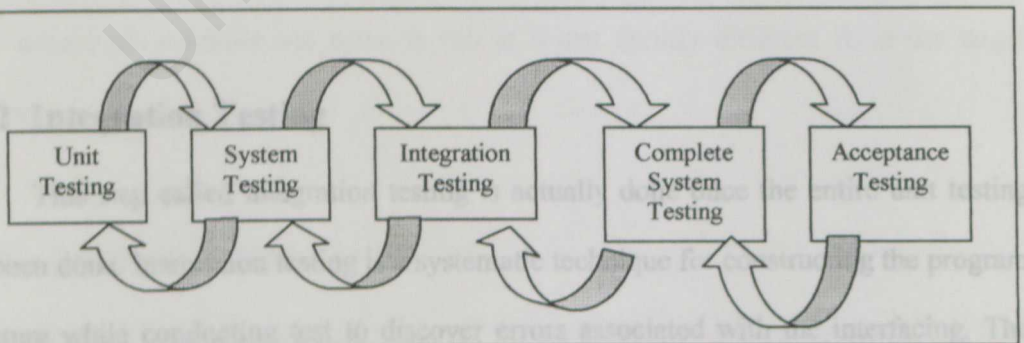
Figure 6.3 Steps in testing process

6.5.1 Unit Testing

Unit testing involves testing of components in a sub function followed by testing of each sub function in module such as whether the right data is deleted from the right table and finally proceed to the testing of the module itself [Pressman]. For example, these components could be anything from checking the validity of input value for email address to parsing a sentence into each individual word etc.

Each sub module of every module was tested to make sure problems were not present, like bugs. Artificial but acceptable test values were entered into each from that accepts data from the user so that user input into the system could be tested.

Thus, unit testing was completed. Logical dependencies were also tested. Tested values were included in the code so that the testing of every important logical statement that would access a particular segment of code could be done.

Logical dependencies were omitted at the sub module level itself. This was done by taking a few precautionary steps. Code segments were rearranged and loop structures were examined as utterly as possible to make certain that all code segments could be reached and would work as expected. Dependencies and their effects were determined by scrutinizing data and logical structures that exist in the code being developed.

6.5.2 Integration Testing

This step called integration testing is actually done once the entire unit testing has been done. Integration testing is a systematic technique for constructing the program structure while conducting test to discover errors associated with the interfacing. The objective of this testing is to take the unit tested module of PMCS and build a program

structure that has been dictated by design. This testing will ensure that the interface such as calling sequence in PMCS is arranged correctly.

Although in the PMCS unit testing was done, there was not much impediments encountered to it. This was because most of the modules were already tested and were functioning quite normally already. But, this does not mean that no adjustments were required. Some fine-tuning had occurred. Much more attention was given to integration testing, though. This was undoubtedly because this project takes the integration of its sub modules as one of the integral part of its research.

Integration tests, also known as modular testing, basically resolves around the verification of all the components working together as par with the description rendered in the system design specifications. The system is hierarchy of components. It is where each component belongs to a layer of design.

6.5.3 Acceptance Testing

In this test, users are required to use program with the help of developer from the installation's step until its operation. The purpose of this step is to test the user understandings of each interface and its functions. The acceptance test is sometimes run in its actual environment but often is run at a test facility different from the target location.

6.5.4 Complete System Testing

System Testing consist of series of different tests which verify that all system elements such as the hardware, software and information, have been properly integrated

110

and perform allocated functions. It also tests the flow of data through the center system. This type of testing was given the most prominence and importance. This is done to uncover any limitations to the system itself, gauge its capabilities and make sure that the entire system is working according to user's specifications.

If the users are satisfied with the system characteristics, the system is ready to be deployed for use. But, if there is a need for change, system modifications would be meted out to meet the user's expectations and requirements to the specifications and functionalities.

The results would indicate whether or not the complete system specifications and objectives have been met. Among the types of testing involved are fundamental and function tests.

6.5.4.1 Fundamental Tests

There are number of these tests that are difficult to measure with some amount of precision. There tests are usually treated as complements to the individual functional and further functional tests. These functional tests would be explained briefly in the next section. For now, the fundamental tests are introduced below:

- **Performance**

The main criterion that is seen is the response time. Performance tests are conducted to make certain that the system's response time is met by user expectations and also that it doesn't exceed the predetermined performance criteria under heavy volume and stress of all sorts. The purpose of performance tests is also

to test run the performance of various functions of the software within a specified hardware configuration. Stress testing can be added to the above explained test.

+ **Reliability**

Reliability testing is monitoring the mean time between failures. According to mathematical models of software reliability, the reliability tests are conducted. Consistency and reliability testing are parallel particularly when the system behaviour is measured for consistency. For example, inputs, outputs and response time.

+ **Serviceability**

This is regarding the ability of the support personnel manning the phone line to acquire enough information to enable the maintenance organization to asses the error and fix it without additional information in the event of crash of the user's software system.

+ **Usability**

This usually concerns the fact that building user interfaces that have pattern already familiar to the typical user. The interface should not be something totally new, which would require much scrutinizing from the user. A familiar surrounding which would lead the user to then use the software through pattern matching and paradigm shifts, something which everyone would do whilst mastering a product is in progress.

+ **Installability**

This installability is based on how easy for a novice to install the software correctly and easily without many impediments independently.

6.5.4.2 Function Tests

Function testing is based on the system functional requirements. The testing is carried out for four modules. The 4 modules are: *Registration, Scheduling, Patient History and Admission/Discharge*. Each module is tested individually to determine whether the system performs as required, further functional tests involve:

- **Documentation Testing**

  All examples used in the user's manual are tested for accuracy and whether the manual reports the correct answers when users obtain when they run the examples.

- **The event list**

  All the possible triggers must be exercised and the expected result compared with the actual results.

- **Specified scenarios**

  The entire set of possible scenarios, or user profiles, specified for an application.

- **Transaction tracking**

  A list of possible transactions, either extracted from the scenarios or from the event list, is tracked through the software system to ascertain that they function correctly from "input" to "output".

- **Error Message Testing**

  Every error message which can be generated by the system extracted from the code and placed in a table, where it can be generated and tested for appropriateness and understandability from the user's perceptive, with the proper response being prompted for.

113

6.6 Testing with The Data and Life Data

Before the system is put into production, all programs are desk checked, checked with test data and checked to make sure that all the modules work together with one another as planned.

Program Testing with Test Data

At this stage, first of all, all the programs written is desk checked to verify the way the system will work. Each steps in the program is checked to ensure the routine works as it is written. This is followed by both valid and invalid data test. These data are run to see if base routine work and also to catch errors. Test data include possible maximum and minimum values, as well as possible in format and codes. Outputs from test data are carefully verified.

Link Testing with Test Data

When programs pass desk checking and checking with test data, it went through link testing. Link testing checks to see if programs that are interdependent actually work together as planned. A small amount of test data are designed to test system specifications as well as programs, is used for link testing. It takes several passes through the system to test all combinations. This because it is immensely difficult to unravel problems if everything is test at once.

Test data that are used cover variety of processing situations for link testing. First, test data are processed to see if the system can handle normal transactions. If the system works with normal transactions, the variations are added, including invalid data used to ensure that the system can properly detect errors.

114

✦ **Full System Testing with Test Data**

When link tests are satisfactorily concluded, the system as a complete entity is tested as well, test data created for the express purpose of testing system objectives are used. System testing includes reaffirming the quality standards for system performance that were set up when initial system specifications are made. Everyone involved in the system once again agree on how to determine whether to system is doing what in the system once again agree on how to determine whether to system is doing what it is supposed to do. This will include measures of error, timeliness, ease of use, proper ordering of transactions, acceptable down time, understandable procedures manuals and others.

✦ **Full System Testing with Life Data**

When system testing with the test data proves satisfactory, the new system is tried with several passes on what is called as "life data" – data that have been successfully processed through the existing system. This allows an accurate comparison of the new system's output with what that is known to be correctly processed output, as well as a good feel of how actual data will be handled.

6.7 Program Testing

Program testing will include judgement on ASP script structure and Microsoft Front Page in the segment program. Normally the tests that are carried out are:

❍ **Segment coverage**- Between script segment and also between controls structures are done in each command at least once.

115

- **Path Test**- This specific test is actually about notifying and testing the paths between the script programs.
- **Data flow test**- To identify each specific variable through each counting, and later on defining the middle path set between the program script codes. This test is conducted manually because it is not effectively supported.
- **Loop Test**- This test includes single loops, connected loops and also nested loops.

6.8 Testing Analysis

After all test are carried out, the system is ready to be operated on the internet. After launching in the internet, the user assurance test can be conducted by getting a response from the user whom are using this system. Further on more changes will be done to establish a better system. This system will also be changed and updated from time to time.

6.9 Chapter Summary

This particular chapter tells us more on the testing phase. Testing is the most critical phase to make sure the system to compromise the need of the user. Just because of this, a good and systematic test should be done to have a full test. The test done are also compromising with the objectives given. This system actually goes through three main stages which are, unit testing, integration test and receiving test.

Having discussed about the system testing, the next chapter will see the system evaluation. This coming chapter will touches various things like problems encountered during the development process, system strength weakness and others.

Chapter 7
System Evaluation

7.1 Introduction

At all phases of the system approaches, evaluation is a process that occurs continuously, drawing on variety of sources and information. The role of this evaluation phase was to determine:

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

System evolution, like software evolution, is inherently costly for number of reasons: (Sommerville, 2001)

- Proposed changes have to be analyzed very carefully both from a business and a technical perspective. They must be approved by a range of people before being put into effect.
- Because sub-systems are never completely independent, changes to one sub-system may adversely affect the performance or behaviour of other sub-systems. Consequent changes to these sub-systems may therefore be needed.
- The reasons for original design decisions are often unrecorded. Those responsible for the system evolution have to work out why particular design decisions were made.
- As systems age, their structure typically becomes corrupted by change so the costs of making further changes increases.

As society becomes increasingly dependent on systems of various types, the amount of effort devoted to evolution rather than new system development is increasing. These existing systems that must be retained are now sometimes called *legacy systems*.

7.2 Problems and Solutions

Throughout this project, a number of obstacles were encountered which surfaced parallel with the development of the project. They would discuss to some extent below:

↓ The System Analysis Phase

Since there was no prior knowledge and experience in developing a system, it was rather difficult to determine the scope of the system. This was even more so where the time frame was supposed to be met. This was then overcome after analyzing and studying the capabilities of ASP technology in particular. It's like finding out the traveling speed, so that the distance that could be covered at the end of the given time could be estimated.

↓ The Designing Phase

The applying of theoretical information gathered in the previous phase into practical use is no doubt a major problem one can face. Due to the lack of experience in designing systems, it was difficult to apply the best solution in a design. Browsing through the Net and previous modules' work was seen to be very informative yet helpful and was somewhat like an unseen guide.

↓ The Implementation Phase

As mentioned prior to this, no earlier knowledge in ASP proved to be difficult when scrutinizing the codes of the respective modules. Most of the modules were written with differing styles and structure with missing indentation where the codes were concerned.

The importance given during programming sessions in the university regarding readability and structure was proven here. As most of the pages were re-coded

during the integration process, almost all the pages were modified to some extent. The main area that was stressed was layout and colour. The problems during database integration too were solved. This proved the Microsoft Access Server's credibility.

Technical problems that were not familiar with had to be solved through a number of ways. Among the many were, reading reference books, the aid from supervisor, friends and also through the Net.

↓ The Testing Phase

Unanticipated appearance of web pages in different web browsers. The appearance of web pages is different during the testing phase, such as different positioning of graphics, text, and tables on these problems couldn't be detected.

↓ Difficulties in Determining The Scope of The System

Because of PMCS are involves 4 modules: - *Registration, Scheduling, Patient History and Admission/Discharge*, so there are a lot of functionality needs to be perform. It is difficult for me to determine the scope of the project, so I study the functionality that existing in the module either manual or automatic ways. Also, the project supervisor, Dr. Diljit Singh has given me some valuable advices and opinion to outline the scope of the project during the initial stages.

↓ Understand on Current System Procedure

All the hospital departments have some procedure to follow when performing the daily tasks. It is about what you need to do before you can do the next things. For example, patient needs to register and get a patient no, every time he/she can to treatment he/she will assign on admission number, and whether are using the

admission number or the patient number to order lab test. All this details need to clarify before developing the system.

So, various references about the hospital departments and organization have been refers from the library or Internet. All this also helps to understand better on the hospital daily tasks and procedures.

+ **No Disclosure to ASP Language**

I have problems when using the ASP language because this is the first time I'm using the ASP language. Rather than that this language is not thought in the calss.

7.3   System Strengths

+ **User Password Validation**

Providing a user-password authentication system prevents unauthorized users from accessing pages that they do not have permission to view. More importantly, unauthorized users are prohibited from accessing records stored in the database.

+ **Friendly User Interface**

PMCS is developed based on GUI. A lot of useful icons are provided which gives users faster access. Besides that, it has very user friendly and consistent environment that is similar to other window application such as moving and closing a window.

PMCS is a window style interface that contains buttons, scroll bars, combo boxes, which makes the system reality easy for the nurses and the physician with little technical knowledge to learn and use. Besides, simple steps or actions like mouse-clicking and enter a simple text is sufficient to process user's requests.

↓ Easier Referencing

In Treatment history, the previous history done on each patient should appear on the screen for easier referencing while the physician proceeds with the treatment. This in turn eliminates the common problem of locating the patient's file.

↓ Transparency

The system on the whole is transparent. Therefore, users do not need to know the intricacies of it, for example where the database is not required. All they need to do is to type in their keywords and view the query results.

↓ Incorporates data validation

Data Validation is done prior to insert record into the database. All the fields in the form will be checked for null value or invalid data type. With this feature, error when inserting record into the database will not occur. Error message will also be prompted to the user if important field is not filled.

7.4 System Weakness

↓ Database Encryption

Data stored in Microsoft Access 2000 is not secured because it is stored in plain text format rather than in encryption format.

↓ Email Checking

The system was unable to validate an email address in terms of deliverability. It only checked for the inclusion of an @. a. and a com. Once it had met the above requirements, it was considered a valid address.

123

↓ Processing Time

As database grows bigger, data volume grows larger, and the processing time will increase, thus response time will increase. This is due to one of the limitations in Microsoft Access 2000.

↓ Internet Based System

This project is based on internet, so the user can only access if they have internet connection.

↓ Browser Dependent

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

↓ Language

Only English as presentation language in the system user interface, maybe it will cause problems for those users who are different medium from English as their communication language. It will cause them have difficulties in understanding this system.

7.5 Future Enhancements

No system is perfect and room for improvement exists always depending on the varying needs and expectations towards it. Among the ways of improving this particular system are:

↓ Saving Storage Space

As the database grows bigger, yearly housekeeping would be an essential feature in the future of PMCS. This is because the records of those patients who have not visited

the clinics/ hospitals for more than a year should be printed out and deleted from the system. This would help to prevent data corruption as bigger files tend to be more easily corrupted. In addition, it helps to save the storage space and thus, increase the performance of the system.

- **Medical Report Format**

System should be able to generate standard medical report for user.

- **Database Upgrading**

To provide higher performance and shorter processing time, Microsoft Access 2000 may be upsized to Microsoft SQL Server 7.0.

- **Support Other Browsers**

As stated, the system requires Microsoft Internet Explorer 4.0 and above for execution. In future, it can be tuned to fulfill other browser requirements such as Netscape for execution. This is because Netscape has a sizeable share in the browser market besides IE.

- **Widen the Scope of PMCS**

Put-in more healthcare issues to make it more comprehensive and complete.

7.6 Knowledge and Experience Gained

- **The important of all the phases in SDLC**

System analysis is an important phase in the system development life cycle (SDLC). This phase also capture user requirement and the main goal of the system. If this phase is wrong defined, it will cause faulty to the system development and later progress. With a

complete and though system analysis, the system that is developed will fulfill all the requirements and achieve its goal.

Learn new software

During the implementation of this project I able to learned new software such as Dreamweaver UltraDev 4, Adobe Photoshop 6.0 and not to forget *Active Server Pages* (ASP).

Time management

It is a good opportunity to brush-up our time management because the project is required to be completed within a limited time frame. Time allocation for different task to completed needs to be well arranged and managed because the development of PMCS was scheduled.

Task break-down management

In order to develop PMCS successfully, it needs to break down into smaller sub tasks and than complete one sub-task by another. The challenge is how to distribute every single sub-task with respect to the time duration given to complete it. Finally, to merge up every sub-tasks of each stage before go beyond another stage of the system development life cycle.

7.7   Chapter Summary

In the finishing of the system, the assess phase and the organizing phase is very important to make sure the produced system is more quality.

In this particular phase there are a few of advantages and disadvantages which are notified. The advantages that are notified are kept while the disadvantages are tried to modify. A few opinions are brought up so that it can overcome the constraints that exist during the operations of the system.

Problems that arise are being notified and redone with better ways so that the same problem would not be repeated in future. A good and a system that have been expected can be produced. Finally, several suggestions are made to enhance the PMCS for better performance in the future, which focuses on solving the system limitation and system weakness, and some other capabilities.

Conclusion

In conclusion, PMCS is indeed a feasible solution for the management of a clinic, because of the reasons listed below:

1. There are many benefits that could be derived from PMCS (these benefits have been discussed earlier in section 1.3)

2. Doctors of new generation are computer literate. For example, the syllabus for the medical students of University Malaya includes an elementary course on the data processing using computers. Therefore, there should be no problem of learning how to use the system.

3. The availability of sophisticated hardware at a much reduced cost will enable more clinics to computerize its management system. Also, it would be affordable to use more powerful machines so as to increase its performance and productivity.

However, computer utility in the medical field should be considered as a contemporary effort and not as a competitive alternative to medical professionals. Also, such system should be considered as aids rather than substitute to doctors.

Besides that, the development of PMCS has also been most beneficial to me. I've successfully develop a comprehensive and feasible system within the given 7 month from June, 2002 to February 2003. During the development, I was able to apply the knowledge and theory learnt during lectures, especially from courses on database management, system analysis and design.

Conclusion

In conclusion, PMCS is indeed a feasible solution for the management of a clinic, because of the reasons listed below:

1. There are many benefits that could be derived from PMCS (these benefits have been discussed earlier in section 1.3)

2. Doctors of new generation are computer literate. For example, the syllabus for the medical students of University Malaya includes an elementary course on the date processing using computers. Therefore, there should be no problem of learning how to use the system.

3. The availability of sophisticated hardware at a much reduced cost will enable more clinics and hospitals to be able to afford the system. Also, it would be affordable to use more powerful machines so as to increase its performance and productivity.

However, computer utility in the medical field should be considered as a contemporary effort and not as a competitive alternative to medical professionals. Also, such system should be considered as aids rather than substitute to doctors.

Besides that, the development of PMCS has also been most beneficial to me. I've successfully develop a comprehensive and feasible system within the given 7 month from June, 2002 to February 2003. During the development, I was able to apply the knowledge and theory learnt during lectures, especially from courses on database management, system analysis and design.

Furthermore, I've acquired the knowledge of a new programming language using the object- oriented approach. In addition, I've also learned to be more independent, resourceful and confident. Thee values are very important in making us a better person.

Finally, there is still much room for improvement in this health information system (Penang Medical Centre System). As mentioned before, the system scope was defined through waterfall model with prototype; this makes the system most expandable in terms of functionality. Enhancement could still made to the system with more features added.

130

References

Books

1. Gary P. Schneider and James T. Perry, "Electronic Commerce", Second Edition, Course Technology, Canada, 2001

2. Ian Sommerville, "Software Engineering", Sixth Edition, Addition-Wesley, England, 2001

3. Kenneth E. Kendall and Julie E. Kendall, "System Analysis and Design", Prentice Hall, New Jersey, 1999

4. Myers. G, "The Art of Software", Wiley, 1997.

5. R.S. Pressman, Software Engineering A Practitioner's Approach, New York, McGraw Hill, 1992

6. Roger S. Pressman, "Software Engineering A Practitioner's Approach", McGraw-Hill, Inc, 1997

7. Sellapan. P, "Database Management Theory and Practice", Sejana Publishing, Petaling Jaya, 1998

8. Shari Lawrence Pfleeger, "Software Engineering Theory and Practice", Second Edition, Prentice-Hall Inc, New Jersey, 2001

9. West, Ray., Muck, Tom., dan Allen, Tom., "Dreamweaver Ultradev 4: The Complete Reference", Berkeley, CA: Osborne/McGraw-Hill, (2001).

Website URL

1. http://www.bayietehealth.com

2. http://www.singaporeplay.com.link.htm

Appendix A

References

Books

1. Gary P. Schneider and James T. Perry, "Electronic Commerce", Second Edition, Course Technology, Canada, 2001

2. Ian Sommerville, "Software Engineering", Sixth Edition, Addition-Wesley, England, 2001

3. Kenneth E. Kendall and Julie E. Kendall, "System Analysis and Design", Prentice Hall, New Jersey, 1999

4. Myers. G, "The Art of Software", Wiley, 1997.

5. R.S. Pressman, "Software Engineering: A Practitioner's Approach, New York, McGraw Hill, 1992.

6. Roger S. Pressman, "Software Engineering A Practitioner's Approach", McGraw-Hill. Inc, 1997

7. Sellapan. P, "Database Management: Theory and Practice", Sejana Publishing, Petaling Jaya, 1998

8. Shari Lawrence Pfleeger, "Software Engineering Theory and Practice", Second Edition, Prentice-Hall Inc, New Jersey, 2001

9. West, Ray., Muck, Tom., dan Allen, Tom., "Dreamweaver Ultradev 4: The Complete Reference", Berkeley, CA: Osborne/McGraw-Hill, (2001).

Website URL

1. http://www.baylorhealth.com

2. http://www.capefearvalley.com.link.htm

3.   http://www.hcds.com

4.   http://www.melaka.gov.my

5.   http://www.mieur.nl

6.   http://www.sparrow.org.com

7.   http://www.tridenthealthsystem.com

8.   http://www.ummc.edu.my/ummc

9.   http://www.fatimah.com.my

10. http://www.selayanghospital.gov.my

Table of Contents

Table of Content

1.0 Introduction

A user manual is a reference guide or tutorial for system users. The manual should be complete and understandable, so sometimes it presents the system to users in layers, beginning with the general purpose and progressing to detailed functional descriptions. First, the manual describes its purpose and refers to other system documents or files that may have more detailed information. This preliminary information is especially helpful in reassuring users that the document contains the type of information they seek. Special terms, abbreviations, or acronyms used in the manual are included for easy reference.

Before you access to the system, make sure you meet the minimum requirements as follow:

1. Window XP Professional

2. Personal Computer with at least 256 MB

3. Internet Connection

1.1 Access Method to Penang Medical Centre System (PMCS)

Because of this system is uploaded to intranet so, this system can be access by using the address below:

<center>http://localhost/pmc/</center>

The first page of this system is shown in the Figure 1.1:
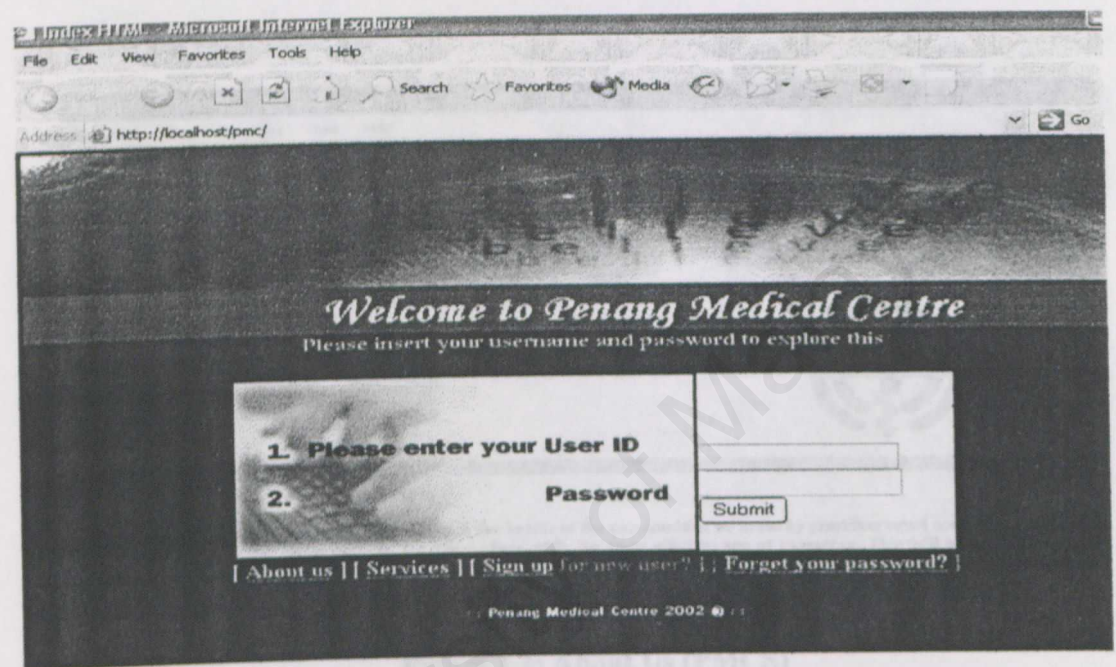
Figure 1.1 First page of PMCS

This manual is divided into 4 parts, which are User, Admin, Doctor and Staff Edition.

1.2 User Edition

1.2.1 Login Module

PMCS is divided into two login module, which is for login user and non-login user.

1.2.1.1 Non-Login User

Non-login user can view the following pages:

- About us
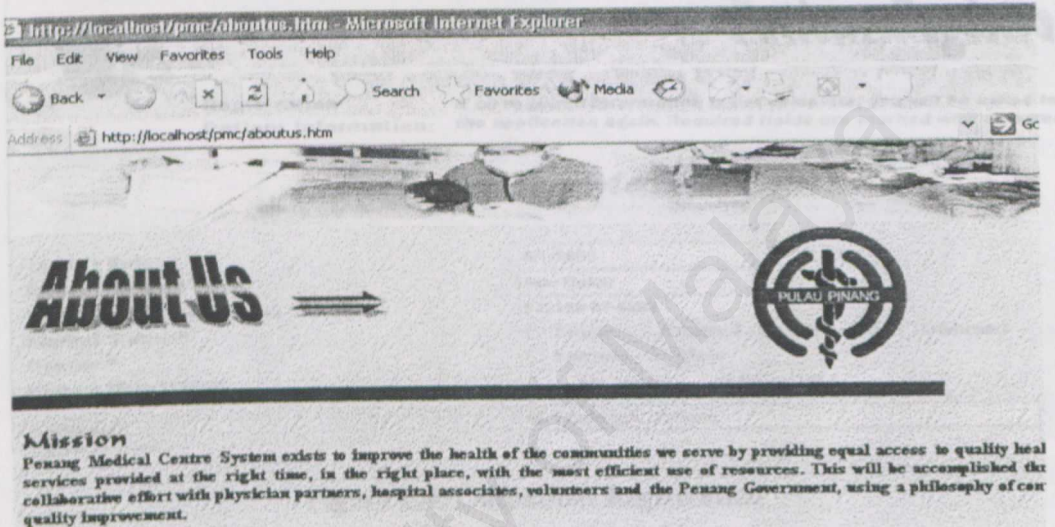
Figure 1.2: About Us (PMCS)

- Services
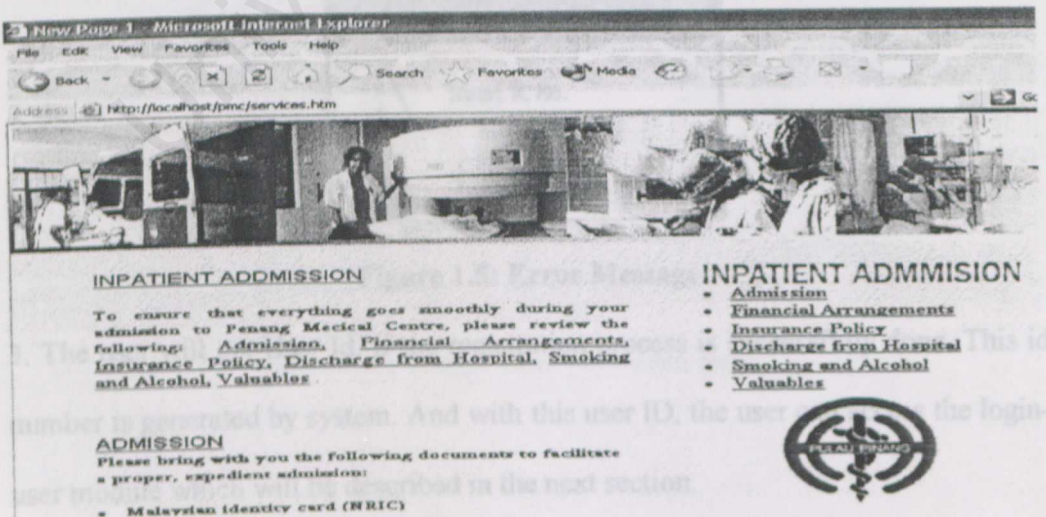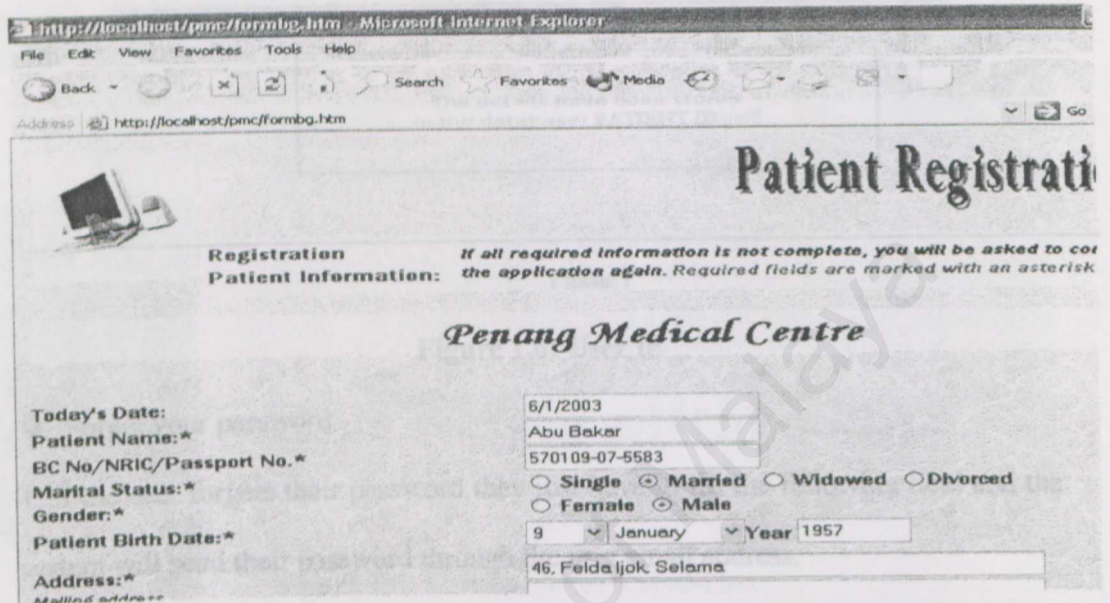
Figure 1.3: Inpatient Services

◆ Sign up for new user?

1. In this page, user can sign up for only outpatient treatment. So, before they can use this system they have to sign up for new user if they never register before in PMCS.

Figure 1.4: Outpatient Registration

2. If the user failed to register particular fields, error message will be pop up as shown below:
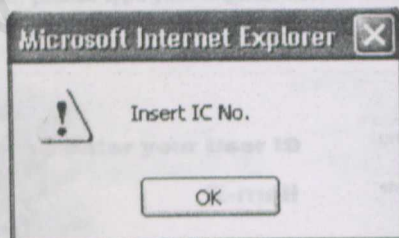
Figure 1.5: Error Message

3. The user will get User Id, if the registration process is successfully done. This id number is generated by system. And with this user ID, the user can access the login-user module which will be described in the next section.
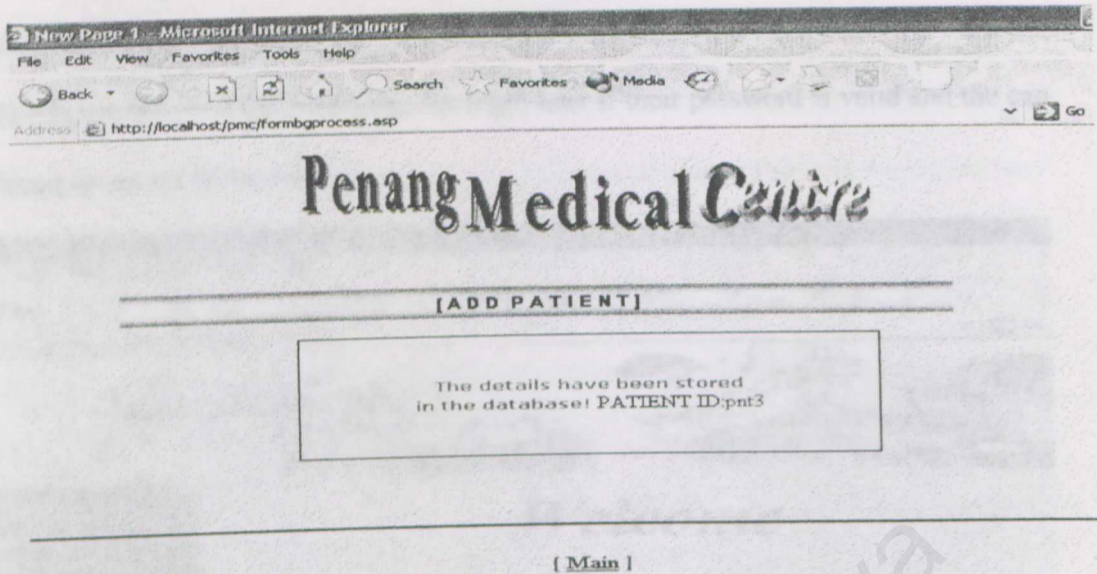
Figure 1.6: User Id

● Forget your password

1. If the user forgets their password they just have to fill the following field and the

system will send their password through the user email address.

Figure 1.7 Forget your password page

1.2.1.2 Login-User Module

1. The figure below is the homepage for login-user if their password is valid and the can continue to access this system.
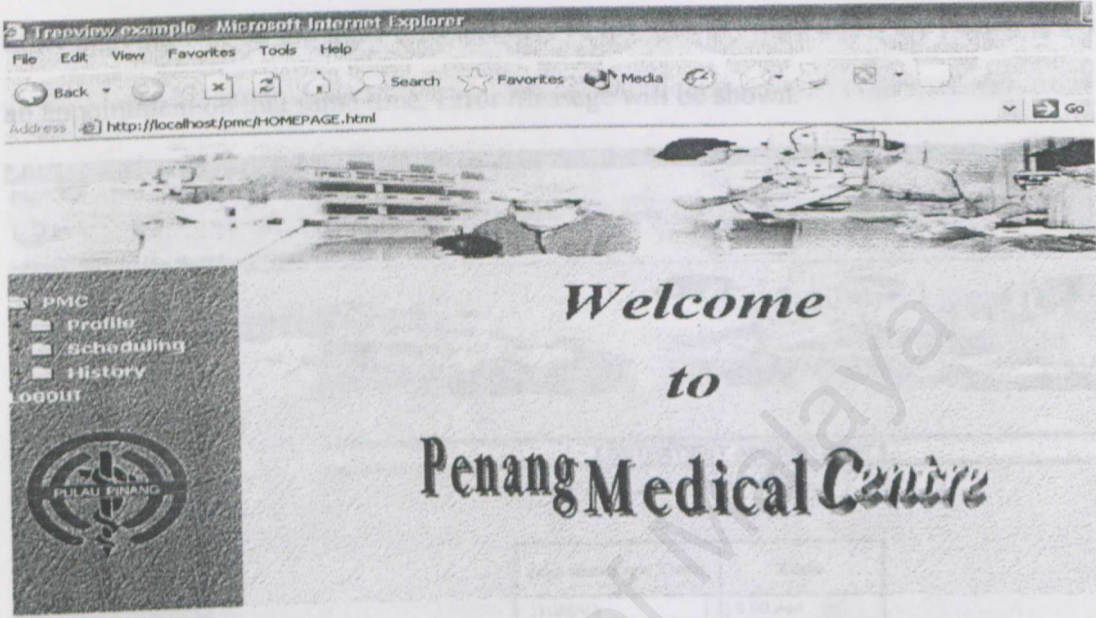
Figure 1.8: Homepage for Login-User

2. The login can view their profile and if they want to edit anything to their profile they just have to click **Edit Profile** and make the changes and **Submit** the details.

Profile

Patient Name:	Hisham
BC No/NRIC/Passport No.	570109-07-5583
Marital Status:	Married
Gender:	Female
Patient Birth Date:	1/9/1957
Address:	46, Felda Ijok, Selama ,
State:	Perak
Email Address:	sham@google.com
Home Telephone:	016-5932135
Emergency Contact:	05-5412364
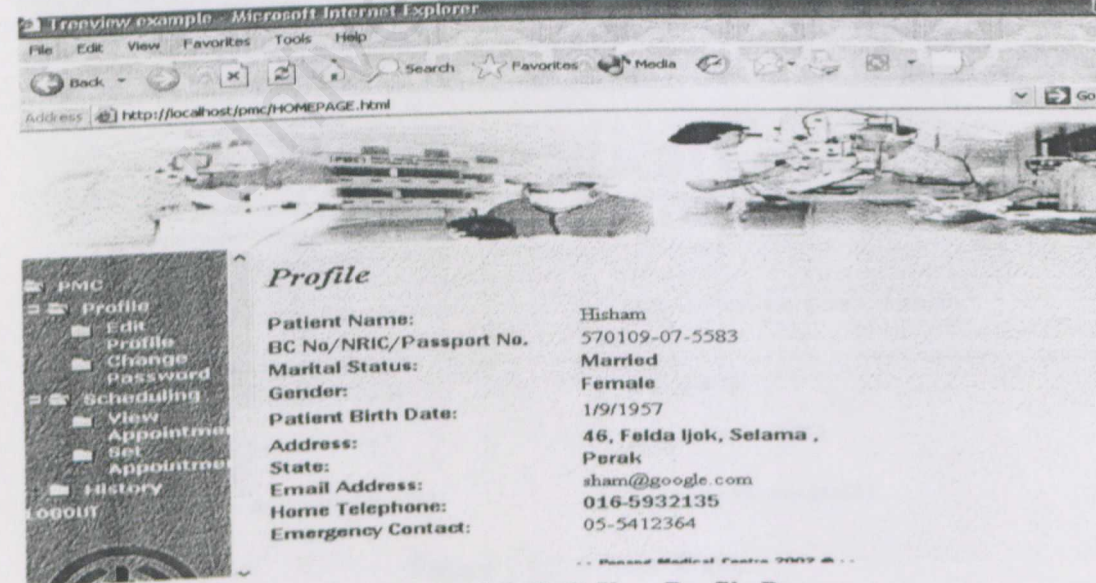
Figure 1.9: Edit User Profile Page

Besides, the user also can change their password. They just have to click **Change Password** and key in their new password and save it.

3. The user can set an appointment with the doctor available in PMC. If the doctor have other appointment on the same time and date so the system won't allow the patient to set an appointment on that same time. Error Message will be shown.
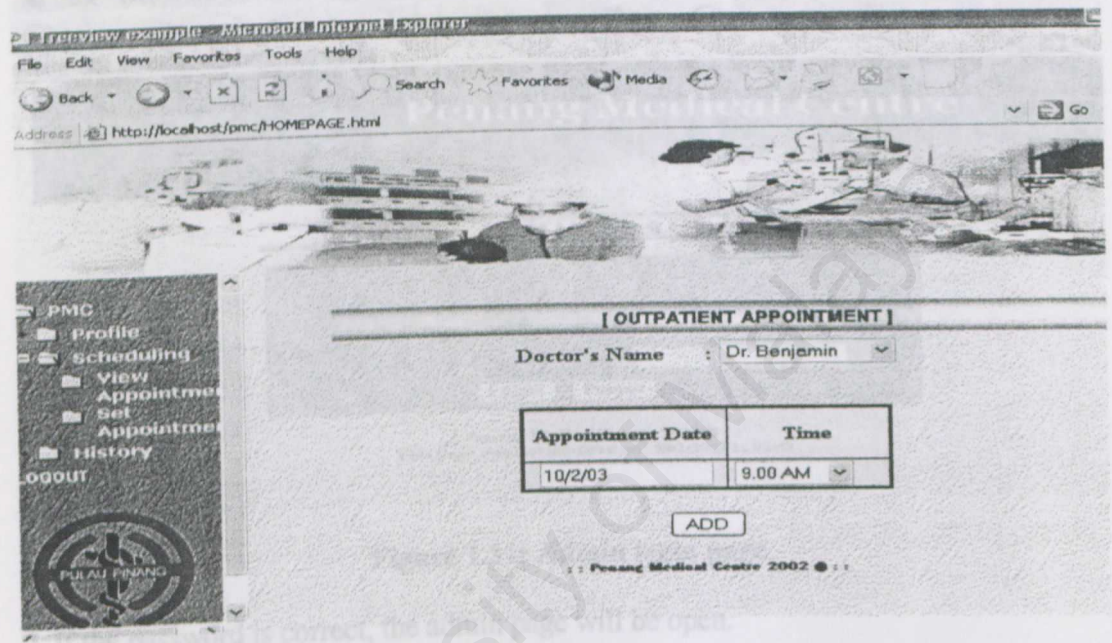
Figure 1.10: Set Appointment
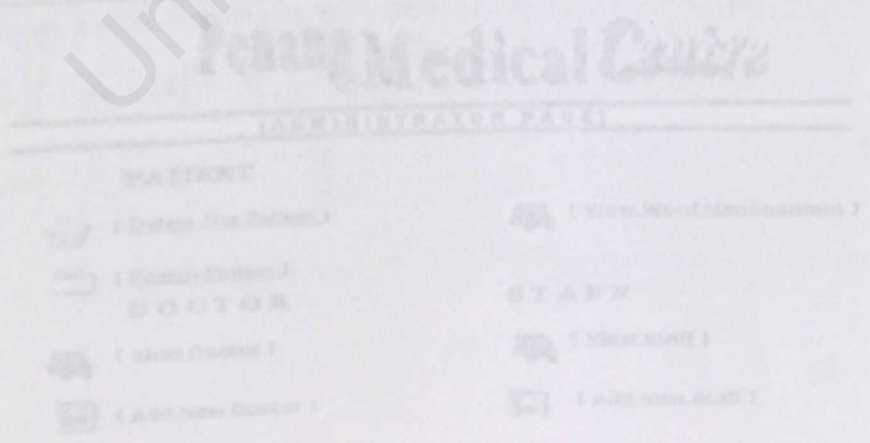
Figure 1.12: Admin Main

1.3 Admin Edition

1. This page is confidential and it cannot access by other users. Admin able to access this site by using the address below:

http://localhost/pmc/adminlogin.htm
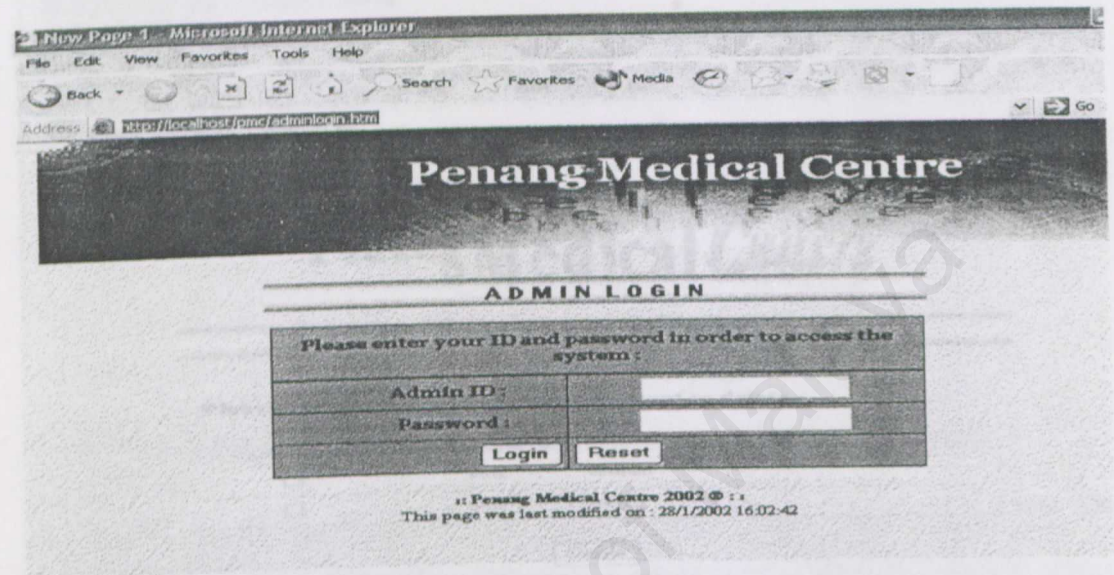
Figure 1.11: Admin login page

2. If the password is correct, the admin page will be open.
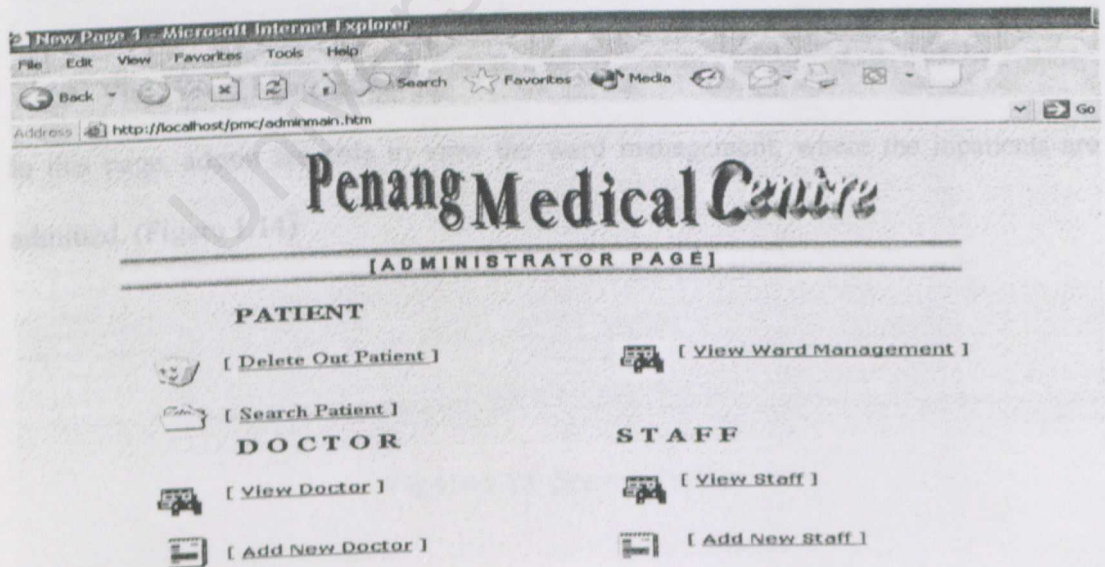
Figure 1.12: Admin Main

3. Admin can modify the details in Patient records, Doctor and Staff.

1.3.1 Patient

- Delete Out Patient

In this page, admin are able to delete the outpatient records.
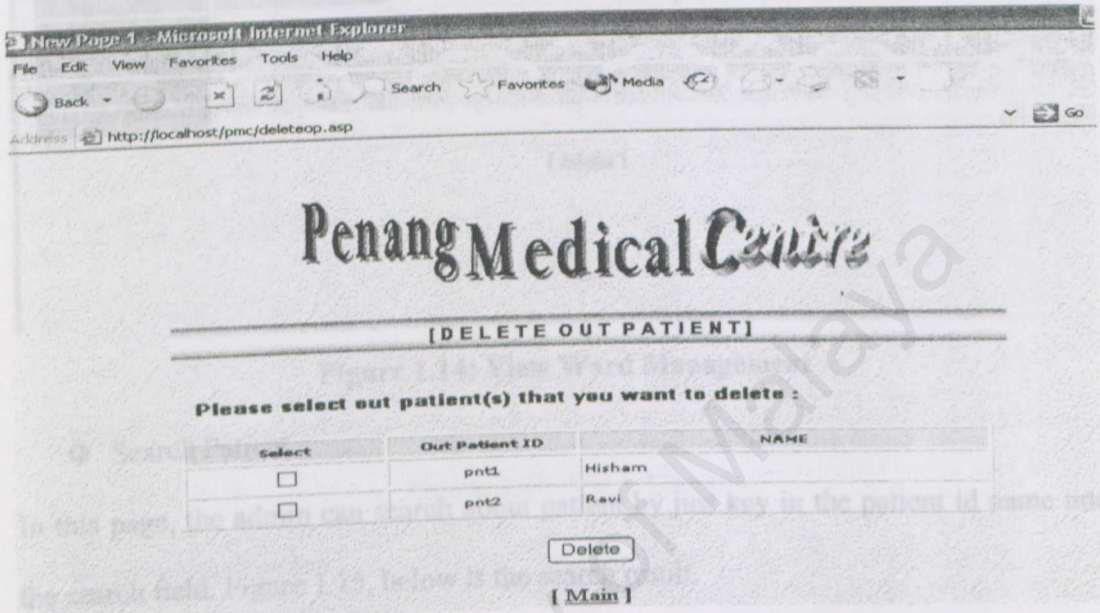
Figure 1.13: Delete Out Patient

- View Ward Management

In this page, admin are able to view the ward management, where the inpatients are

admitted. (Figure 1.14)

Figure 1.14: View Ward Management

◉ Search Patient

In this page, the admin can search about patient by just key in the patient id name into

the search field. Figure 1.15, below is the search result.
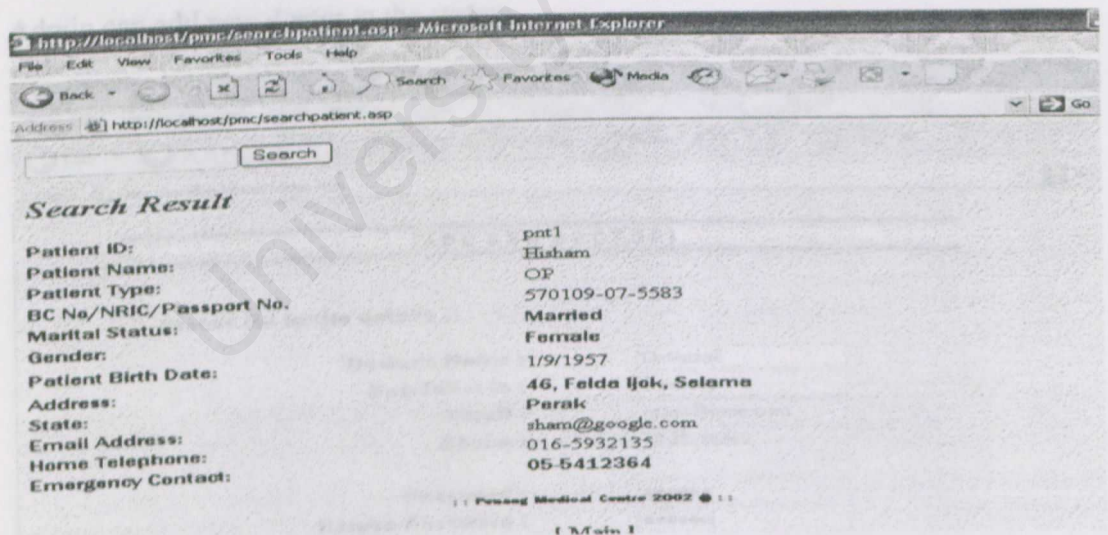
Figure 1.15: Search Patient

1.3.2 Doctor

● View Doctor

Admin are able to view the list of doctor available in the system.
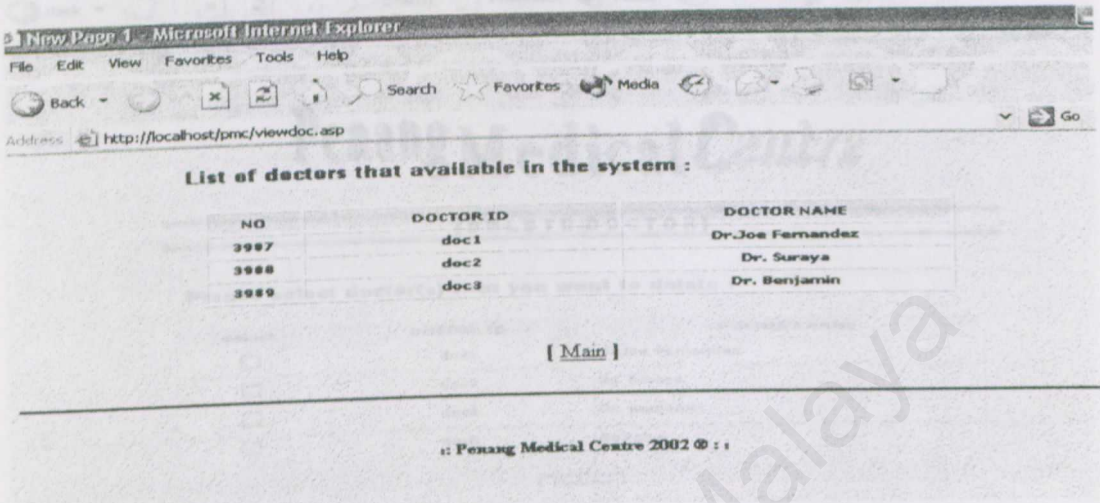
Figure 1.16: View Doctor

● Add New Doctor

Admin can add new doctor to the system.
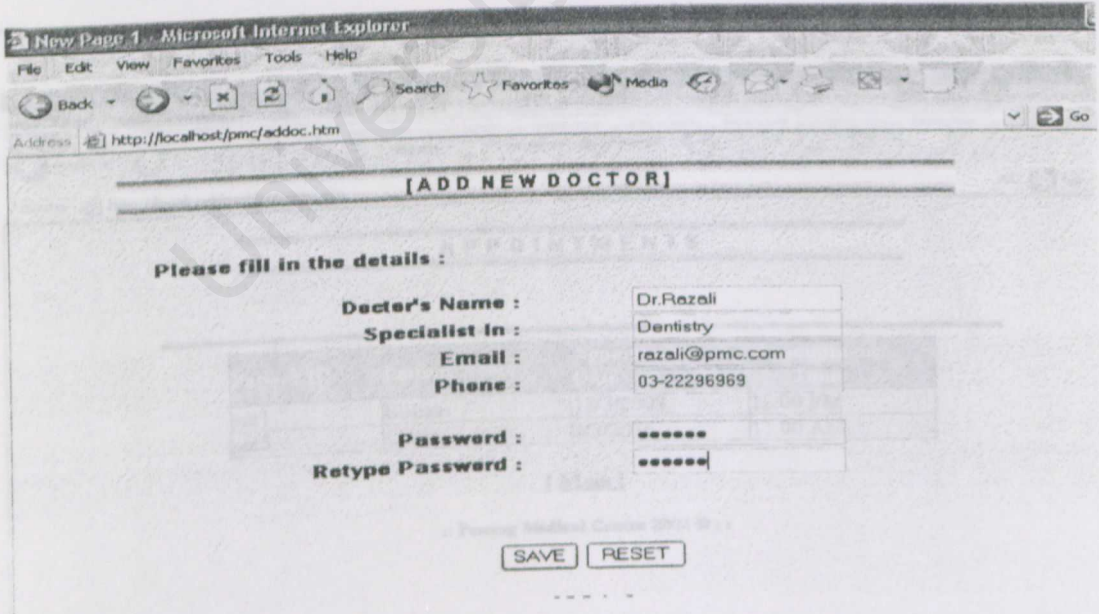
Figure 1.17: Add New Doctor

● Delete Doctor

Admin can delete doctor from the system.

Figure 1.18: Delete Doctor

1.4 Doctor Edition

● View Appointment

The doctor can view patient appointment that is under his name.

Figure 1.19: View Appointments

○ Treatment

Doctor has to key in the treatment report for each patient that treated.

Patient ID	Date Treated	Remarks
pnt5	7/1/2003	sick and cold

SAVE

[Main]

Figure 1.20: Treatment Page

1.5.1 Nurse

● Patient Personal Info

The nurse has to key in the patient personal info according to medical check to each

patient.

1.5 Staff Edition

- Staff Login

Different type of staff can access to this page.
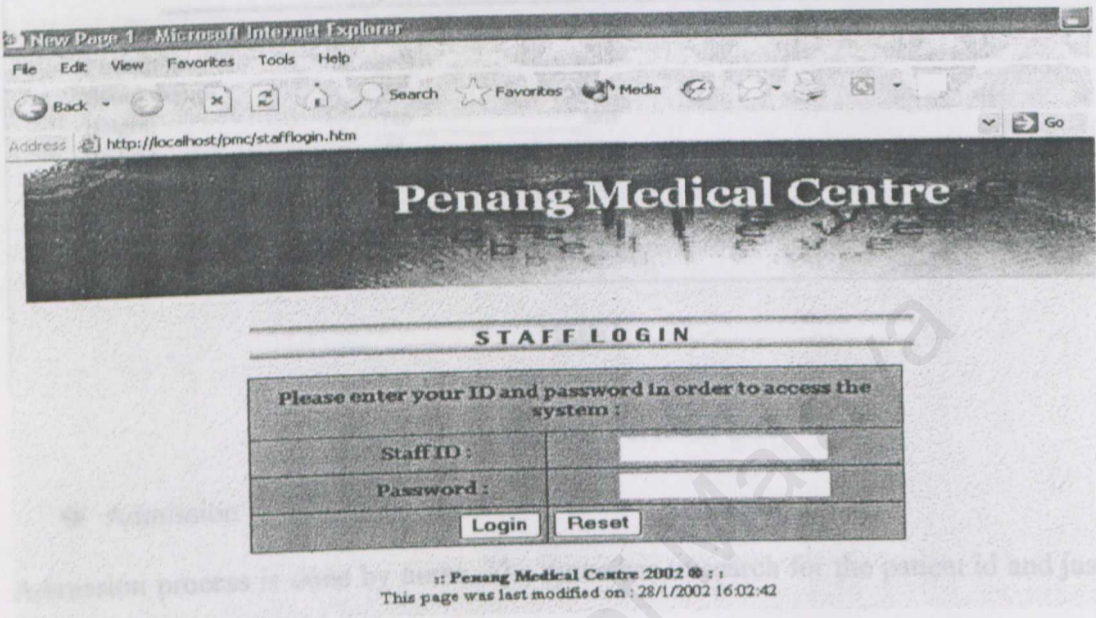
Figure 1.21: Staff Login

1.5.1 Nurse

- Patient Personal Info

The nurse has to key in the patient personal info according to medical check to each

patient.

Figure 1.23: Admission

Figure 1.22: Patient Personal Info

◉ Admission

Admission process is done by nurse. The nurse has to search for the patient id and just

save to admit them in the ward. The system will provide the bed number and the ward

number automatically.

Figure 1.23: Admission

● Discharge

Nurse has to search for the patient id and just click discharge, the system will remove the

patient from system.

Figure 1.24: Discharge

1.5.2 Pharmacist

● Prescription

Pharmacist has to key in the prescription according to the doctor's treatment.

Figure 1.25: Prescription

1.5.3 Lab Assistant

○ Lab Test

Lab assistant has to key in the lab results in this page.

Figure 1.26: Lab Test

1.6 Logout Page

The logout page for this system is shown below:

Figure 1.27: Logout Page