WXET3182 CLINIC CARE INFORMATION SYSTEM (CCS)

LEE MEI KOON
WET 000 016

Supervisor: **Dr. Rosli Salleh**
Moderator: **Mr. Noorzaily Mohamed Noor**

Abstract

The purpose of this study is to develop a clinic management system used for clinic in Malaysia. A main objective is to provide services including manage patient records, manage medicine stocks and produce management reports. System will reduce processing steps, lessen the use of papers and help staff to search the information with less time. The system is developed on a window platform. A Microsoft Access database management system is used. Visual Basic 6.0 is used as a software tool for development. Two-tiered client server architecture is used to support the Clinic Care Information System (CCS). Waterfall model of the Software Development Life Cycle is employed in the development of this system.

Acknowledgement

I am deeply indebted to my project supervisor, Dr. Rosli Salleh for helping and guiding me conduct this project from its earliest stage to the final stage of the project. Meanwhile, I would like to thank my project moderator, Mr. Noorzaily Mohamed Noor for taking his time to attend my VIVA session and his feedback for the project.

Many thanks to "Klinik Pelajar Universiti Malaya" executives who provide the useful information that I needed. I would like to thank every one of my fellow friends and course mates for their helpful and cooperating in providing with useful information.

In particular, I would like to thank my family for their patience and support in this project. I can only hope the finished project meets their high expectations.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 PROJECT DEFINITION

Clinic Care Information System (CCS) is a healthcare solution tool to create a better health service for the clinic in Malaysia. It allows a more efficient and effective control over the clinic's resources, time management and catering to a patient's requirement.

CCS utilizes the latest technology to simplify the daily operation of a clinic. The system allows users to access all demographics, patient progress, medication, medical center's records and pharmaceutical order in systematic way.

CCS is an integrated system in a LAN environment with Client Server architecture. It is developed in Microsoft Windows environment. Visual Basic 6.0 is used as a software development tool. It provides friendly graphical user interface (GUI) with ease-to-learn, and flexibility to the users. The users will be able to get information more quickly from database in a paperless environment.

These potential users are the doctor, pharmacist, and nurse or staff. Password control is needed in order to access to the system in their respective scope. The overall result will be more efficient administration to support the health care being offered and quality of service will be more readily available.

1.2 PROJECT OBJECTIVES

The overall objectives of the Clinic Care Information System are as following:

- To eliminate user manual work.
- To save time and reduce dependence on clinician.
- To design a consolidated operations for Malaysia clinics.
- To enhance the available clinic system in Malaysia.
- To create an effective and user friendly interface for the users.

1.3 SCOPE AND LIMITATION

1.3.1 The Project Scopes

Clinic Care Information System is developed specially for the clinic in Malaysia. It converts almost of the manual operations into computerized methods.

The project will cover the following features:

- Develop the computerized system for registering patients.
- Develop the computerized system for recording all related data.
- Develop a database system to store data in more secure way.
- Develop an automatically updated function for drug inventory.

1.3.2 The Project Limitations

Clinic Care Information System is limited as followings features:

- This system is not connected to Internet. Thereby, it can only access by authorized medical personnel in a local-area-network (LAN).
- Appointment can only be made through telephone or face-to-face. The patients can't directly make appointment using the system itself.
- No panel doctor function for companies.

1.4 OVERVIEW OF THE SYSTEM

To attain the most added values to users, the system is focused on the main types on clinical processes. Each process will reduce the workload of a clinic and hence increase the performance of the clinical management.

The key processes of the Clinic Care Information System are as followings:

- **Centralized Patient Information Storage**

  Patient's information will be stored in paperless and organized manner. Retrieval and management of all patient's information will no longer be a difficult task.

- **Appointment Tracking**

  No more flipping through the appointment book to keep track whether patients kept their appointments, rescheduled or missed their appointments. CCS keeps track of everything for the system's users.

- **Control of Drug Inventory**

  Keeping a tight watch over the clinic's drug is important. Expiration date of the drug must be closely supervised, as no clinic will ever want to overlook and issue their patients with expired drugs.

- **Print**

  The simple printing process allows for all medical reports to be easily printed as a hard copy. This may include printing of the medical certificate, patient medical report and others.

- **No Hassle of Training**

No training is needed to learn how CCS works. CCS is extremely ease to use in which all new staff will be able to pick up in short time.

1.5 EXPECTED OUTCOME

Clinic Care Information System (CCS) is an integrated system that supports all operational areas within a clinic. It is expected to provide the following features:

- **User friendly.**

  The CCS should be user friendly in order to ease the user navigation and also control. The interface design should help users to identify the main function of the system. Besides, short cut keywords of the system should be clear to make the program more users friendly.

- **Security.**

  The CCS should be able to identify the users and the strangers. Username and password are required from the user to enter the system. It is needed to prevent unauthorized users from accessing the system for their personal advantages.

- **Attractive.**

  The system should be developing in such a way that attracts user attention while using it. Additional features and interesting interfaces are the main keys to make the system attractive and interesting.

- **Faster data transfer rate.**

  The CCS will have faster data transfer rate. With the faster data transfer rate, file can be sent to another user in shorter time, and also the respond time could be reduced.

- **Availability.**

The CCS is designed in the sense that it is easy to maintain and enhance in the future. Additional module can be easily implemented to the original version.

1.6 PROJECT SCHEDULE

For successful project, a project schedule for the whole development's activities is planned out. The project is broken down into two phases; the first phase involves project planning research, analysis and design. The coming semester is going to involve the second phase, which is the testing and implementation of the system.

To achieve a systematic progress and ensure on-time delivery of the system, Gantt Chart is designed. Figure 1.1 shows the duration scheduled for the activities of Clinic Care Information System (CCS).

Clinic Care Information System (CCS) Schedule from June 2002 to September 2002

ID	Task Name	Start	Finish
1	**Introduction**	**Mon 6/10/02**	**Thu 6/27/02**
2	Resourse Searching	Mon 6/10/02	Thu 6/27/02
3	Understanding CCIS	Mon 6/10/02	Tue 6/18/02
4	Distribution Scope/Module	Mon 6/17/02	Tue 6/25/02
5	**Literacture Review**	**Fri 6/21/02**	**Mon 7/22/02**
6	Existing System Review	Fri 6/21/02	Mon 7/1/02
7	Study Project Technologies	Tue 6/25/02	Mon 7/22/02
8	Analysis Previous Thesis	Tue 6/25/02	Mon 7/22/02
9	**Methodology**	**Mon 7/22/02**	**Tue 7/30/02**
10	Waterfall Model review	Mon 7/22/02	Tue 7/30/02
11	**System Analysis**	**Mon 7/29/02**	**Thu 8/15/02**
12	System Analysis	Mon 7/29/02	Thu 8/15/02
13	Analysis Requirement	Mon 7/29/02	Thu 8/15/02
14	Result Analysis	Mon 8/5/02	Wed 8/14/02
15	Tools Analysis	Mon 8/5/02	Fri 8/9/02
16	**System Design**	**Thu 8/15/02**	**Tue 9/3/02**
17	Proccess Design	Thu 8/15/02	Fri 8/23/02
18	Interface Design	Mon 8/19/02	Tue 8/27/02
19	Database Design	Mon 8/26/02	Tue 9/3/02
20	**SYSTEM IMPLEMENTATION**	Thu 10/10/02	Tue 11/12/02
21	**SYSTEM TESTING**	Mon 11/11/02	Mon 12/2/02
22	**SYSTEM EVALUATION**	Tue 12/10/02	Tue 12/24/02
23	**REPORT WRITING**	Thu 12/26/02	Thu 1/2/03

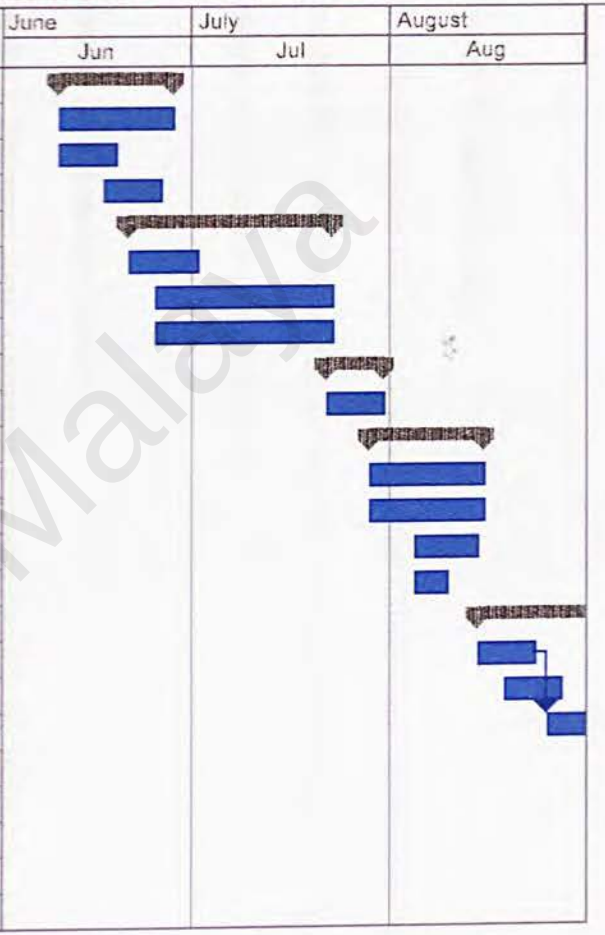


Figure 1-1: Project Schedule of Clinic Care Information System.

1.7 CHAPTER SUMMARY

This report is covered 8 chapters and organized as the following manner:

Chapter 1: Introduction

This chapter shows the overall view of the Clinic Care Information System (CCS). It describes the objectives of the system, the scope and limitation of the project, project outcome and its development schedule.

Chapter 2: Literature Review

Chapter 2 discusses some computer concepts, which are useful for basic understanding before applying them to the system. It reviews on the project study, features, capabilities, system architecture and others.

Chapter 3: Methodology

This chapter discusses methods and techniques that are used to solve the project problems. It shows how alternative approaches attempt to address the problems.

Chapter 4: System Analysis

This chapter fairly discusses the system analysis includes current system review, functional and non-functional requirements, hardware and software requirements.

Chapter 5: System Design

Chapter 5 ends with a brief description of the design stage of CCS. It describes the new system interface design, process components and system modules.

Chapter 6: System Implementation

System Implementation transforms the idea from the system design to program coding. It describes the tools for creating the database for this CCS and also programming lanaguages.

Chapter 7: Sytem Testing

This is the stage of the testing process where it has been done step by step from a single unit to a much more complicated sub module, before going into the whole CCS program.

Chapter 8 : System Evaluation

Syetem evaluation has to done before the full system can be delivered to the system user. Throughtout the system development life cycle, all the member involved in this project will work together to evaluate the system in order to give feedback for the eventual improvement.

2. LITERATURE REVIEW

2.1 THE IMPORTANCE OF LITERATURE REVIEW

To bring Clinic Care Information System (CCS) into existence, many researches and studies have been undertaken. A literature review gives developers an intuitive understanding of the kind of information required to carry out the project.

Another purpose of a literature review is to grants the knowledge in the existing system. Most importantly, it helps to indicate the strengths and weaknesses of several development tools. This can help the developer to evaluate the tools before choose the right tool to develop the system.

2.2 WHAT IS A COMPUTER-BASED INFORMATION SYSTEM

A computer-based information system (CBIS) is composed of hardware, software, databases, telecommunications, people, and procedures that are configured to collect, manipulate, store and process data into information.

The components are illustrated in Figure 2-1.

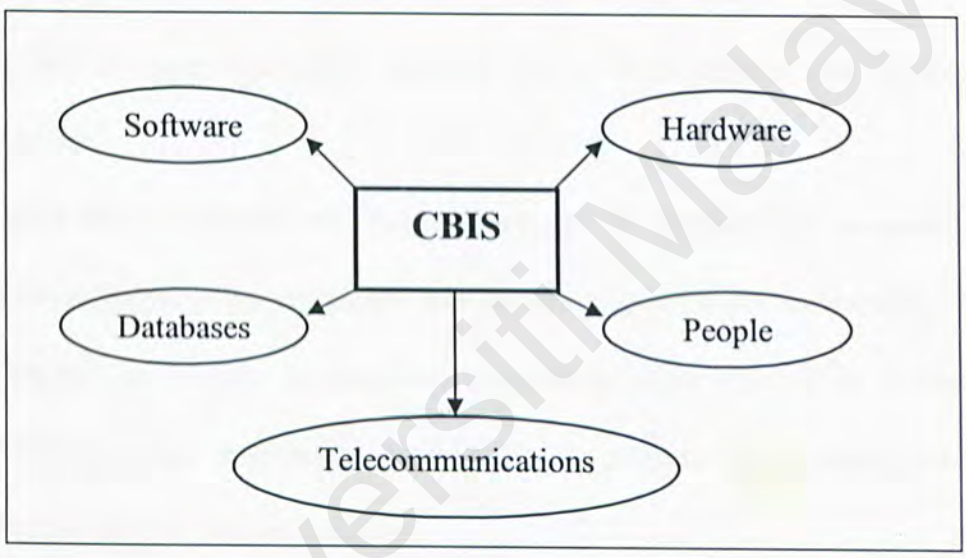


Figure 2-1: The Components of a Computer-Based Information system.

2.3 CURRENT SYSTEM REVIEW

2.3.1 Non-Computerized Medical Center System

The general medical center's workflow of non-computerized system is started with registration patients using their identity card or their name to paper files. And duplication of patient's file is happened since an unsystematic of the manual system. Thereafter, patient will be put in queue before a doctor is assigned to the patient.

The staff has to search the patient medical record from cabinet and bring it to the assigned doctor.

The assigned doctor will then read the patient's details, proceed with consultation, note the symptoms, identify the diagnosis and an prescription with handwriting. And the prescription will be transfer to pharmacy department either by staff or by the patient. Once the pharmacy has received the prescription, the pharmacist will check the available drug before issues it to patient manually.

After dispensing the medication, the nurse or staff will type and write out the receipt and, if necessary, medical leave for the patient.

As conclusion, the manual system is extremely slow down the clinical work and unfit for the staff as well as incredibly stressful.

2.3.2 Computerized Medical Center System

Due to systems based on computers are increasingly being used widely and inefficiency operation using manual system; many medical centers are changing their traditional system to computerized system.

The general medical center's workflow of computerized system is started with registration patients using their identity card or their name to computer system. And duplication of registration is not allowed. Thereafter, a doctor will be assigned to the patient in queue.

The assigned doctor will then retrieve the patient's details from the system, proceed with consultation, note the symptoms, identify the diagnosis and keys-in an prescription. Once the pharmacy has received the prescription, it will immediately dispense the medication to the patient.

After dispensing the medication, the nurse or staff will print out the receipt and, if necessary, medical leave for the patient.

2.3.3 Existing System Review

Today, clinic computerized system is not a new approach anymore. There are many excellent clinic systems have been developed to replace the manual system.

2.3.3.1 HealthCare Information Management System (HIMS)

HealthCare Information Management System (HIMS) is an online clinic system. It is designed using the Relational Data Model, which allows for query and report generation by the end user. HIMS uses the Client Server Architecture and is suited to Windows NT or 95 Server.

HIMS has the windows95 graphical user Interface in the front end and is written in Microsoft Visual Basic 5.0. It utilizes Microsoft Access 97 Database as a Database Management System.

HIMS includes some features such as:

- Designed for speed of data entry and inquiry – minimize clicks and keystrokes.
- A sophisticated Medical Records Management including integration with smart cards.
- A daily report of anticipated revenues.
- Procedure analysis based on time units.
- Aged patient accounts receivable by financial class.
- Aged delinquent accounts report by department and financial class.
- Aging accounts receivable by insurance carriers.

- Cash analysis.
- Diagnosis history.
- Practice financial revenues daily, monthly, and yearly.
- Physician productivity reporting.
- Hospital census.
- Online Medical Records Repository – allowing for both statistical and complex reporting and others.

The following shows some pages of the screen shot of the HIMS system user interfaces (Figure 2-2 to 2-3). Figure 2-2 shows the Appointment Layout of HIMS. It allows nurse to schedule any appointment with effortless.

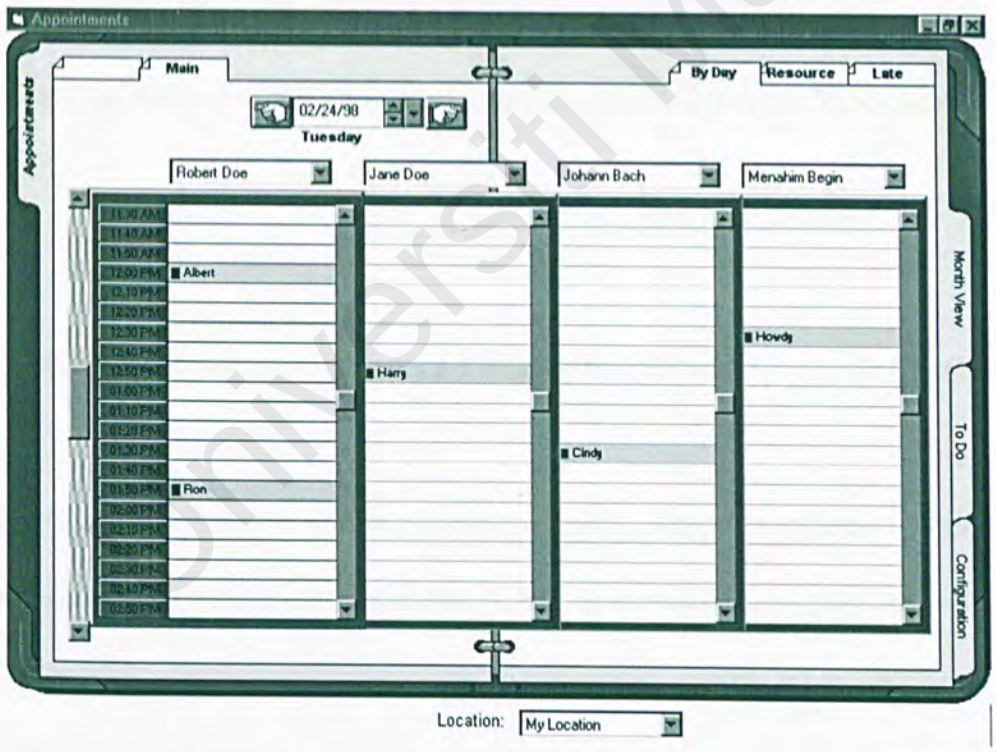


Figure 2-2: The Appointment Interface of HIMS.

Figure 2-3 shows the Doctor's Appointment layout of HIMS. The doctors have their own time table.

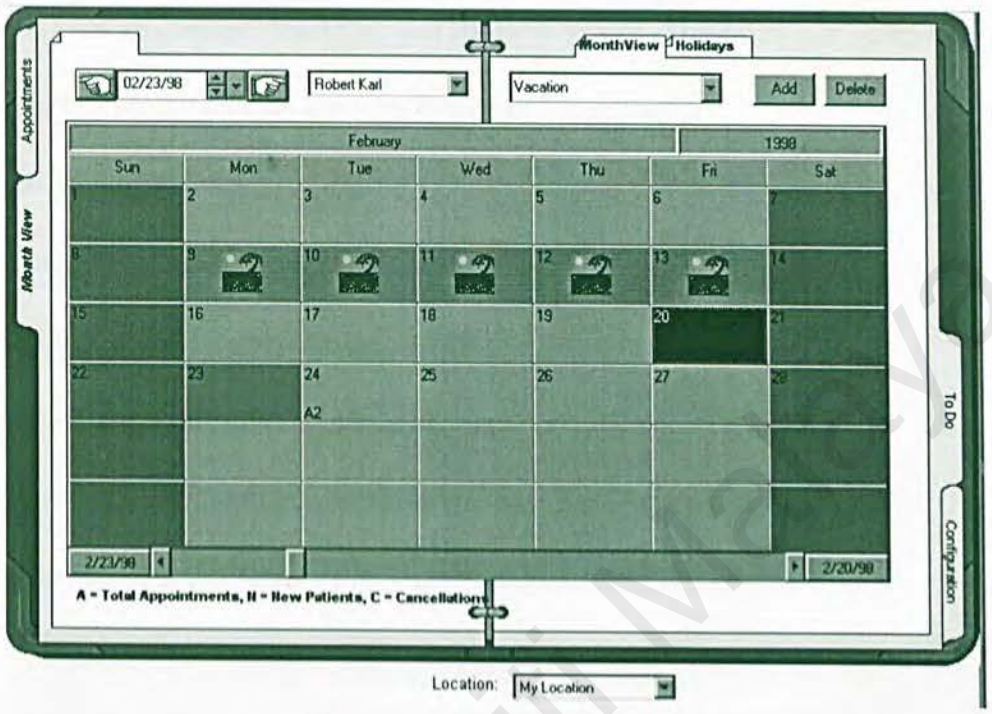


Figure 2-3: The Doctor's Appointment Interface of HIMS.

2.3.3.2 Traditional Chinese Medication Clinic Management System (TCMC)

This project is to develop a Chinese network based computer system to computerize some of the daily jobs. The culture of the TCMC is quite conservative and it only has a few modules such as registration, diagnosis and password control modules.

The system is required by the clinic to run in a LAN environment or may be web based with Client Server Architecture:

- The front end is run on P5 233 with Windows 95.
- The backend is run on P5 433 or higher with Windows NT.

- Backend RDBMS is Microsoft Access.

TCMC is written in Active Server Pages (ASP). The web server manages the operation of the LAN. It handles and distributes the requests of clients. Internet Information Server 4.0 (IIS 4.0) is used to connect front end and the web server. The users can access the system by using web browser such as Internet Explorer 4.0 or higher.

The features of TCMC are:

- Computerize the management of the clinic.
- Record and systemize the information of patients
- To store the information of herbs.

The following shows some screen shots of the TCMC system user interfaces (Figure 2-4 to 2-5). Figure 2-4 is the patient registration layout of TCMC system.

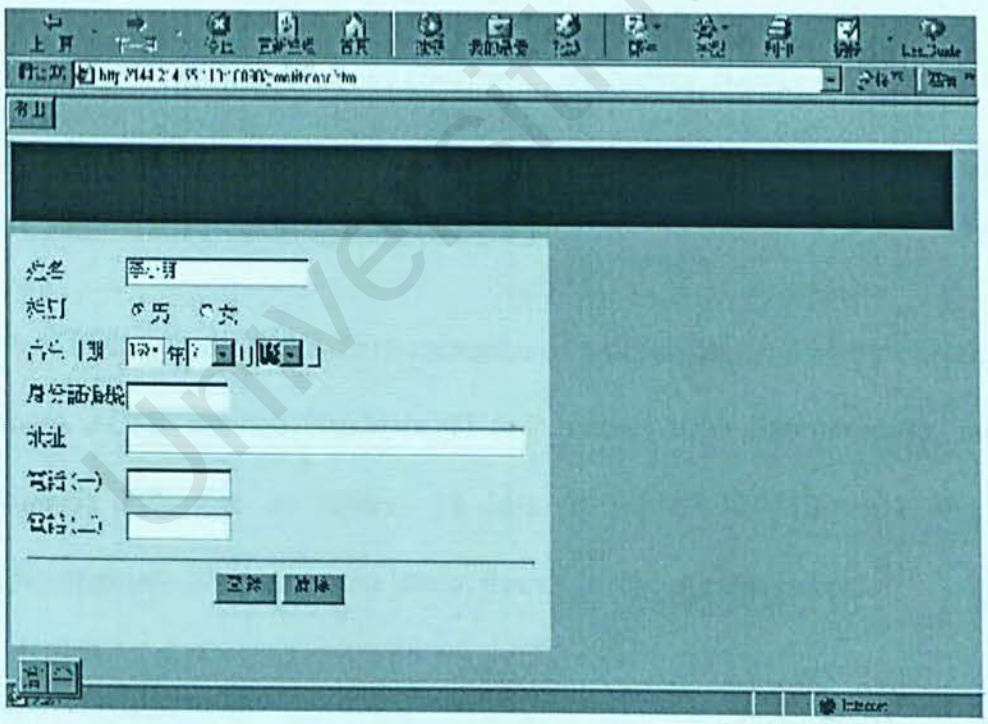


Figure 2-4: The Patient Registration Interface of TCMC.

Figure 2-5 shows the diagnosis layout of TCMC.

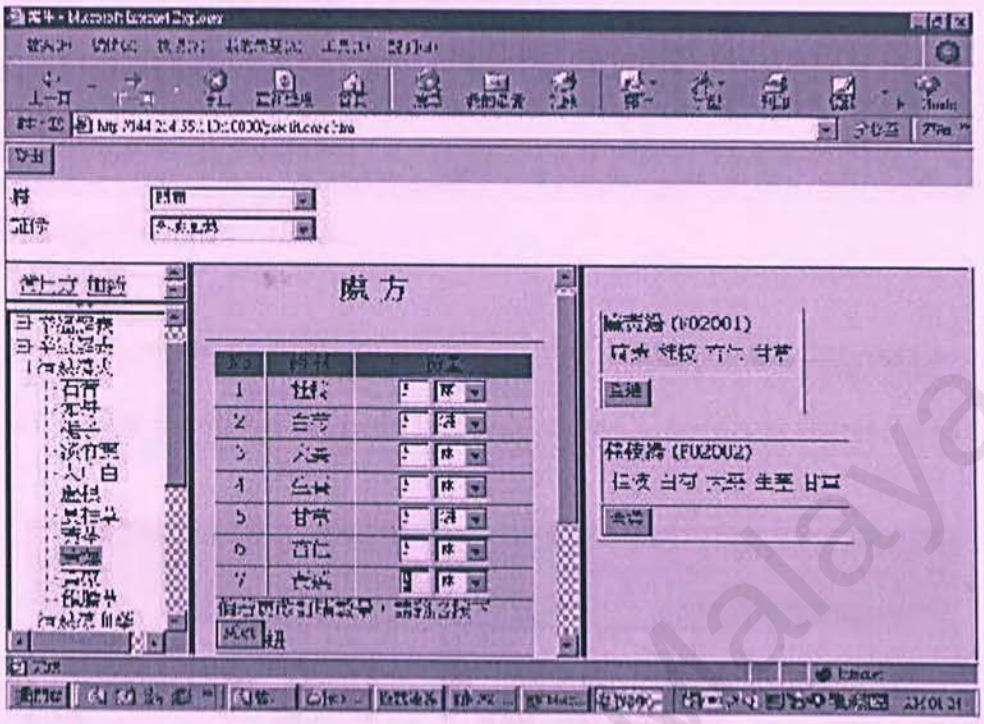


Figure 2-5: The Diagnosis Interface of TCMC.

2.3.3.3 e-Clinic Management System (e-CMS)

This system is designed for the daily operation of a clinic on local area network (LAN) environment. e-CMS runs on Windows NT or Windows 2000. Recommended minimum architecture is PC- 486 or higher, 18 MB of RAM, 10 MB hard disk space. e-Clinic Management System has an interactive GUI. The system features:

- Patient registration.
- Doctor consultation and diagnosis.
- Dispensing of medication.
- Drug inventory.

- Printing of medical certificate, receipts, and referral letters.
- Online viewing of patients' past medical record.
- Panel invoicing.
- Setting patient appointment.
- Setting of staff duty roster.

The following shows some pages of the screen shot of the system user interface (Figure 2-6 to 2-8). Figure 2-6 shows the main layout of the e-Clinic Management System.

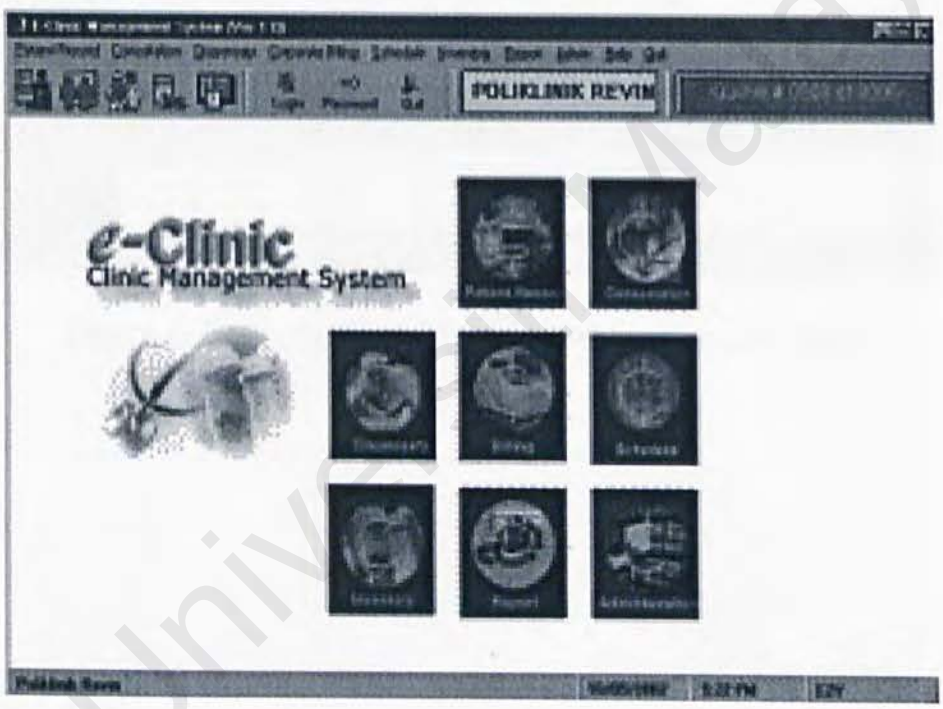


Figure 2-6: Interface of e-CMS's main page before user login.

The following layout is displayed when the doctor accesses the diagnosis and prescription module.

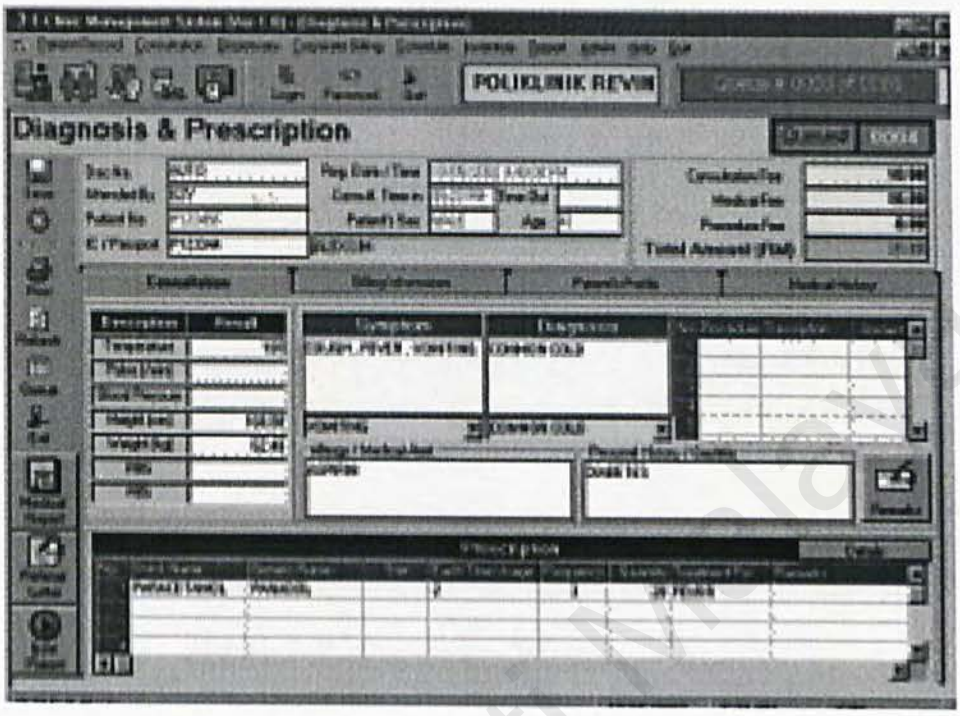


Figure 2-7: Diagnosis and Prescription Interface Of e-CMS.

The following layout is displayed when the pharmacist clicked the dispensary module.

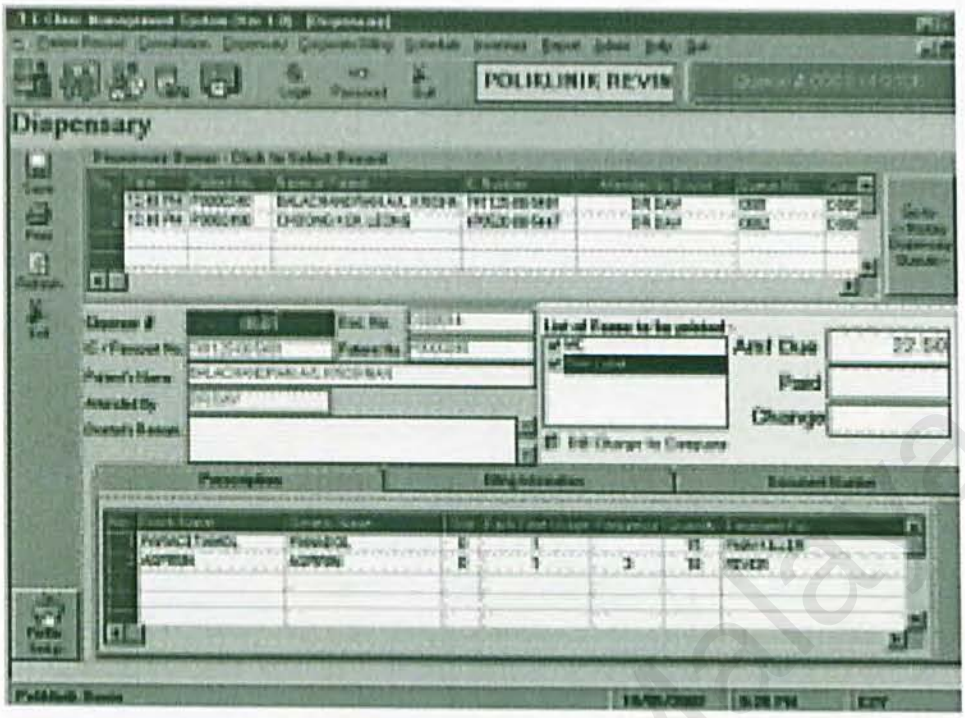


Figure 2-8: Dispensary Interface of e-CMS.

2.3.3.4 Clinic Management System 2000 (CMS2000)

Clinic Management System 2000 (CMS2000) is a Windows based clinic management system. It can be used as a stand-alone version or as a local area network version linking several computer terminals in a medical center environment.

CMS2000 features the following core modules:

- Patient Registration & Queue Management.
- Patient History Records (Electronic Records).
- Drug invoicing, Dispensing & Label Printing.
- Inventory Management.
- Reports and Statistical Analysis.

- Accounting for Private Corporate Patients.
- Patient Records Documentation.

The following is the screen shots of seven core modules of the CMS2000. The following layout is displayed when registration patient, consultation, dispensary, drug inventory and son on modules access by doctor, pharmacist, and nurse respectively.

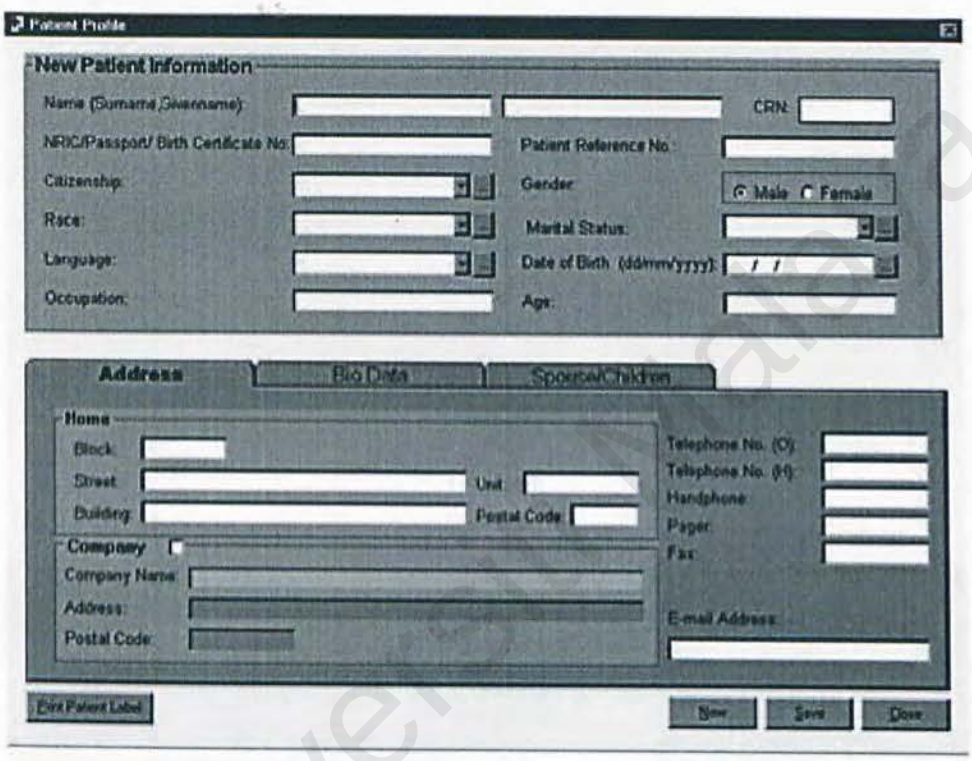


Figure 2-9: Patient Registration & Queue Management Interface of CMS2000.

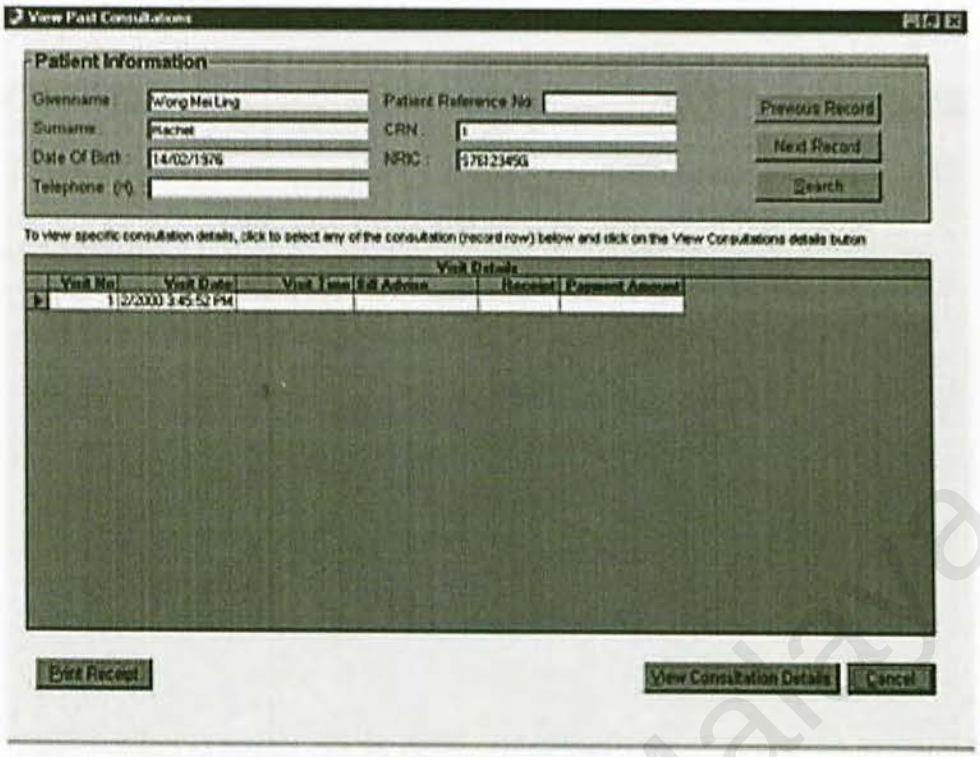


Figure2-10: Patient History Records Interface Of CMS2000.

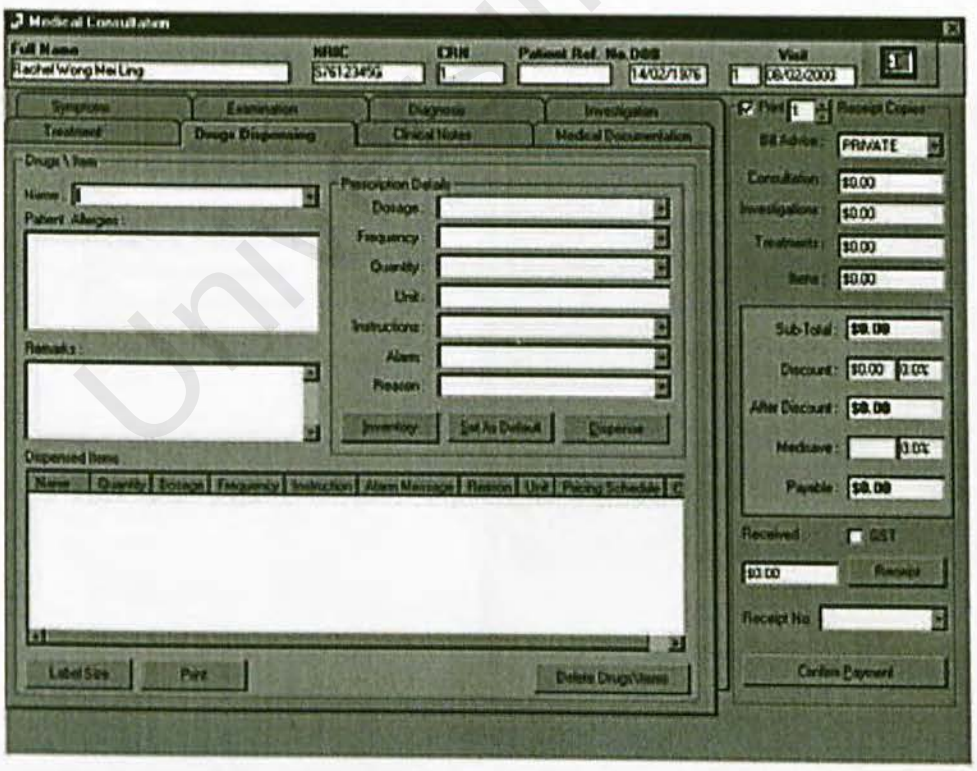


Figure 2-11: Drug invoicing, Dispensing & Label Printing Interface Of CMS 2000.

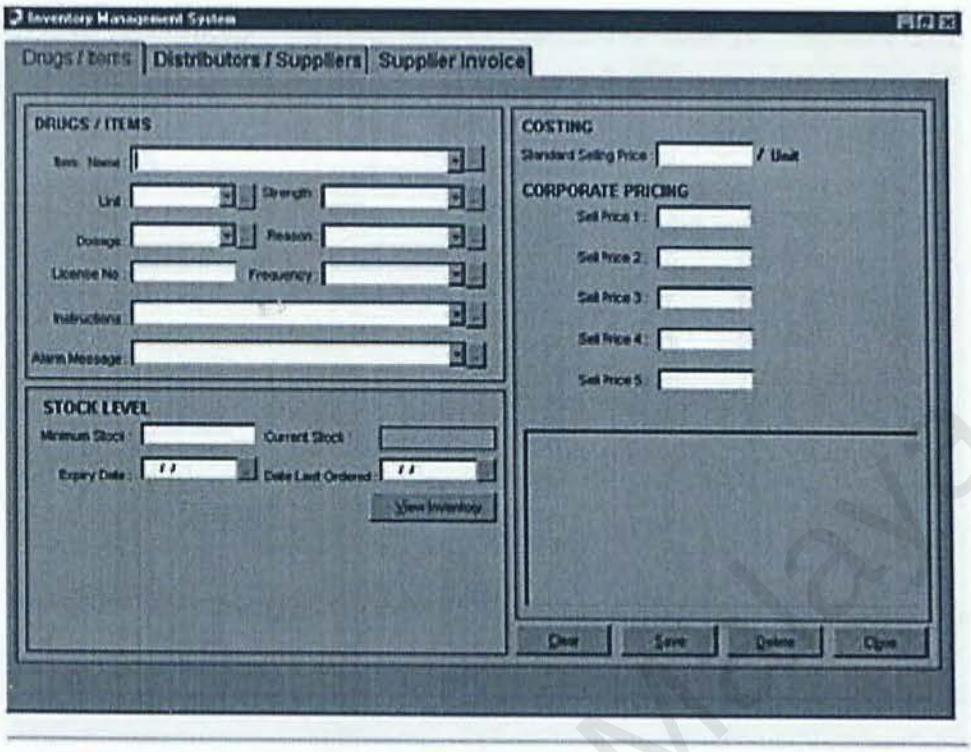


Figure 2-12: Inventory Management Interface Of CMS2000.

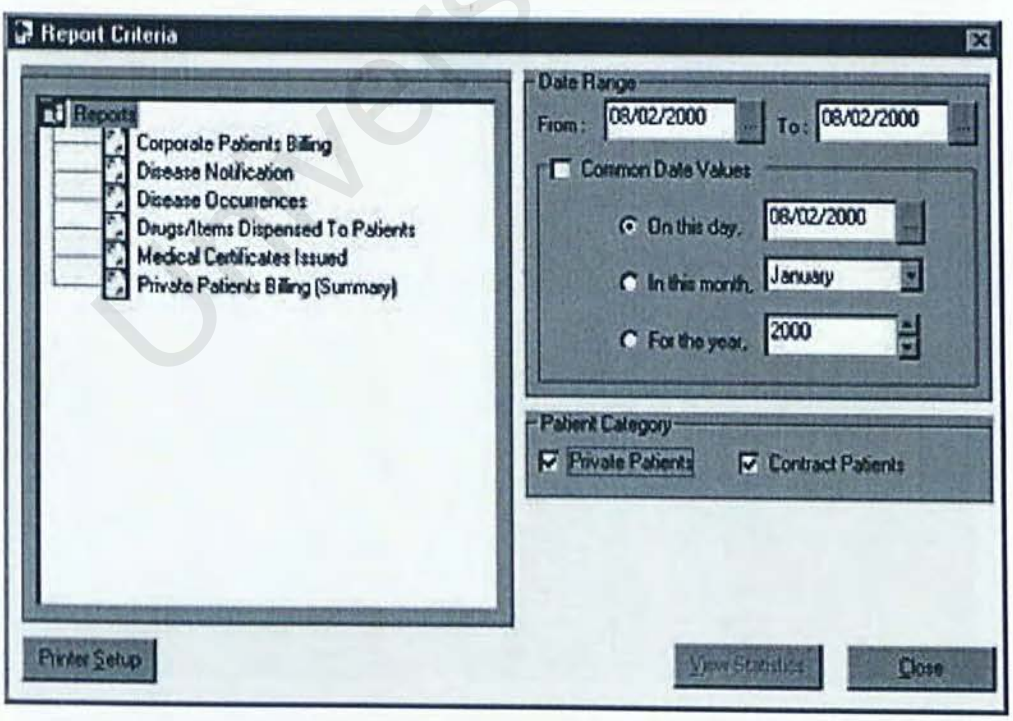


Figure 2-13: Reports and Statistical Analysis Interface Of CMS2000.

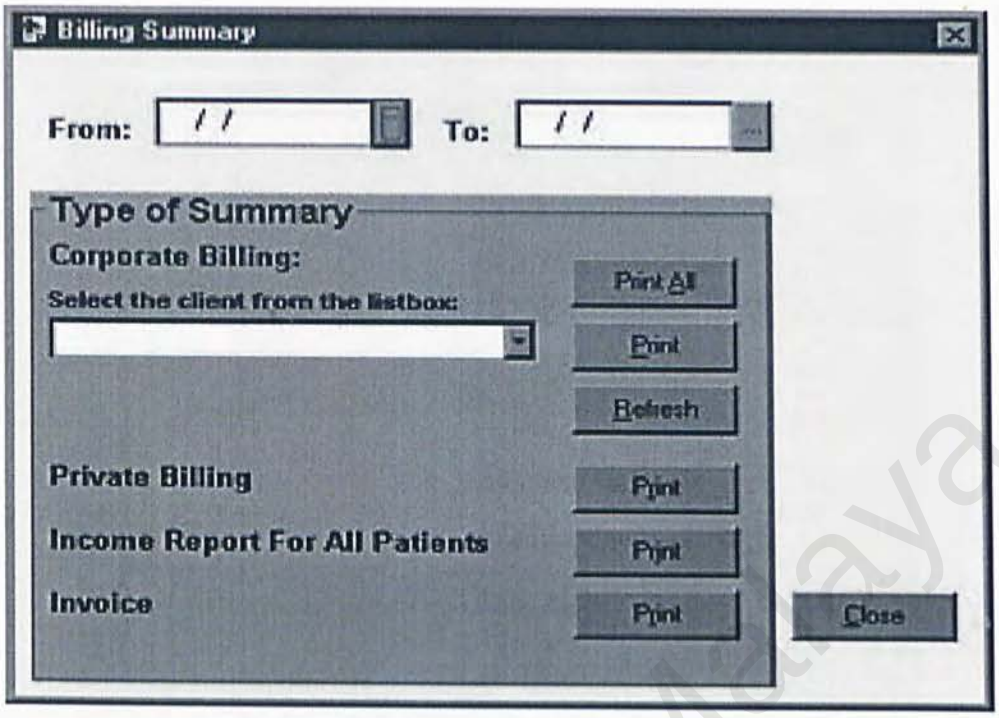


Figure 2.14: Accounting for Private Patients Interface Of CMS2000.

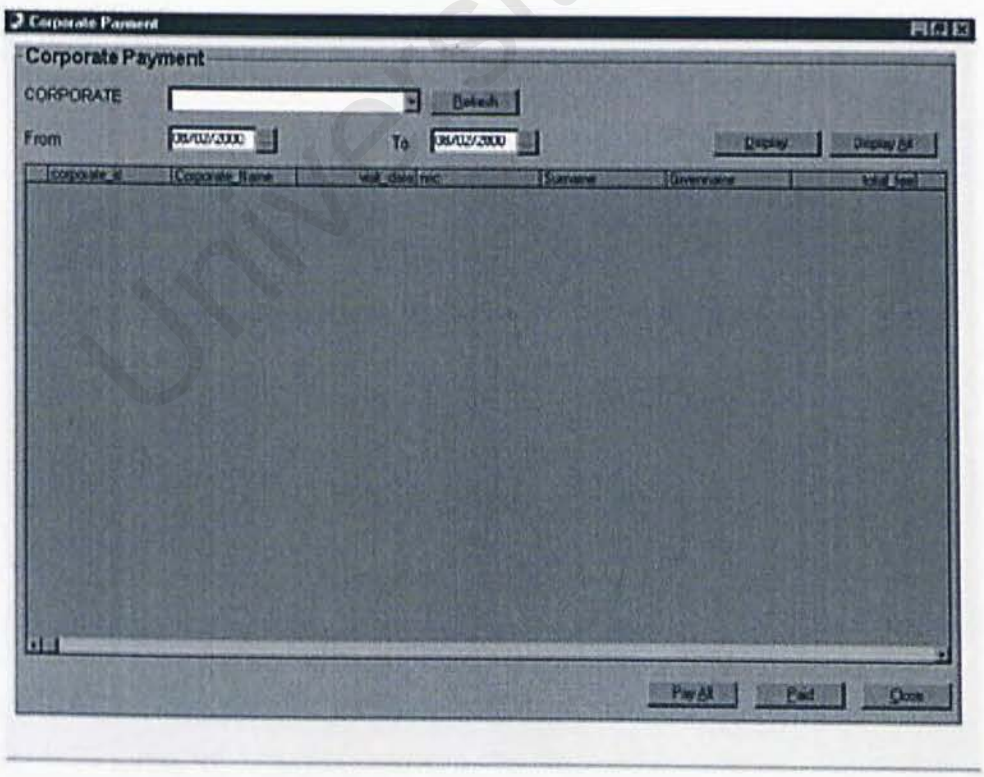


Figure 2-15: Accounting for Corporate Patients Interface Of CMS2000.

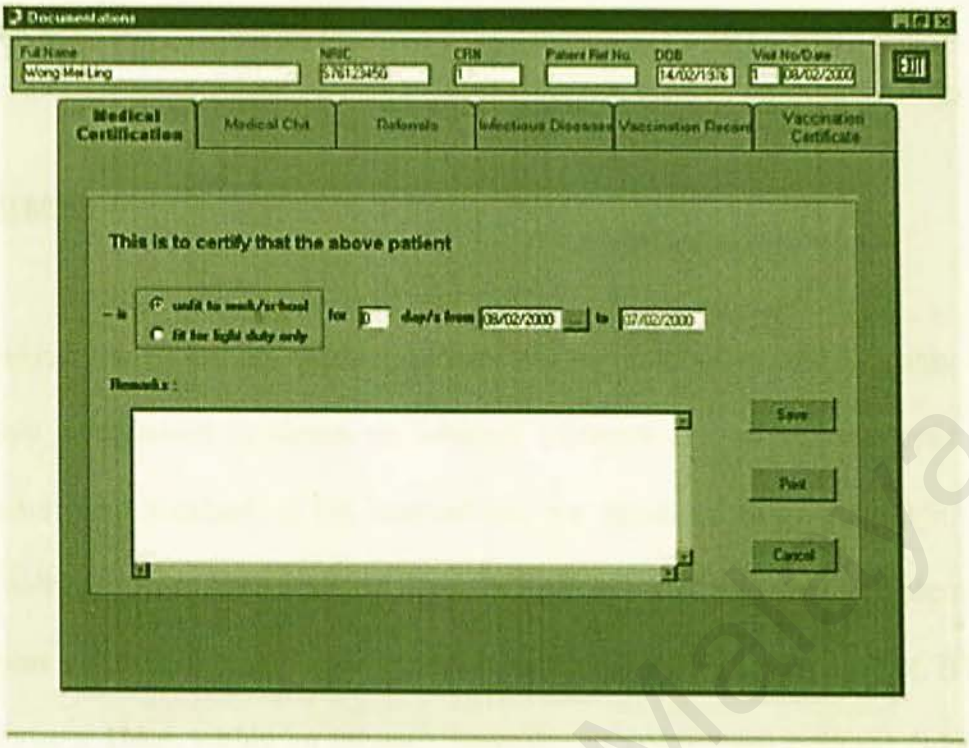


Figure 2-16: Patient Records Documentation Interface Of CMS2000.

2.4 INTRODUCTION TO INTRANET, EXTRANET AND INTERNET

2.4.1 Intranet

Technically, intranets are not much different from the Internet, except that only selected individuals are allowed to access an intranet. Intranets are private networks use the infrastructure and standard of the Internet but are separated from the public Internet through software programs known as firewalls. It is low-cost way to distribute corporate information and uses Internet-based protocols, including TCP/IP, FTP, Telnet, HTTP and Web browsers. Users within an intranet network can venture out onto the Internet, but unauthorized users cannot come in to this private network.

2.4.2 Extranet

An Extranet is a network application that uses the Internet for secure business relationships with partners, suppliers, and customers. It is like an Intranet that the users can share with their partner organizations anywhere in the world. The advantages of Extranets are companies can extend their internal systems to external business partners.

Uses of Extranets:

- Extranet can let companies coordinate design, manufacturing, scheduling and delivery across the supply chain.

- Companies can leverage the Internet to provide support to customers and partners, and to access remote workers all over the world.
- Secure Extranet applications can be use to reduce the cost of networks for financial transactions.

2.4.3 Internet

The Internet is the largest computer network in the world. Advanced Research Project Agency (ARPA) of the U.S. government in 1969 conceived it and was first known as the Arpanet. Nowadays, It is also a network of computer networks, connecting all of the computers on their private networks to this massive public network. The Internet use Transmission Control Protocol/Internet Control to exchange information between two parties.

2.4.4 Conclusion

The comparison between Intranet, Extranet, and Internet can refer to table 2.1.

	Intranet	Extranet	Internet
Network	Private network.	Private network.	Public network.
Geography	Located within a building.	A more widely geographical area, covering several building.	Spread to the whole world.
Ownership	Owned by the organization.	Owned to the organization, but with	Nobody owns the Internet.

		extended services to other parties.	
Number of users	Less then a few hundred.	Thousands.	Infinity.
Protocols	Own set of protocols.	Internet protocols.	Internet protocols.
Information sharing	Accessible within the organization only.	Accessible within a company and authorized users outside organization.	Everyone who has the Internet connection.

Table 2.1: Comparison between Intranet, Extranet, and Internet.

2.5 CLIENT SERVER ARCHITECTURE REVIEW

Client Server architecture refers to the manner in which components of the applications and functions of all the systems are stored, where they are stored, and how and where they are executed. Today, the common utilized architectures are the two-tiered architecture, the three-tiered and N-tiered architectures.

2.5.1 Two-tiered Architecture

Two-tiered splits the processing works in two. The user interface runs on the client and the database is stored on the server. However, the majority of the application logic runs on the client. The client typically sends requests to a server database. This architecture called fat client because a big chunk of the application runs on the client side.

In this Two-tiered architecture as show in Figure 2-17, user runs on the client. It sends the request over a network to the server. The server processes the request and returns the results. To access data, clients must know how the data is organized and stored on the server side.

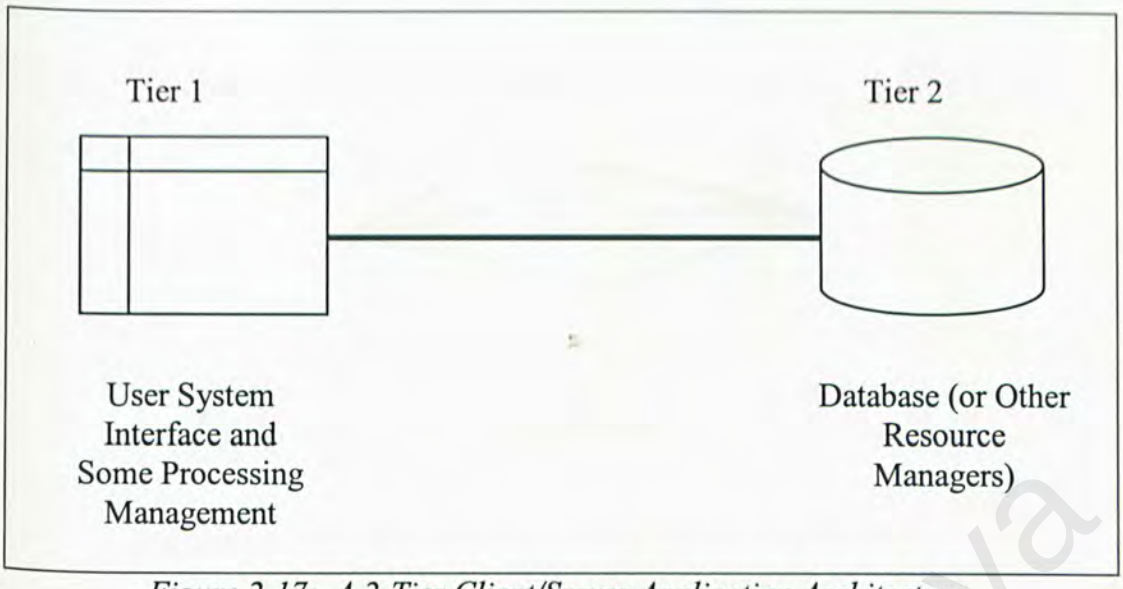


Figure 2-17: A 2-Tier Client/Server Application Architecture.

2.5.2 Three-Tiered Architecture

The three-tiered architecture introduced to overcome the limitations of the two-tiered architecture. In the three-tiered architecture, a middle tier was added between the user system interface client environment and the database management server environment. It consists separate processes of each tier. For example, the user interface runs on the user's computer (the client). The middle tier that process data, which runs on a server and is often called the application server. And a database management system (DBMS) that stores the data required by the middle tier. It runs on a second server called the database server.

In this Three-tiered architecture as show in Figure 2-18, when a client sends a service request, the middle tier hands it and processes the request. The application logic then accesses the database and returns the result.

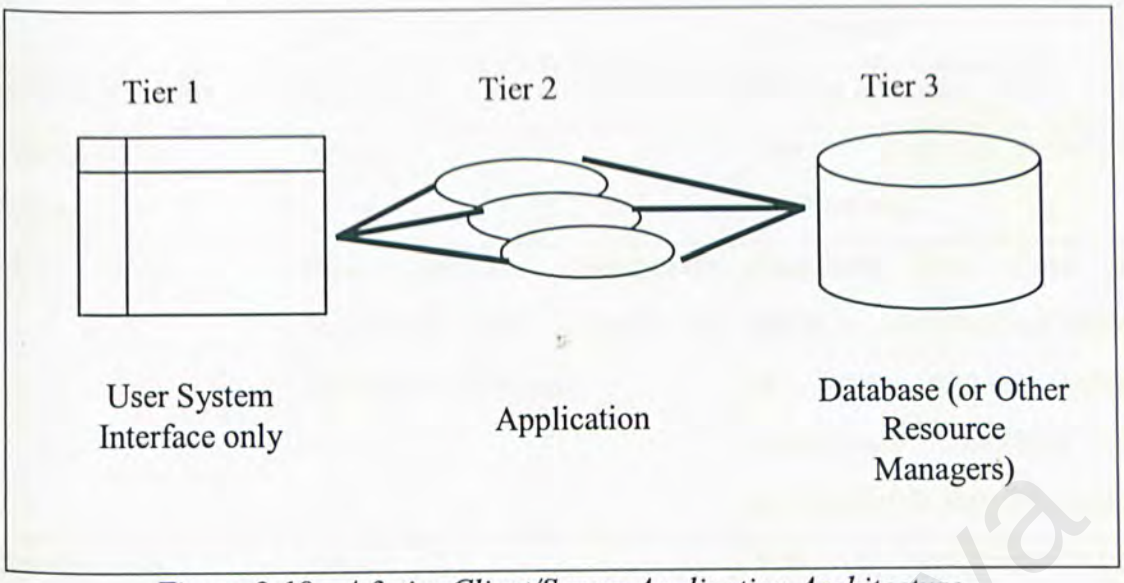


Figure 2-18: A 3-tier Client/Server Application Architecture.

2.5.3 Conclusion

The comparison between the two-tiered and three-tiered architectures can refer to table 2.2.

	2-Tier	3-Tier
System Administration	Complex (more logic on the client to manage).	Less complex (the application can be centrally managed on the server).
Security	Low (data-level security).	High.
Encapsulation of data	Low (data tables are exposed).	High (the client invoke services or methods).
Performance	Poor (many SQL statements are sent over the network).	Good (only services requests and the responses are sent between the client and server).
Scale	Poor (limited management of client communication links).	Excellent (can distributes loads across multiple servers).
Application reuse	Poor (mono-application on	Excellent (Can reuse services

	client).	and objects).
Ease of development	High.	Getting better.
Server-to-server infrastructure	No.	Yes (Via sever-side middleware).
Internet support	Poor (internet bandwidth limitations make it harder to download fat clients).	Excellent (thin client are easier to download as applets or bean; remote service invocations distribute the application load to the server).
Heterogeneous database support	No.	Yes (3-tier application can use multiple database within the same business transaction).
Hardware communication choices	Limited (only has a client and a server).	Excellent (all 3-tier may reside on different computers, or the second and third tiers may both reside on the same computer).
Availability	Poor (can't fail over to a backup server).	Excellent (can restart the middle tier components on other servers).

Table 2-2: Two-tiered Versus Three-tiered Client/Server Architectures.

2.6 GRAPHICAL USER INTERFACE (GUI) REVIEW

A GUI is a computer human interface on a computer. A GUI is a program interface that uses the computer' graphics to make the program easier to use. Well-designed GUIs can free the user from learning complex command languages. The user issues commands via the GUI to computer applications. GUI usually has three major components. These three components are a windowing system, an imaging model, and an application program interface (API). The windowing system builds the windows, menus, and dialog boxes on the screen. The imaging model defines the fonts and graphics on the screen. The API is the user specifies how and what windows and graphics appear on the screen.

The principles for good GUI design are listed as the following:

- **Understand people.**

  Applications must reflect the perspectives and behaviors of their users.

- **Be careful of different perspectives.**

  Developers should understand well the meaning of metaphors when come to the icon design. They should also need to know the overall behavior of the application well before start designing it.

- **Design for clarity.**

  GUI design is often not clear to the end users. One effective way to increase the clarity of applications to develop and use a list of reserved words. By using the same set of reserved words in all the applications, end user will find it much easier to understand for the basic instructions. Some of the basic reserved words are *OK, Cancel, Close, Exit, Help*, and *Save*.

- **Design for consistency.**

  Good GUI design will apply consistency in the entire behavior throughout the application. Consistency in designing interfaces is very important because every new behavior the developers provided in software can become an anxiety and will create confusion to the user.

- **Avoid an amount of information.**

  The interface should display only what the user needs to perform the current operation. Good GUI design will consider using the understandable graphical icon to represent an idea, instead of using text. It can educe errors and performance time by the user.

- **Color.**

  Presenting different groups with different color and use a highlighting to indicate an item is selected. Brightness will be used to show the items are not active at a given time

- **Manuals and online help** are needed to explain program features and how to use them to solve real world problems.

- **Provide Undo/Redo capabilities.**

2.7 TECHNOLOGY REVIEW

There are several suitable platform can be used for the development of the system. Each platform must be able to provide a suitable and comfortable environment for the development process. Several platforms have been analyzed such as Windows 98, Windows 2000, and Linux.

2.7.1 Windows 98

Windows 98 is the new version of the windows operating system after Windows 95. This operating is designed specially for the consumers. It allows the PC to perform and play better.

Windows 98 has improves on some key area, including operating application an average of 36 % faster, rendering Internet pages up to 25% faster and shutting down the PC up to 2 to 5 times more quickly than Windows 95.

Windows 98 has empowers a new range of hardware and entertainment functionality. It has included native support for Universal Serial Bus (USB), which make setting of the hardware become much easier.

2.7.2 Windows NT/2000

Windows 2000 is the latest evolution of Windows NT. The features inside the Windows NT/2000 are quite similar to Windows 95. However, WnidowsNT/2000 contains some of the new features, mainly concerned with security and network operation.

Windows NT/2000 gives access rights to user into several network services, including electronic mail and WWW. Every user would be assigned a username and password. For the first time a user use, he or she changes the password. After each NT/2000 session, user must log out to prevent unauthorized person using the account.

Windows NT/2000 has dramatically increased the stability for network user and allows users to have one account across an entire network.

2.7.3 Linux

Linux is free; Linux is an operating system that acts as a communication between the hardware and the software of a computer system. The Linux kernel has all the features that every operating system should have. Some of the features included are multitasking , virtual memory, fast TCP/IP drivers, shares libraries, multi-user capability and also protected mode.

Linux is really a collection of small, easy-to-use commands. Linux has a free X Windows Graphical User Interface (GUI), similar to Microsoft Windows, which allows most X Based, programs to run under Linux without any modification.

Networking support in Linux is advanced and superior to most other Operating Systems. As an Internet server, Linux is a good choice. It supports all of the most common Internet protocol, and easily fits into Local Area Networks.

2.7.4 CONCLUSION

Table 2-3 shows the comparison between Microsoft Windows 98, Microsoft Windows 2000 Server, and Linux.

	Microsoft Windows 98	**Microsoft Windows NT Server**	**Linux**
Security	Not secure.	High security.	High security due to login facilities.
Cost effective	Expensive.	Affordable with fully functional Internet Server.	Free via download.
User interface	Graphical User Interface (GUI).	Graphical User Interface (GUI).	Strong command line plus GUI via Window Managers.
Stability	Fewer stables	. Stable.	Very stable.
Load Balancing	Not capable.	Available in Enterprise Edition.	-

Table 2-3: Comparison Between Windows 98, Windows NT Server 4.0 ,and Linux.

2.8 RELATIONAL DATABASE REVIEW

A database management system (DBMS) gives the user access to their data and helps them transform the data into information. It allows users to easily create, update, and extract information from databases. There are several suitable databases can be used for the development of the system. Two relational databases such as Microsoft SQL Server 7.0 and Microsoft Access 2000 have been analyzed.

2.8.1 Microsoft SQL Server 7.0

Microsoft SQL Server 7.0 (MS SQL) is a Sequential Query Language (SQL) Server. It provides scalable, powerful data warehousing, and integration with Microsoft Office 2000. MS SQL is suitable to implement for large office with an amount of data.

2.8.2 Microsoft Access 2000

Microsoft Access 2000 is a relational database management (RDBMS) software created by Microsoft for small office or home user to use for storing data in relational format. It is designed for the Microsoft Windows operating system, such as Windows 9x, Windows NT, and Windows 2000. It can be used as a database in client/server system with data access interface paradigm such as Remote Data Object (RDO) and Data Access Object (DAO).

2.8.3 Conclusion

A comparison between these two relational databases was done such as Table 2-4.

Microsoft SQL	Microsoft 2000
Support larger database system.	Support small database system.
Allow multi user update the data at the same time.	Only one user can update the database at one time.
Highly security protection.	Provide basic authority protection.
Support large amount of user.	Not recommended.
Support large amount of transaction.	Not supported.
Designed for Client/Server computing.	Basic support for Client/Server architecture, but is most recommended for personal use.
Full programmability support.	Minimum programmability support.
Need skills and practice.	Specially design to beginner developer.
More expensive.	Cheaper.

Table 2-4: A Comparison between Microsoft SQL 7.0 and Microsoft Access 2000.

2.9 DATA ACCESS PARADIGM

There are a number of ways to connect to a database. You can use a System DSN, a DSN-less connection, or the native OLEDB provider. Several of standard interfaces for accessing database servers have been analyzed.

2.9.1 OLE DB

In the mid 1990s, Microsoft announced OLE DB, is an object-oriented interface that encapsulates data-server functionality. OLE DB was designed for relational databases and for many other types of data.

OLE DB is the next generation of the data access technology. Microsoft SQL Server 7.0 takes advantages of OLE DB within the components of the SQL Server itself. OLE DB is Microsoft 's strategic system–level programming interface to manage data across the organization.

OLE DB is an open specification designed to build on the features of ODBC. ODBC was created to access relational databases, and OLE DB is designed to access relational and non-relational information sources, such as mainframe ISAM/VSAM and hierarchical databases e-mail and file system stores, text, graphical and geographical data, and custom business objects.

OLE DB defines a collection of COM interfaces that encapsulate various databases management system services and allows the creation of the software components that implement such services. OLE DB component consists of data providers (that contain

and expose data), data consumers (that use data), and service components (that process and transport data).

OLE DB interfaces are designed to help components integrate smoothly so that OLE DB component vendors can bring high quality OLE DB components to the market quickly.

2.9.2 ODBC

In the early days, database connectivity was difficult. Everybody had their own database formats, and developers had to know a low level API for each database they wished to develop. There was a push for a universal API, an API which would work for numerous data stores. It was about this time that ODBC, or **O**pen **D**atabase **C**onnectivity, which was an early attempt at creating this universal API. A number of databases conformed to this standard, and became known as *ODBC-compliant databases*. ODBC-compliant databases consist of Access, MS-SQL Server, Oracle, Informix, etc.

The Open database Connectivity (ODBC) was developed in the early 1990s. ODBC is an interface that can process a DBMS without any program coding changes. It allows a developer to create a single application that can access database supported by different DBMS product without needing to be changed or recompiled.

ODBC was created to address the part of the problem that concerns relational databases and data sources, such as spreadsheets. Developers can call database servers using ODBC without using the native DBMS.

2.9.4 Conclusion

The following describes the revolution of some database history.

There are a number of ways to connect to a database, such as use a System DSN, a DSN-less connection, or the native OLEDB provider.

In the early days, database connectivity was difficult. Everybody had their own database formats, and developers had to know a low level **API** for each database they wished to develop. There was a push for a universal API, an API which would work for numerous data stores. It was about this time that **ODBC**, or **O**pen **D**atabase **C**onnectivity, which was an early attempt at creating this universal API. A number of databases conformed to this standard, and became known as *ODBC-compliant databases*. ODBC-compliant databases consist of Access, MS-SQL Server, Oracle, Informix, etc.

But, ODBC wasn't perfect. It still contained a lot of low-level calls, and was difficult to develop with. Developers had to focus more on low-level communications with the database, as opposed to being able to concentrate on getting the data they needed and using it how they saw fit. Along came Microsoft's solution**: DAO**, or **D**ata **A**ccess **O**bjects. Bellow is the example of the DAO codes:

```
ObjItem.AddNew
ObjItem.Name = "Chair"
ObjItem.Price = 10
ObjItem.Update
```

Table 2-5: DAO Codes

RDO (**R**emote **D**ata **O**bjects) came after DAO, targeted for distributed database architecture), and then **ADO**. However, these have all had their shortcomings. According to Microsoft, "ODBC provides native access to SQL data" and "DAO provides high-level objects to data". Even DAO and RDO require the data in a data store to be in SQL format (Structured **Q**uery **L**anguage). In response to these shortcomings, Microsoft introduced **OLE DB**, a COM-based data access object which provides access to all types of data, and even provides access to disconnected data stores (for example, if you're on your laptop, you can easily view a snapshot of the database from the last time you synced up).

OLE DB sits between the ODBC layer and the application. For example, with ASP pages, ADO is the "application" that sits above OLE DB. The ADO calls are first sent to OLE DB, which are then sent to the ODBC layer. It can connect directly to the OLE DB layer.

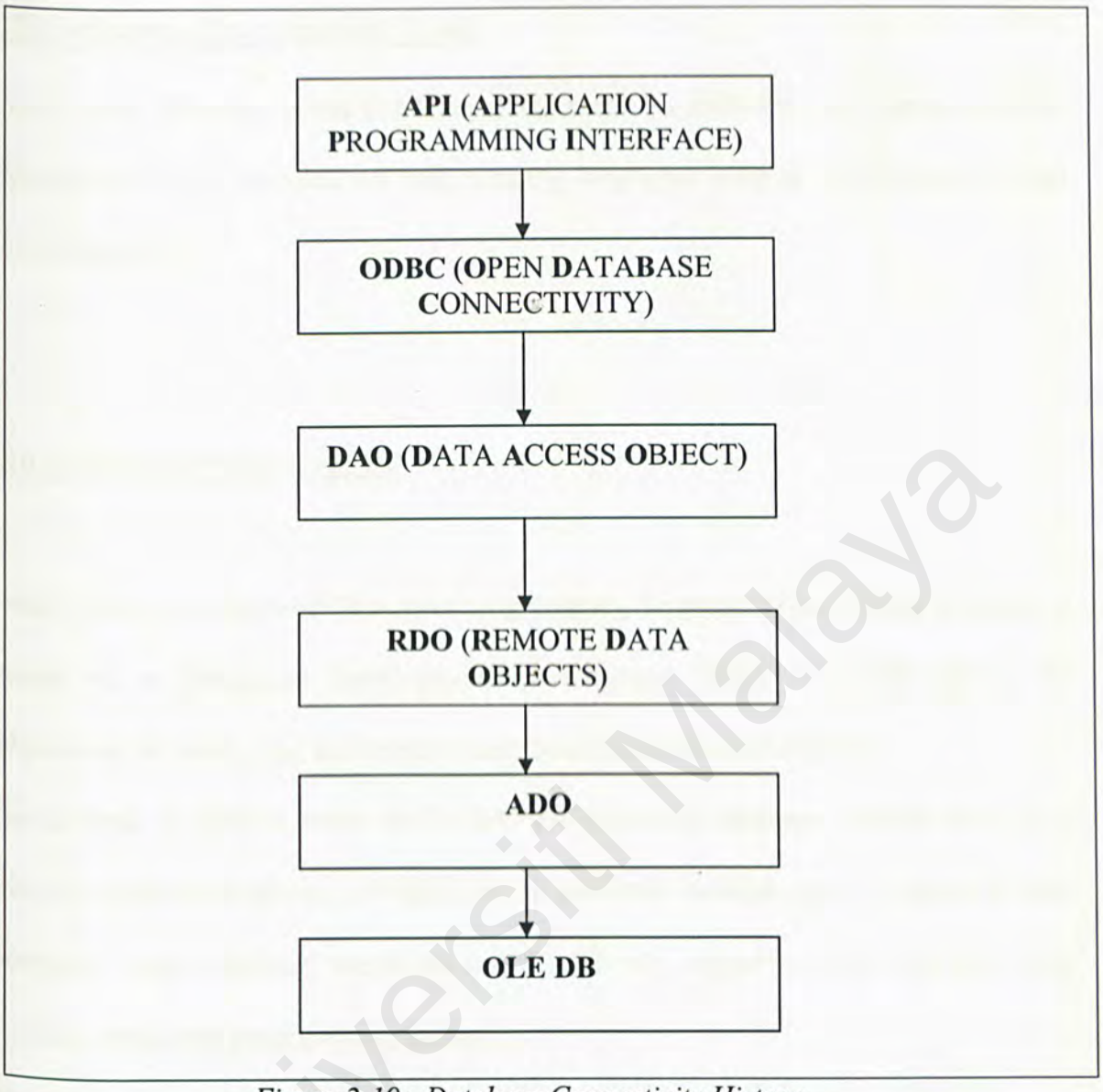


Figure 2-19: Database Connectivity History

The following table (as Table 2-6) shows the survey and analysis done by *Wrox*.

Performance Comparison					
SQL			**Access**		
	OLEDB	DSN		OLEDB	DSN
Connection Times (ms):	18	82	Connection Times (ms):	62	99
Iterating through 1,000 Records Times (ms):	2900	5400	Iterating through 1,000 Records Times (ms):	100	950

Table 2-6: The Performance Comparison of SQL and Access According to Their Database Connectivity.

2.10 Software Development Tools

Programming language is the most important part of a software development project. This section will be analyzed two programming languages, such as Visual Basic 6.0 and PowerBuilder 7.0.

2.10.1 Microsoft Visual Basic 6.0

Visual Basic is a Microsoft Windows programming language. Visual Basic program is created in an Integrated Development Environment (IDE). The IDE allows the programmer to create, run, and debug Visual basic program conveniently.

Visual Basic is derived from the BASIC programming language. Visual Basic is a distinctly different language, giving user the powerful features such as graphical user interfaces, event handling, access to the Win32 API, object oriented features, error handling, structured programming and others.

Visual Basic has done a great job in simplifies Windows application development.

2.10.2 PowerBuilder 7.0

PowerBuilder 7.0 delivers a brand new look and feel to the development environment. Sybase has enhanced its IDE to provide more flexible tool to help the developers with software creation. Now, PowerBuilder 7.0 has incorporated new technology for Internet, component, and n-tier software solutions.

In PowerBuilder 7.0, Sybase completely reworked the Debugger's interface to include its Pane/View metaphor. PowerBuilder 7.0's new Internet features include Wed Data Window (also called the HTML Data Window), ActiveX Data Window, and support for JPEG and GIF.

PowerBuider 7.0 can develop front-end application, which access Relational Database Management System (RDBMS) without coding in Third Generation Language, such as C++ or C. It uses its own power script that is a basic-like language that uses screen called pointers to graphically put together application. Power script is a Fourth Generation Language.

2.10.3 Conclusion

The following figure shows the survey and analysis done by Microsoft and Independent Hardware and Software Testing (NHST).

	Visual Basic 6.0	**PowerBuilder**
Debugger	Excellent	Fair
Compiler speed (debug)	Excellent	Excellent
Compiler speed (release)	Excellent	Poor
Compiler warning/error	Good	Poor
Editor	Excellent	Fair
Language	Good	Good
Installation	Good	Poor

Wizards	Good	Good
Report writer	Excellent	Good
Query writer	Excellent	Fair
Database designer	Excellent	Fair
Multithreaded support	Fair	Poor
Code browsing	Excellent	Poor
SQL syntax	Excellent	Fair
Win 32 API	Good	Poor

Table 2-7: Usability result Among Visual Basic and PowerBuilder.

2.11 CHAPTER SUMMARY

Due to the comparison done for each technology, Visual Basic 6.0 has been chosen as the development tool with using Microsoft Access to create and manage CCS database. Two-tiered Client Server Architecture and Windows platform were chosen for this project.

3. SOFTWARE DEVELOPMENT METHODOLOGY

Software Development is a time consuming and complex process. To produce a system, many different activities and processes need to be undertaken. Therefore, in order to control the process a standard model has been used to develop Clinic Care Information System (CCS).

3.1 WATERFALL MODEL

The waterfall model is also known as the "*traditional software lifecycle*". It follows a structured analysis and structured design approach in development. Figure 3-1 shows the model of the waterfall methodology. This waterfall model breaks the development cycle into six phases. Each phase is fully completed before the following phase is started.

The first phase in the waterfall model is **Systems Investigation**. Potential problems and opportunities are identified and considered to meet the goal of the system. Systems investigation attempts to answer the question " *What is the problem, and is it worth solving?* " The primary result of this phase is the problems and opportunities have been committed.

Once the systems investigation phase is finished, the outputs flow into the **Systems Analysis**. In this stage, systems analysis attempts to answer the question " *What must the system do to solve the problem?* " This phase involves the study of existing systems and work processes to identify strengths, weaknesses, and opportunities for improvement.

The users are interviewed, their requirements are analyzed, and the outcome of studying the existing system is produced.

The outputs from analysis are used to develop a design of the system. **Systems Design** seeks to answer the question " *How will the system do what it must do to obtain the problem solution?* " The result of this phase is a technical design. It details system outputs, inputs and user interfaces.

In the **Coding** phase, the programmer will develop the system according to the design. This is a phase, where transforming the design specification into software program.

Once completed, the system enters the **Systems Implementation and Testing** phase. The users are prepared to use and test the system. It is to test the operational system whether its meets the needs and goals of the system.

The final phase of waterfall model is **Systems Maintenance and Review**. It is to maintain and modify the system. The purpose is to make sure the system continues to meet any changing needs.

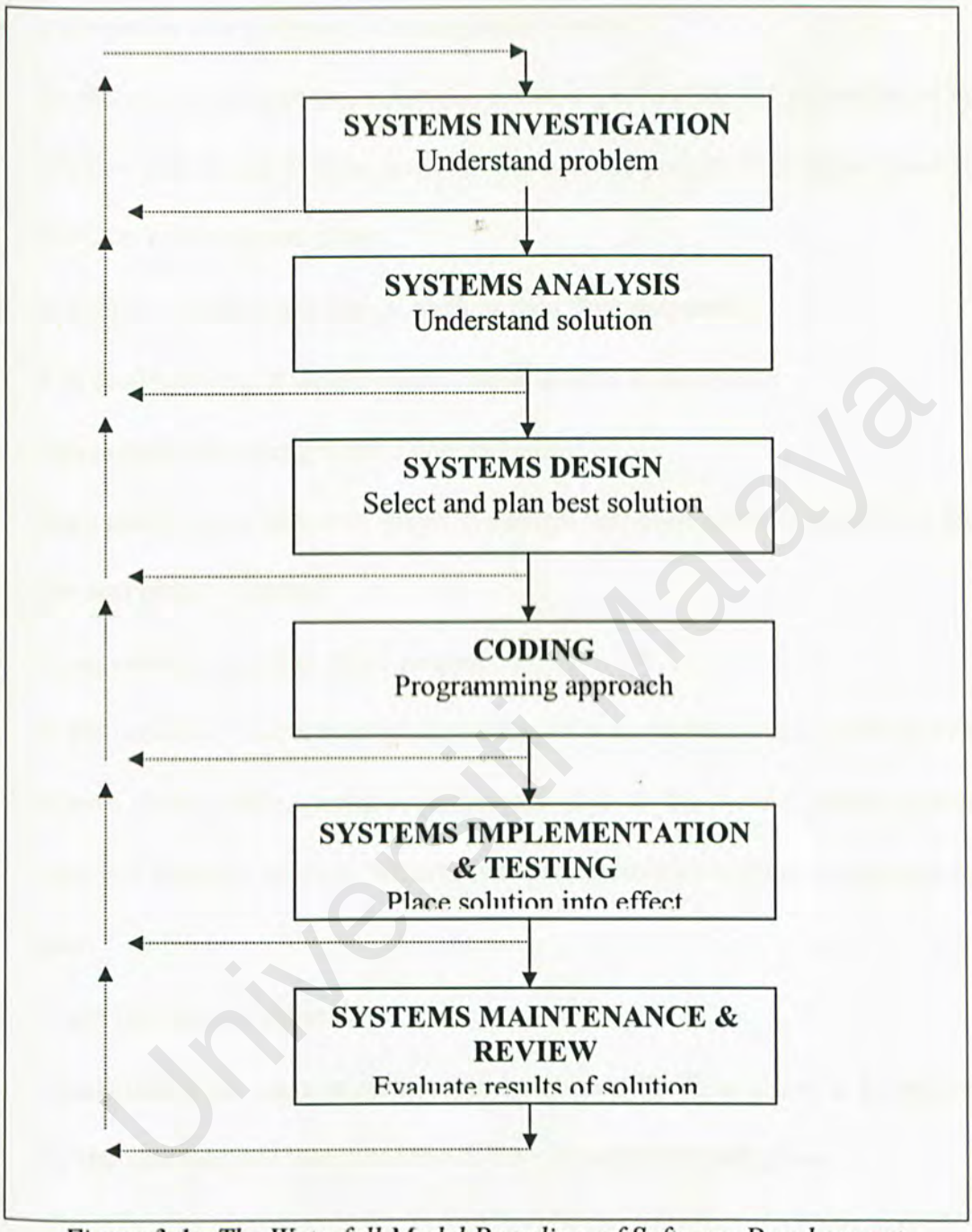


Figure 3-1: The Waterfall Model Paradigm of Software Development.

The advantages of the waterfall model are as follows:

- It allows for a large degree of management control.

  At the end of each phase, a formal review is performed and a decision is made whether to continue with the project, terminate the project, or perhaps repeat some the tasks of the current phase.

- It creates much documentation, such as data flow diagrams.

  It is useful when it is time to make modifications to the system.

- Developers and users get more understanding.

  Each phase is put into clear order. It ensures that each phase is completed before the next phase is started.

- It can estimate timeline of the project.

  If the user involved in each phase of the software development model, they may always change their system requirements. Due to the system cannot deliver on time and increase the cost. Waterfall model requires a minimum participate of the user.

- Users feel more comfortable.

  Validation of the users to system will be done at the final phase. It is importance for the user may not have time to validate the output at each phase.

3.2 CONCLUSION

With this methodology, the project will be carried out systematically and delivered on time.

3.3 CHAPTER SUMMARY

This chapter describes the waterfall model, which is used to develop the Clinic Care Information System. Each phase of the model is discussed in detail. The advantages of this model are also clarified.

4. SYSTEM ANALYSIS

System analysis is the phase where the requirements are analyzed and a detailed specification is produced for the new system.

4.1 FACT-FINDING

Fact-finding is used to find all the requirements that a user may want the new system. It also find out how the current system operates, what the user does and does not like the current system.

The fact-finding of the project was carried out in three forms, such as Internet surfing, reading and research, and interviewing.

- **Internet Surfing.**

  Many overview of the clinic management software and the information regarding this system are available online. By analysis the resources can help to determine the outcome of the desired system.

- **Reading and Research.**

  Studying the previous thesis and the reference books in document room and library has been done to gain a more understanding of the strengths and

weaknesses of the system. It gives a guideline to carried out Clinic Care Information System.

- **Interviewing**

  Interview was conducted with Klinik Pelajar Universiti Malaya executives. The purpose of this interview was to gain an understanding of the workflow of the clinic. In order to run this interview well, the interview did not go too long so that the executives would not become bored.

4.2 CURRENT SYSTEM ANALYSIS

Studying on current systems has been done on Chapter 2 Literature Review (Please refer to Section 2-3). Below are the findings on those current systems.

4.2.1 Non-computerized Medical Center System

Today, many clinics in Malaysia are still using manual system to carried out their daily tasks. The problems of the manual system are analyzed and explained as follows:

- **Paper orientated environment.**

  Large space is needed to keep all the documents and patient records.

- **Less security.**

  Paper files are easily misplaced and lost, and duplicated records is occurred when the clinician unable to find the previous record or file.

- **Time consuming task.**

  Finding files and data in the cabinet is the time consuming and highly dependence on clinician.

- **Data inconsistency**

  It occurs when the clinician is unable to read the handwriting record or file, which causes misspell problems.

- **Poor management in drug inventory.**

  Pharmacist has to count the quantity of the drug from time-to-time. The manual system is unable to produce up-to-date report of the drugs.

4.2.2 COMPUTERIZED KLINIK PELAJAR UNIVERSITI MALAYA SYSTEM

Klinik Pelajar Universiti Malaya System is currently using a computerized system to perform their various clinical tasks. However, there are a few problems related to the system. Theses problems are discussed as follows:

- **No appointment allowed.**

  Patients will get the doctor's consultation based on the '*first come first serve*' concept. Patients are not allowed to make any appointment for the coming consultation.

- **Still paper orientated environment.**

  Patients need to carry their medical report from registration department to consultation and dispensing department. Each department of the Klinik Pelajar UM is not linked and integrated.

- **Time consuming task.**

  The staffs check the patient details in book and the transfers the details to the computer system. This leads to considerable amounts of time and effort spent going trough the book.

4.2.3 EXISTING SYSTEM SUMMARY

Many clinical systems have been developed with the powerful functions and features, especially from western countries and have been put their system online.

The summary of existing systems is discussed as follows:

- All those existing systems are using Electronic Medical Records.
- Most if the existing systems provide secure online registration and appointment making through Internet.
- Pharmaceutical inventory management is handled effectively.
- Friendly graphical user interface in those existing systems.
- Perfect and error-free billing.
- Online medical records repository.
- Printing of labels, medication certificates, invoices, medical reports are available.
- Electronically payments through Internet.
- Automatic entry of amount tendered for credit card and cheque payments.

4.3 REQUIREMENT ANALYSIS

The requirement analysis is done in order to understand the requirements for the system. There are two types of the requirement, non-functional and functional requirements.

4.3.1 Non-functional Requirements

A non-functional requirement is a description of a restriction on the system that limits developer's choices for constructing a solution to the problem. These constraints usually narrow developers selection of language, platform, the implemented technique or tools. The non-functional requirements that should be met by Clinic Care Information System as follows:

- **User friendliness.**

  The project is designed to the concept of user friendliness. The users have the ability to use the program at the lowest possible of getting confused with the interface of the program. The interface of the system is designed to suit the users need.

- **Reliability.**

  The system should has an extensive testing to keep away all possible error and failure might occur. It is to ensure all the functions and features of the system perform exactly with the requirement of the users.

- **Security.**

The system should be able to discriminate between users and strangers. Users who would like to access the system have to enter a set of the user name and password for authentication.

- **Performance.**

  This system should be easily implemented in any environment and requires a minimum of hardware specifications.

- **Robustness.**

  An error message will be displayed when the error is detected. The system should control over the input is valid before continuing for the further data processing.

4.3.2 Functional Requirements

A functional requirement describes an interaction between the system and its environment. The functional requirements for Clinic Care Information System are determined from the outcomes of the fact-finding, such as interviewing, the studying of the existing systems, and the previous thesis.

CCS consists five modules. These modules are System Management, Patient Information Management, Appointment Management, Drug Management, and Registration Management. The Figure 4-1 is the modules being discussed in the project.

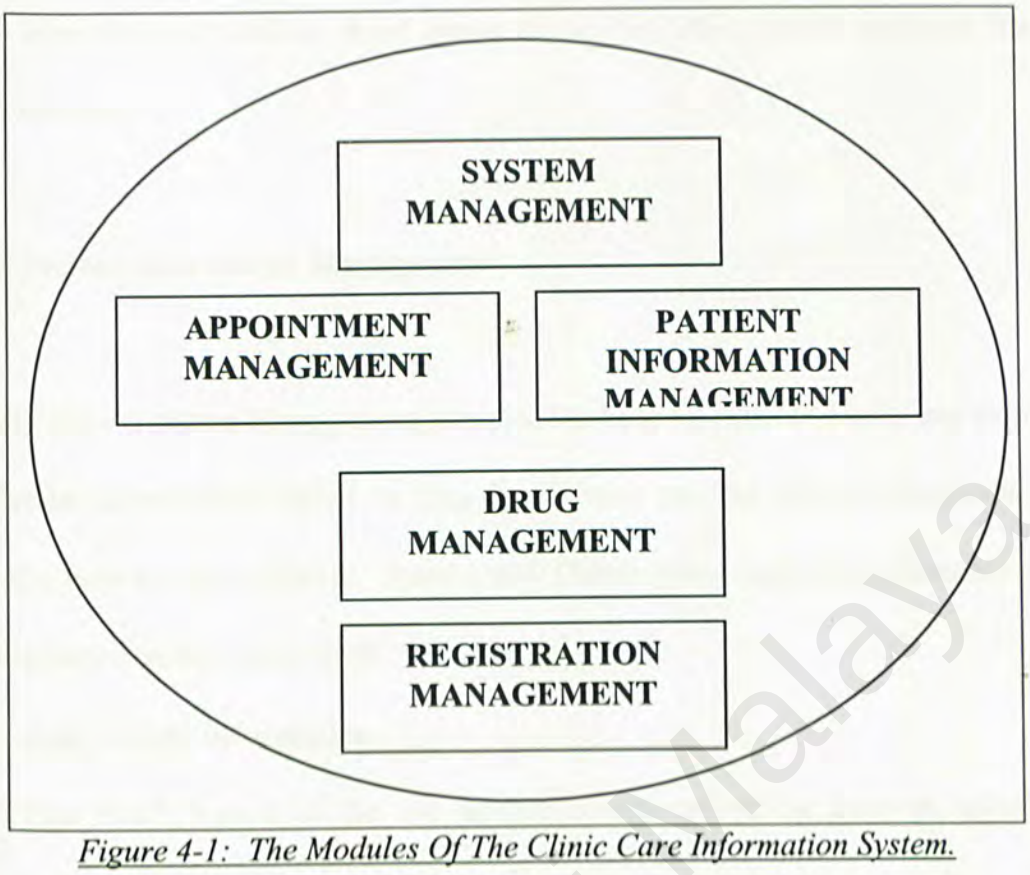


Figure 4-1: The Modules Of The Clinic Care Information System.

4.3.2.1 System Management

System Management Module provides the administrator of the CCS control over the system users.

- **Add User Profile.**

  The system administration can add a new account for an authorized user, set password and fill in the full personal information.

- **Edit User Profile.**

  The system administrator can edit the user details and change the password.

- **Delete User Profile.**

When the user account is not longer in use, the details can be removed from the system.

4.3.2.2 Patient Information Management

The Patient Information Management Module is to keep all patients details and records. It will also be automatically linked to drug dispensation process. The key functions are to allow the user to Add, Modify, Search, and Delete either individual elements of the patient's record or the record itself.

- **Add Patient Information.**

  The 'Add' feature of the sub-module covers everything from an addition of patient details, such as newly discovered patient allergy.

- **Modify.**

  This feature is designed to allow changes to be made to patient records. It covers everything from a simple change of address performed by nurse through to editing and storing of patient's history by an authorized doctor.

- **Search.**

  Searching of the patient's record allows for immediate viewing of patient's medical history, contact details, and others.

- **Delete.**

  This feature is allowed to delete the patient records.

4.3.2.3 Appointment Management

This module schedules the patients' s consultation appointment. The appointment settings include Add, Modify, Search, and Delete the patient's appointment records.

- **Add Patient Appointment.**

  Addition of an appointment.

- **Modify.**

  An appointment is changed either on patient's request or due to extraneous factor.

- Search the patient's Appointment Records.

  User can search patient's Appointment Records.

- **Delete.**

  Cancellation of appointment can be effected either for a specific appointment, for a patient in general, or for a particular period of time.

4.3.2.4 Drug Management

Drug Management Module is designed to give the clinic to have complete control over the inventory of drugs. It provides a view of expiry date of drugs. With this sub-module, the clinician can Add, Modify, Search, and Delete the drug details easily.

- **Add Drug.**

  This feature allows the new or exiting drug to be added to the system.

- **Modify.**

  This feature allows the user to make changes of the drug details.

- **Search.**

  The sub-module provides an overview of drug details, such as the quantity.

- **Delete.**

  When a drug is dispensed, the system automatically updates the database. The user can remove the drug from the database at a single click.

4.3.2.5 Registration Management

Registration Management Module allows user to manage the patient registrations. It will be automatically linked to Patient Information Management module. This sub-module consists Pre-registration, Modify, Search, and Delete features.

- **Pre-registration**.

  Addition a new patient.

- **Modify.**

  This feature allows the user to make changes of the patient details.

- **Search.**

  This sub-module allows the user search the patient records from the database.

- **Delete.**

  Deleting the patient records.

4.4 SYSTEM REQUIREMENTS

The system requirements describe the minimum requirements of software and hardware needed to support the system. Table 4-1 shows the software and hardware requirement for the project.

Hardware Requirements	
Components	**Descriptions**
• Microprocessor	-Pentium 166 .
• Random Access Memory (RAM)	-32 MB RAM.
• Storage	-6.4 GB hard disk.
• Additional Devices	-CD-ROM Drive, Printer.
Software Requirements	
Components	**Descriptions**
• Operating System	-Microsoft Windows 98 or later in standard.
• Database Management System (DBMS)	-Microsoft Access
• Programming languages	-Visual Basic enterprise edition version 6.0

Table 4-1: The Hardware and Software Requirements.

4.5 CHAPTER SUMMARY

In this chapter, the process of clarifying what the user needs and what the system must do have been considered. In order to achieve this project 's goal, one is to agree on the solution on the conceptual design, follow by detailed analysis of the user requirements, and then determine the system specification. Data collection through study the existing systems, previous thesis and interviewing have been done. The results of the finding have been analyzed and produced an analysis model on how this project should be build and also focus on the system requirement.

5. SYSTEM DESIGN

System Design is the crucial part in the development of system. The high effort is needed to put all the parts together into a workable system and become an existing system. The system design will include the functionality design, database design, and user interface design.

5.1 Functionality Design

System functionality design is based on the system functionality requirement. It translates the system requirements into system functionality.

In this stage, system is decomposed into smaller modules. A modular system is easy to understand, code, debug, and maintain.

The project design is focused on the system structured design and data flow diagram.

5.1.1 Structure Chart

This chart is based on the functionality modules. It depicts the functions of Clinic Care Information System (CCS). The CCS is decomposed into five main modules, such as Figure 5-1.

i. System Management Module.

ii. Patient Information Management Module.

iii. Appointment Management Module.

iv. Drug Management Module.

v. Registration Management Module.

Each module will be divided into sub-module in the following section.

5.1.1.2 Structure Chart of Clinic Care Information System

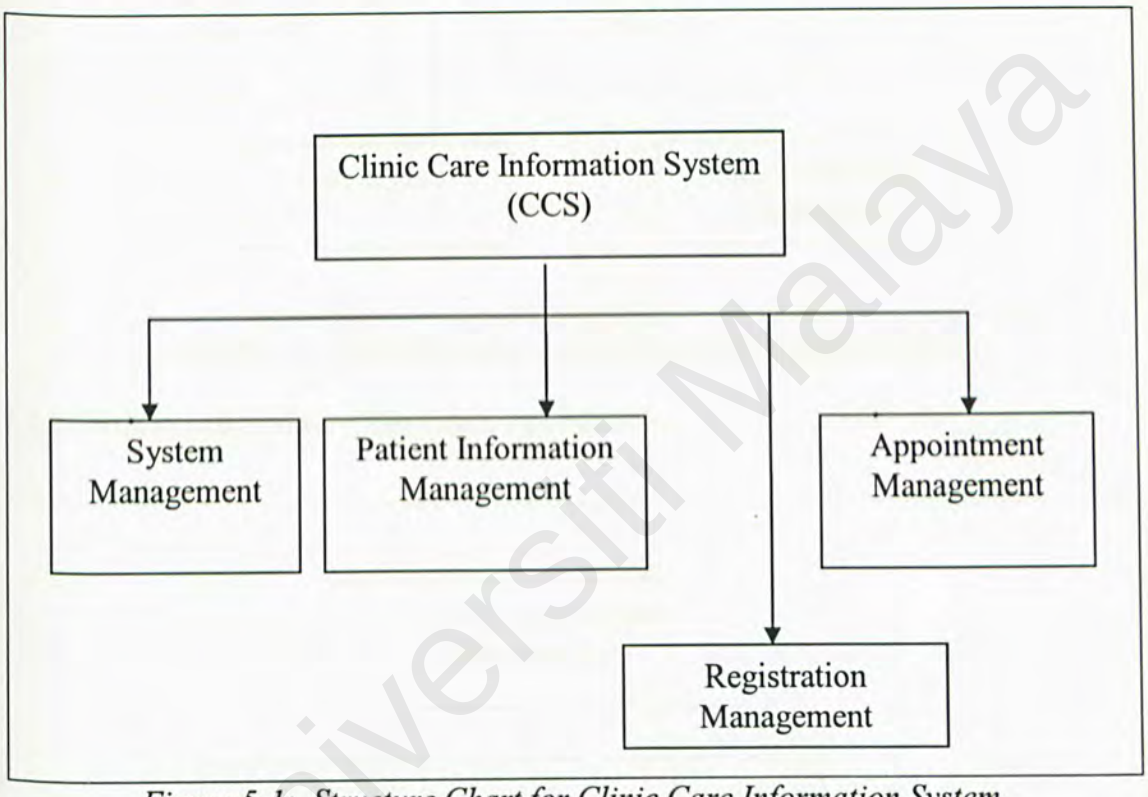


Figure 5-1: Structure Chart for Clinic Care Information System.

5.1.1.2 System Management Module

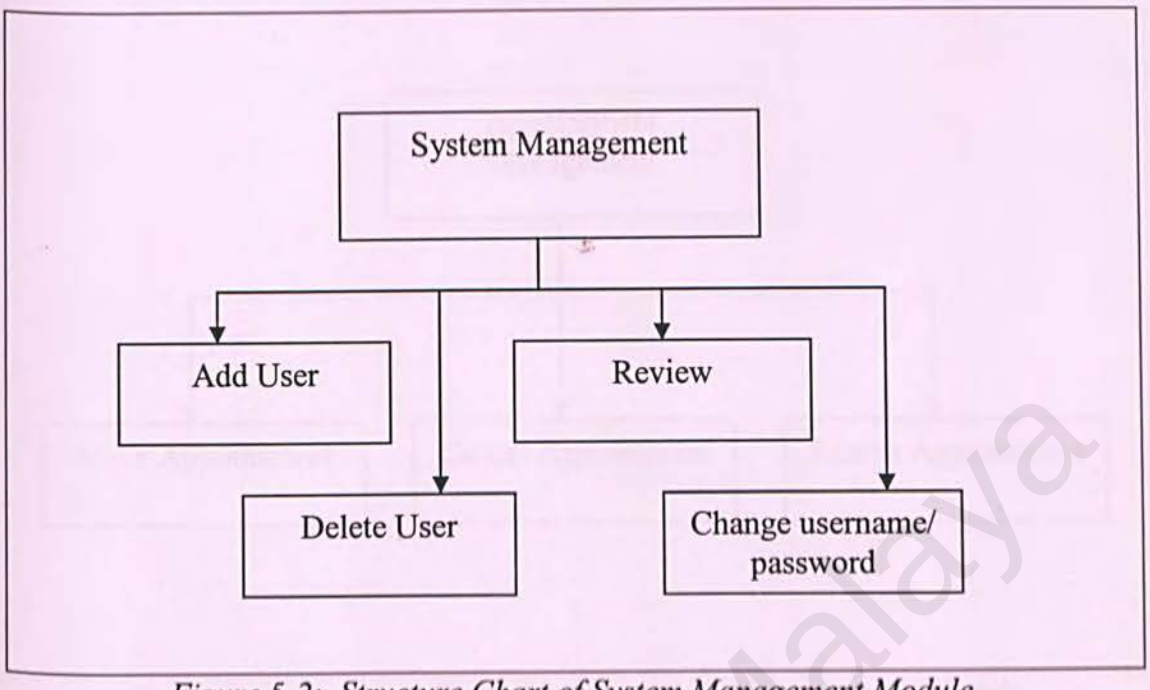


Figure 5-2: Structure Chart of System Management Module.

5.1.1.3 Patient Information Management Module

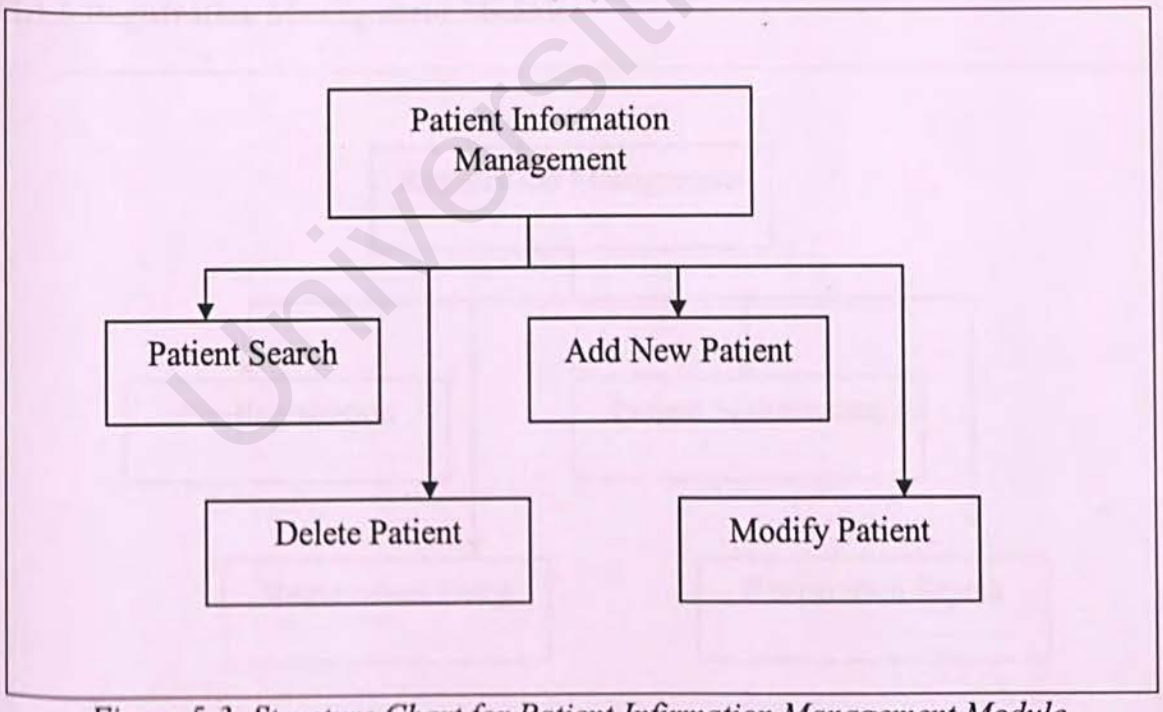


Figure 5-3: Structure Chart for Patient Infirmation Management Module.

5.1.1.4 Appointment Management Module

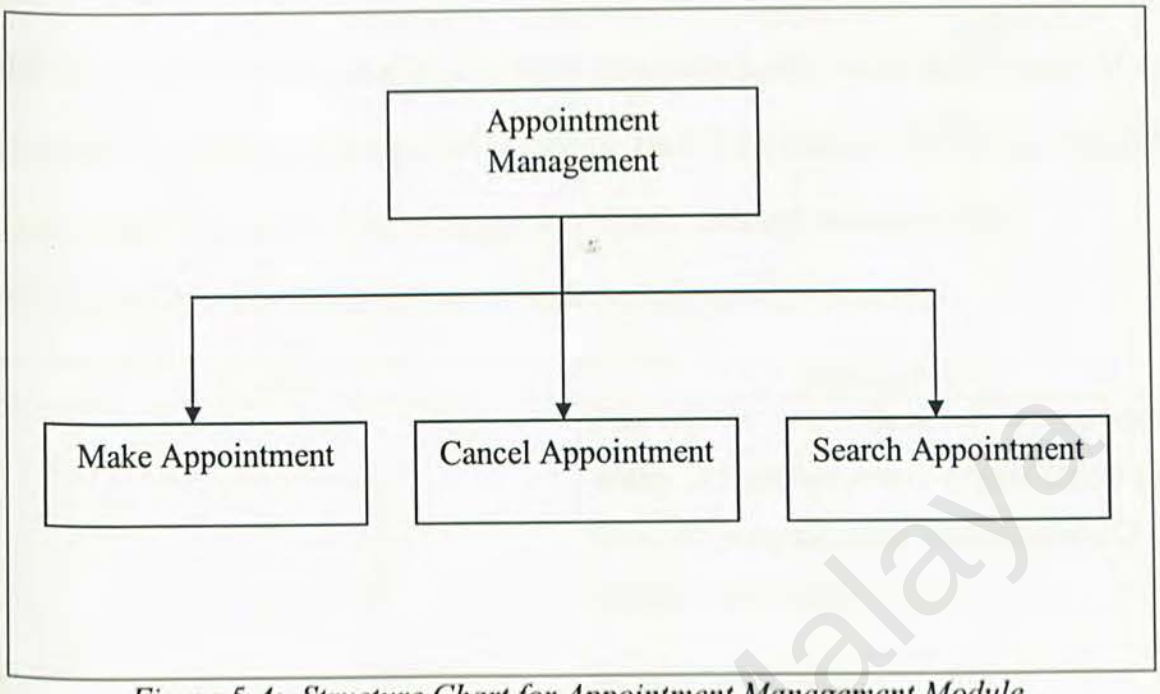


Figure 5-4: Structure Chart for Appointment Management Module.

5.1.1.6 Registration Management Module

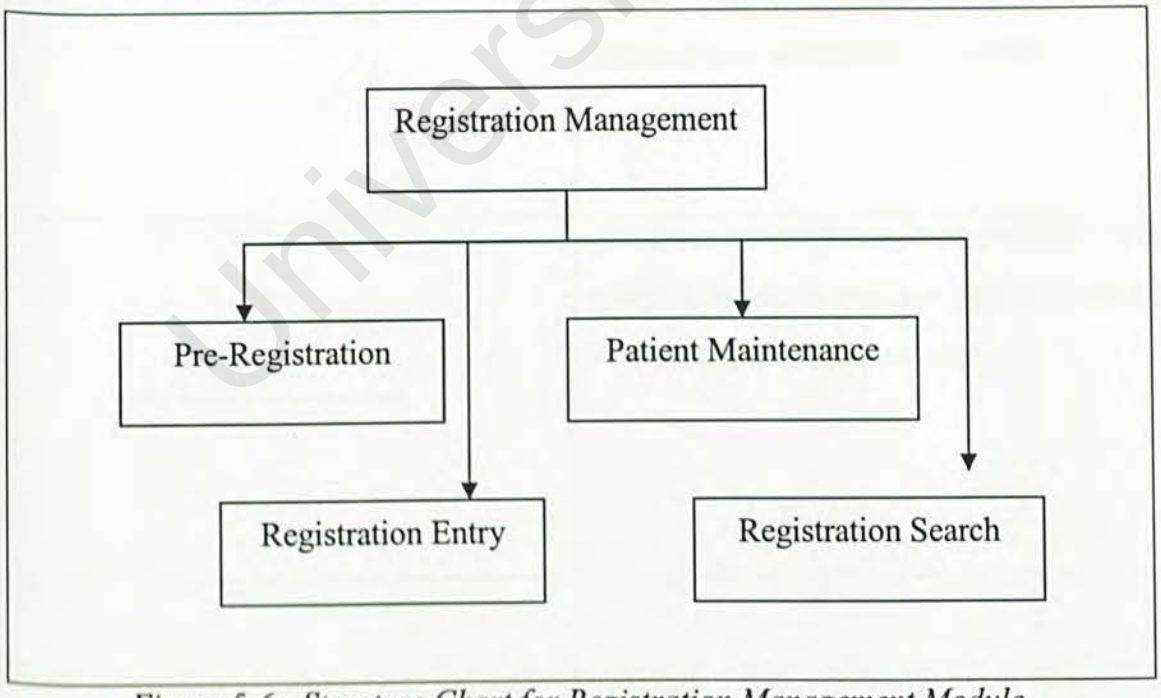
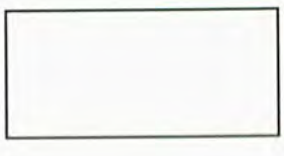
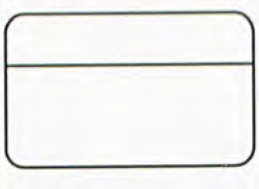
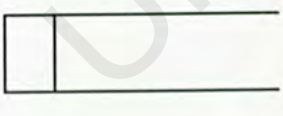


Figure 5-6: Structure Chart for Registration Management Module.

5.1.2 Data Flow Diagram (DFD)

DFD is a tool in s system analyst's toolbox. It is much easier to get an overview of an information system by studying a DFD then by readying a lengthy description. Gane & Sarson approach is used for the instructions of Clinic Care information System.

The symbols shown in Table 5-1 are the basic building blocks for a DFD.

Symbols	Description
	The square box represents an **external entity**. An outside source or destination for data. An external entity provides data. It is labeled with a singular noun.
	A rounded rectangle, called a **process** box, represents an activity. A process box label usually begins with a verb to show that a process does something to the data.
	An open-ended rectangle represents a **data store**. It is data at rest, such as when data is stored on a file on a hard disk.

	An arrow represents a **data flow** – data that will be processed or stored by the system. Arrows at the end of a data flow indicate what happens to the data. Whether it is headed for a process, data store or external entity.

Table 5-1: The DFD Symbols.

5.1.2.1 Context Diagram, O Level and Level 1 Diagrams

Below is the Context Diagram of the project.

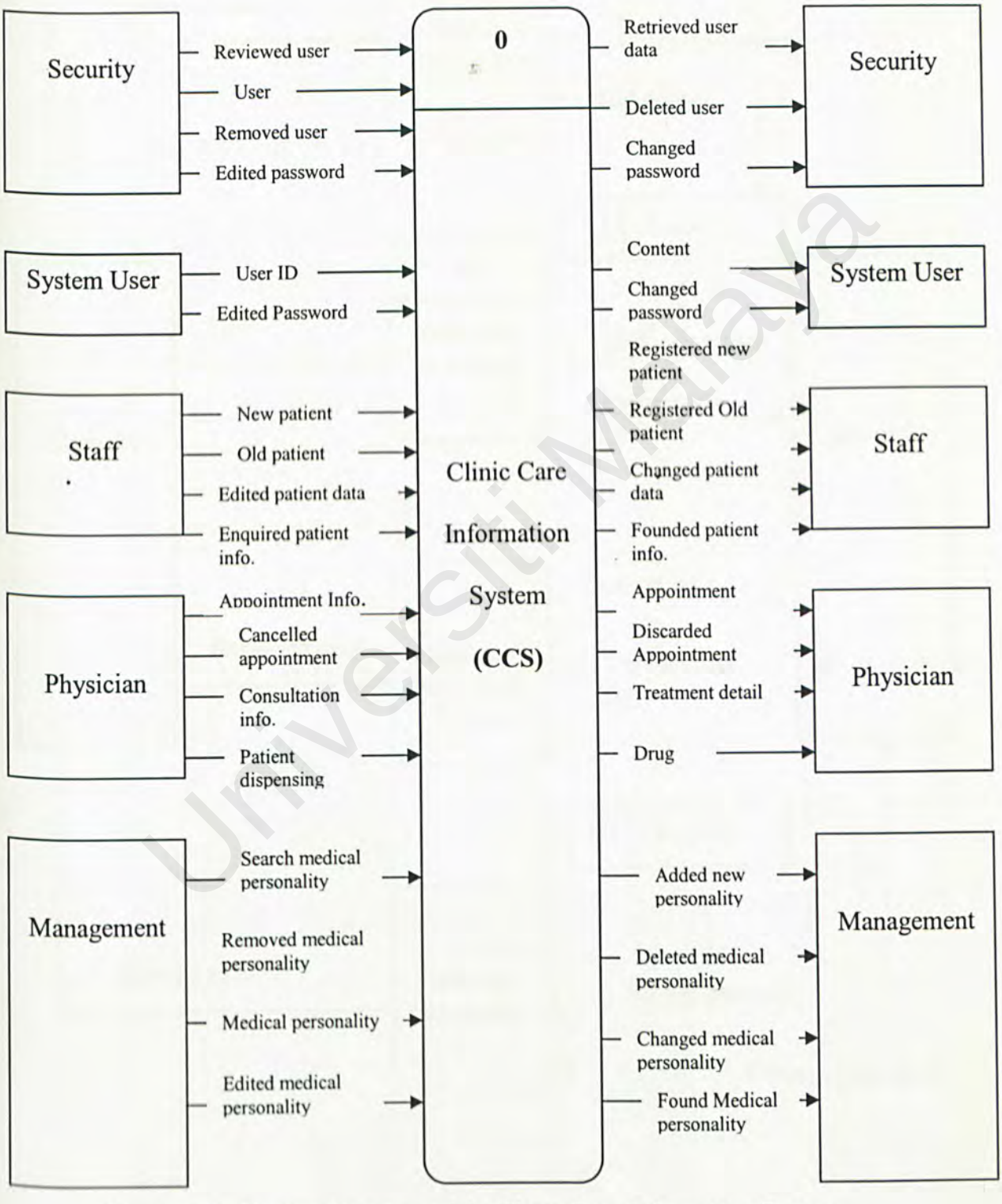


Figure 5-7: Context Diagram of the Clinic Care Information System.

Figure 5-8: O Level Data Flow Diagram of Security for CCS.

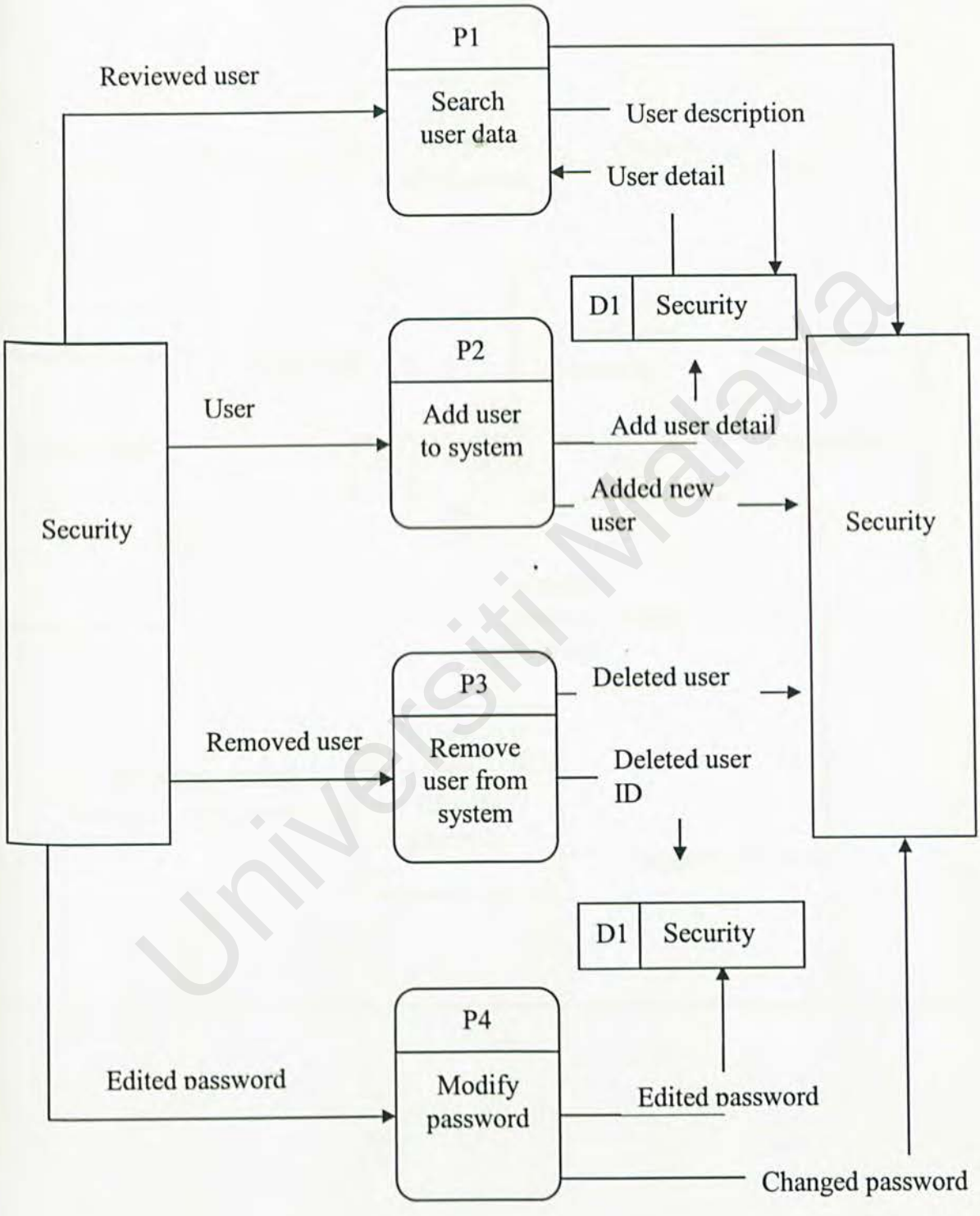


Figure 5-9: O Level Data Flow Diagram of System User for CCS.

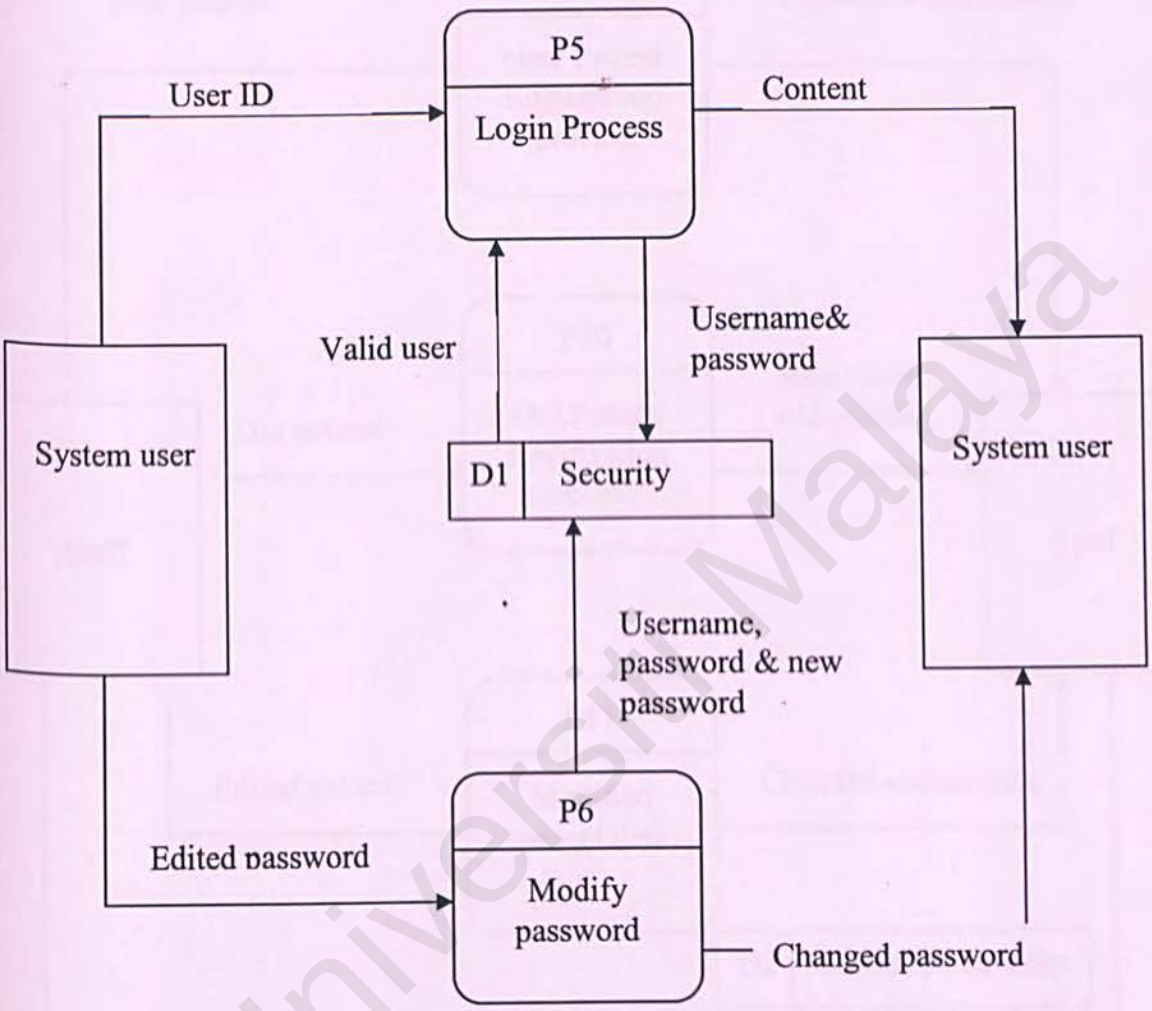


Figure 5-10: O Level Data Flow Diagram of Staff for CCIS.

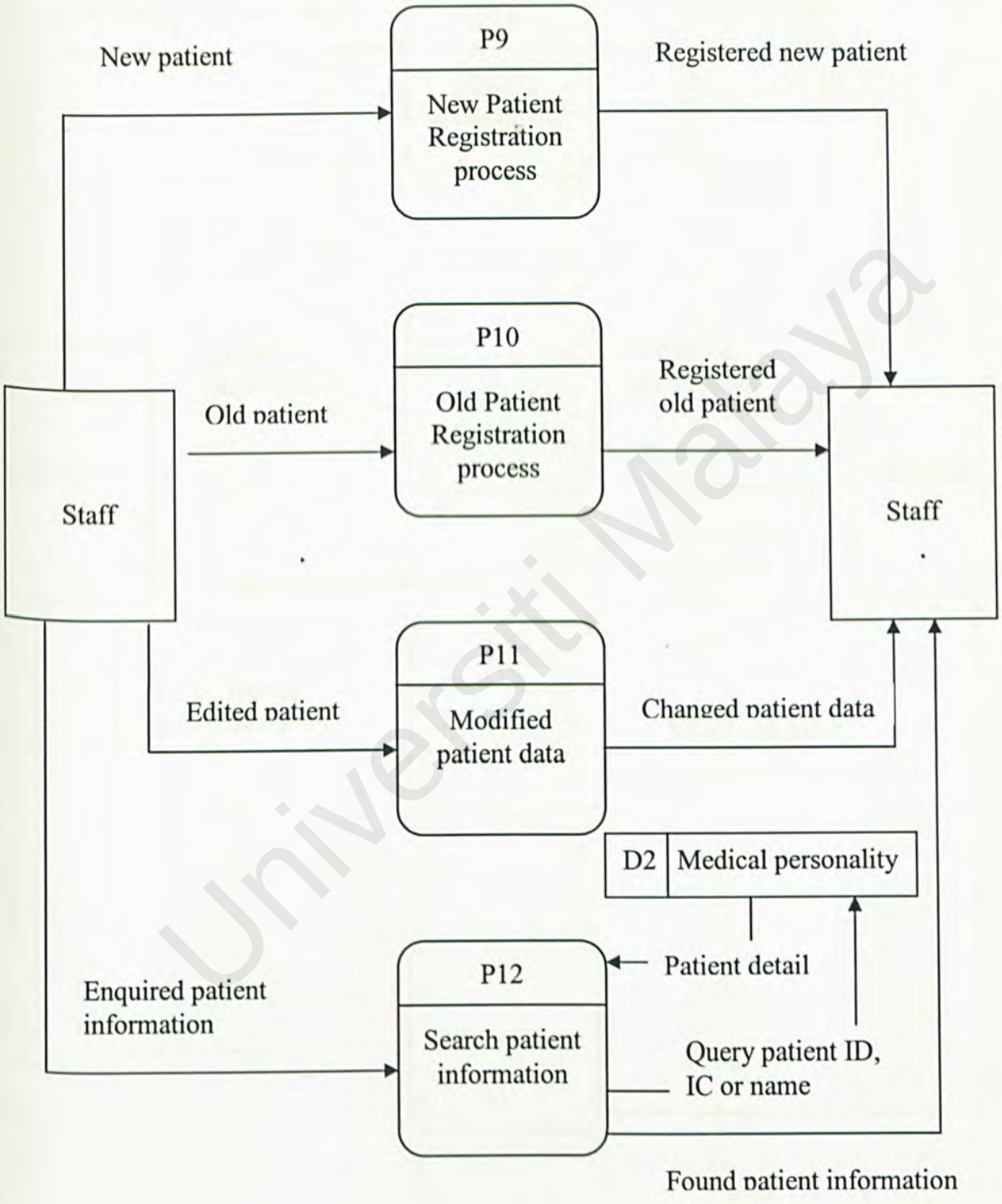


Figure 5-11: O Level Data Flow Diagram of Physician for CCS.

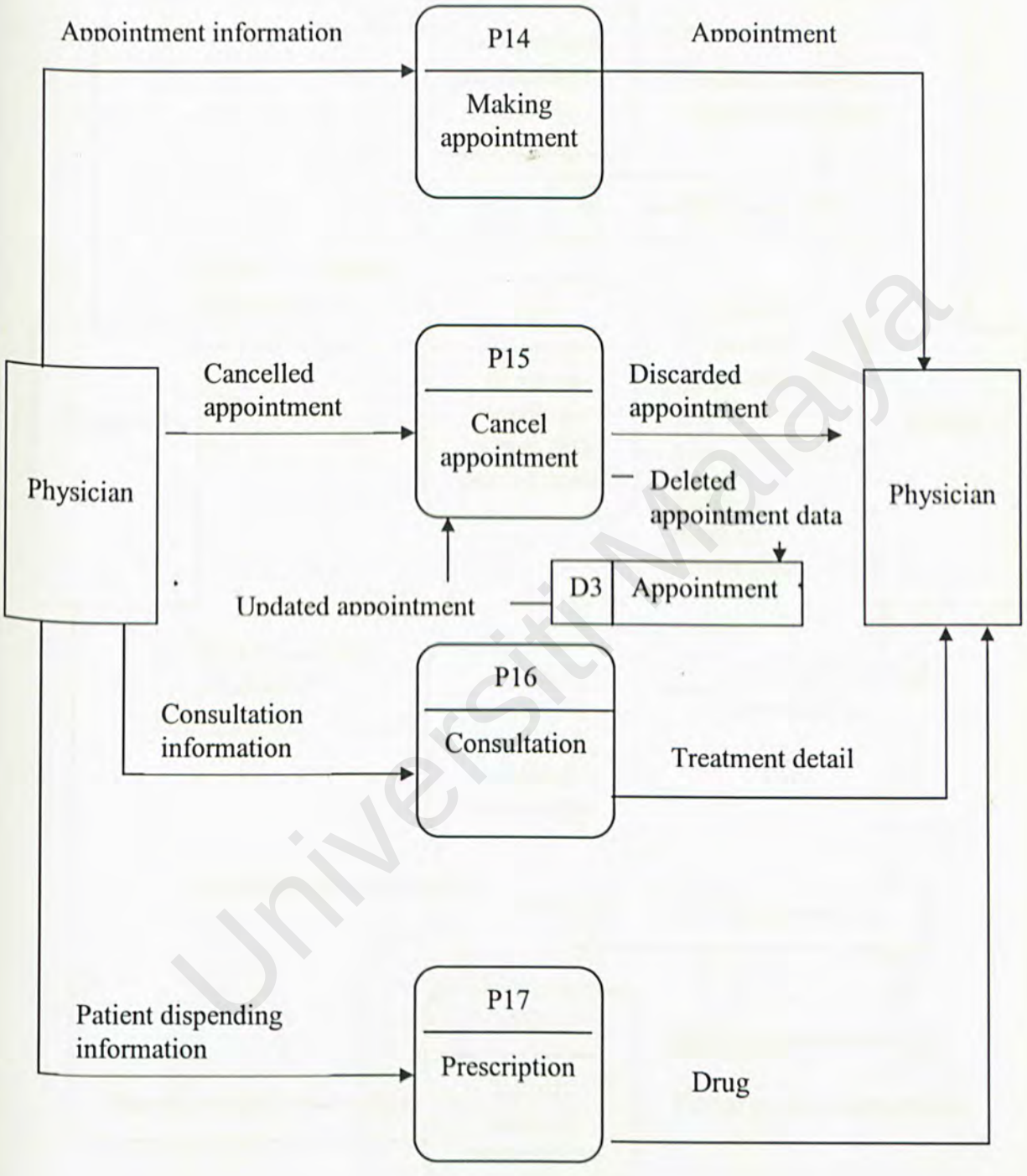


Figure 5-14: O Level Data Flow Diagram of Management for CCS.

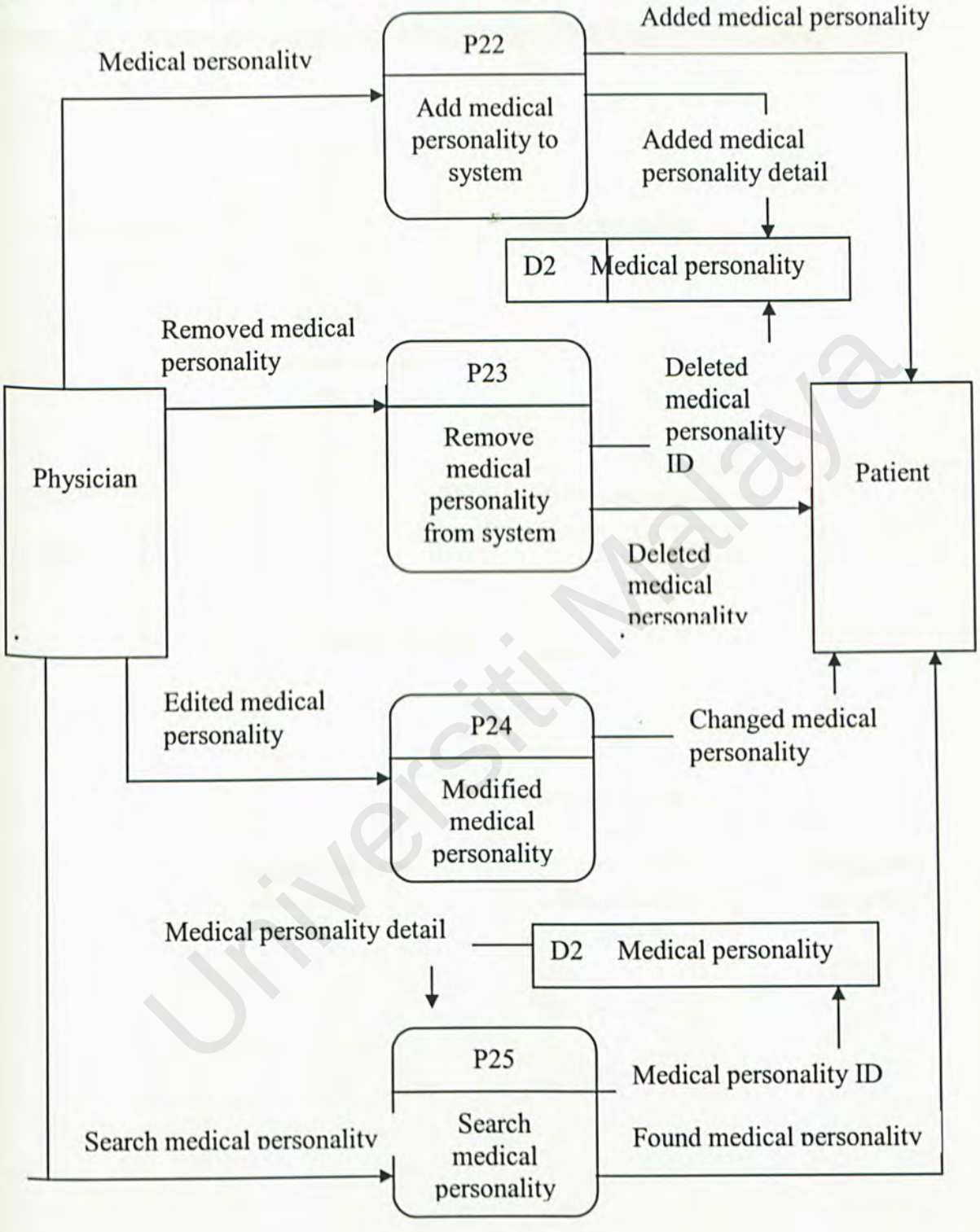


Figure 5-15: First Level Data Flow Diagram for New Patient Registration (P9).

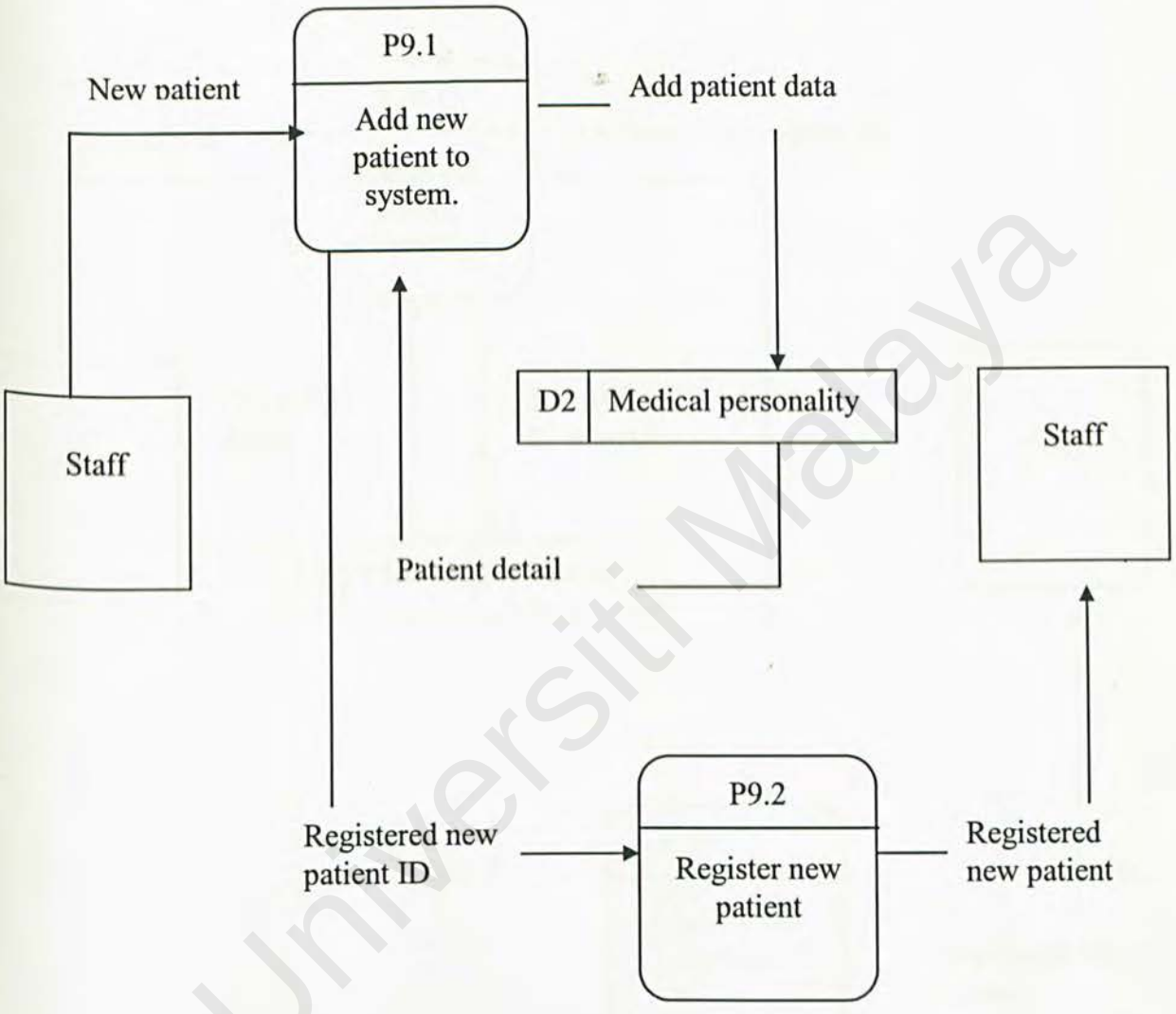


Figure 5-16: First Level Data Floe Diagram for Old Patient Registration (P10).

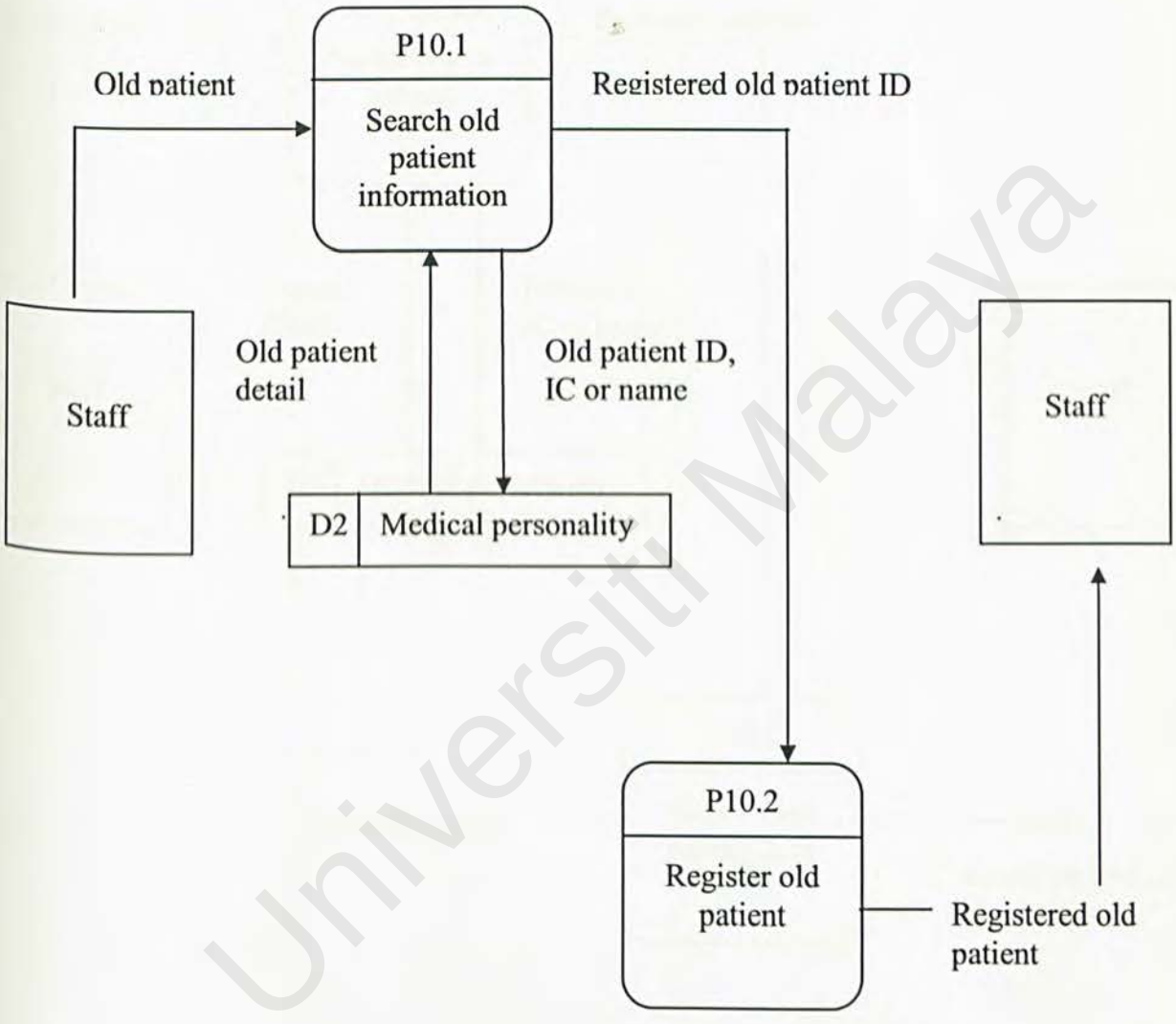


Figure 5-17: First Level Data Flow Diagram for Modified Patient Data (P12).

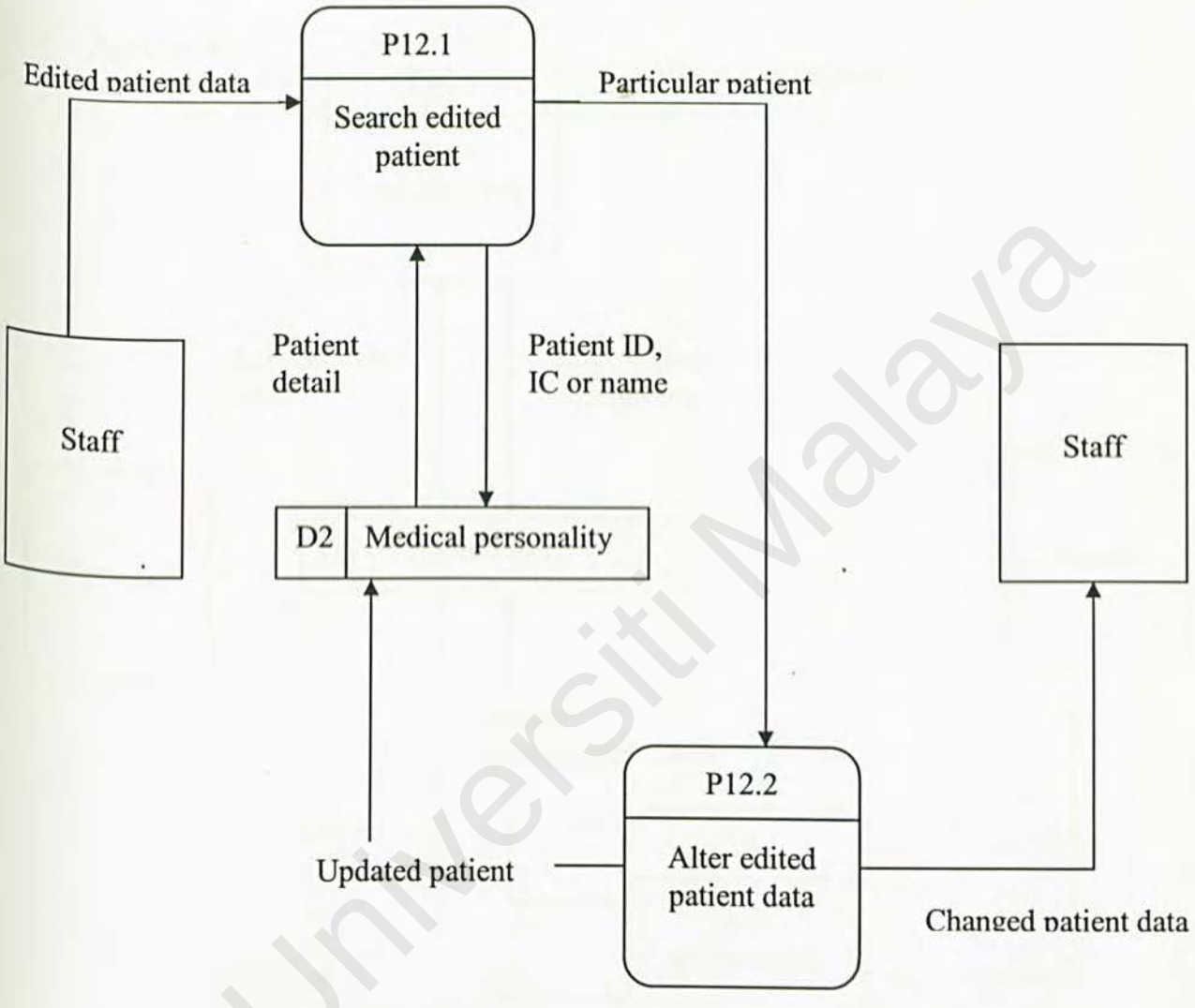


Figure 5-18: First Level Data Flow Diagram for Making Appointment (P14).

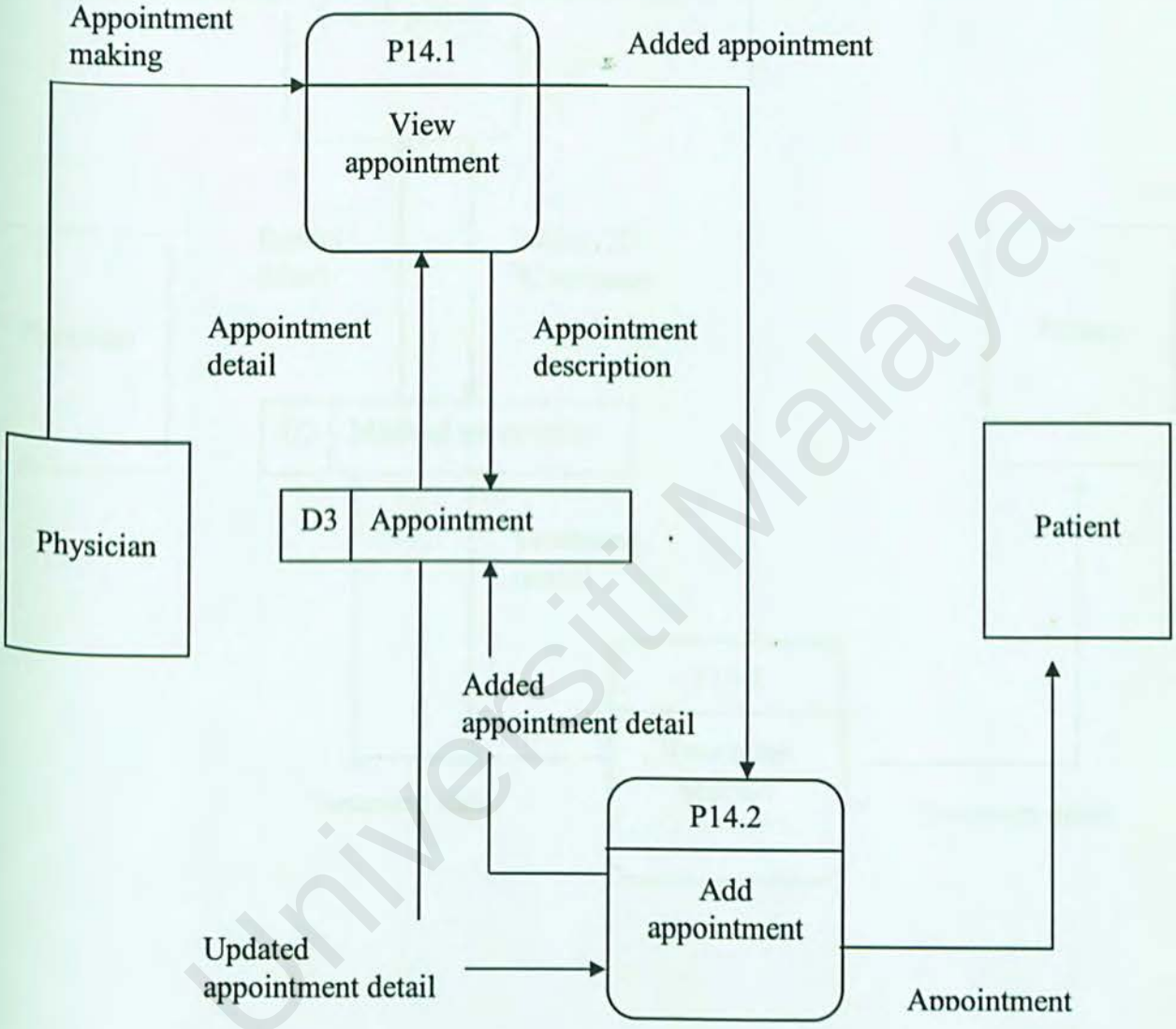


Figure 5-19: First Level Data Flow Diagram For Consultation Process (P16).

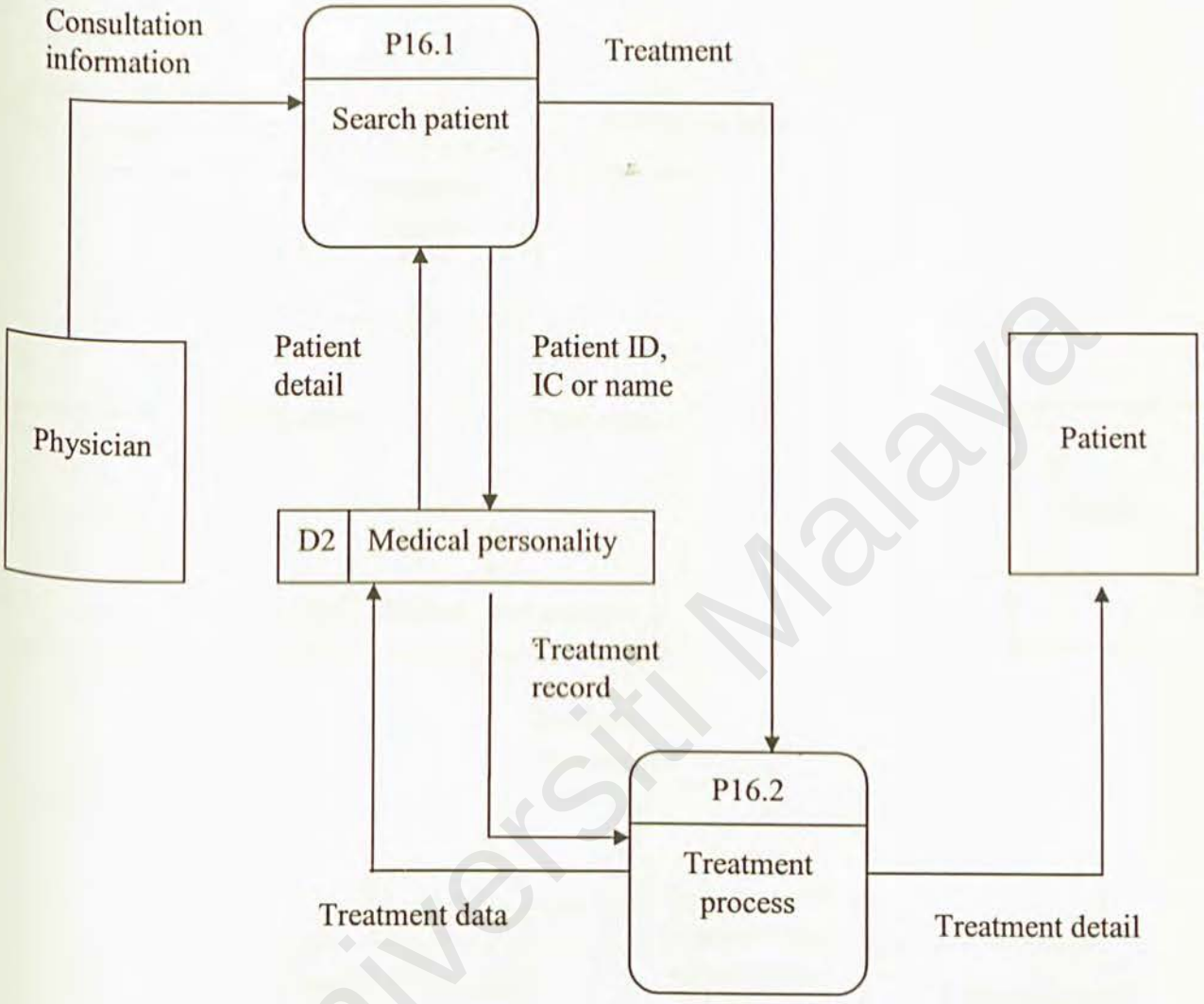


Figure 5-20: First Level Data Flow Diagram for Prescription (P17).

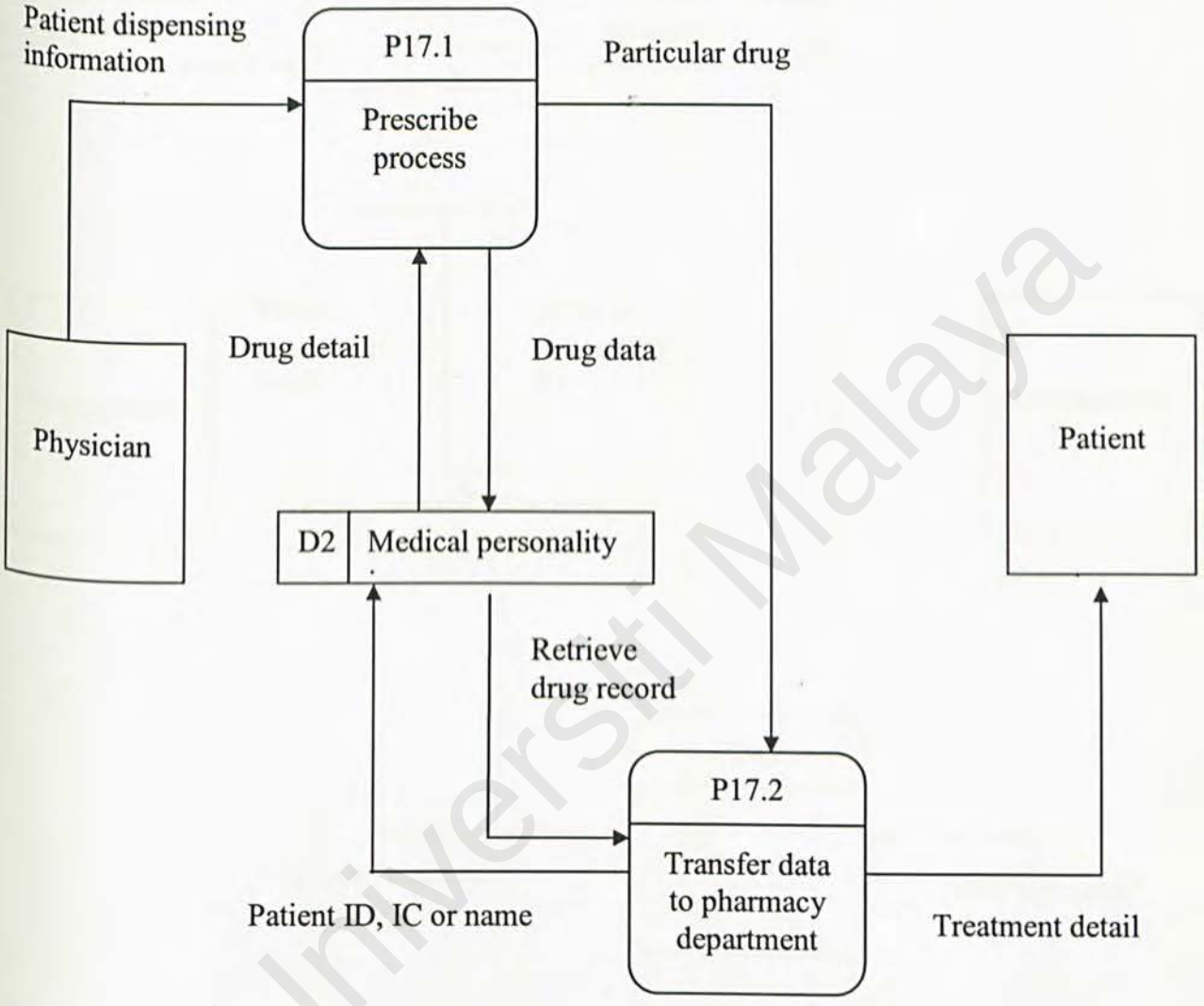
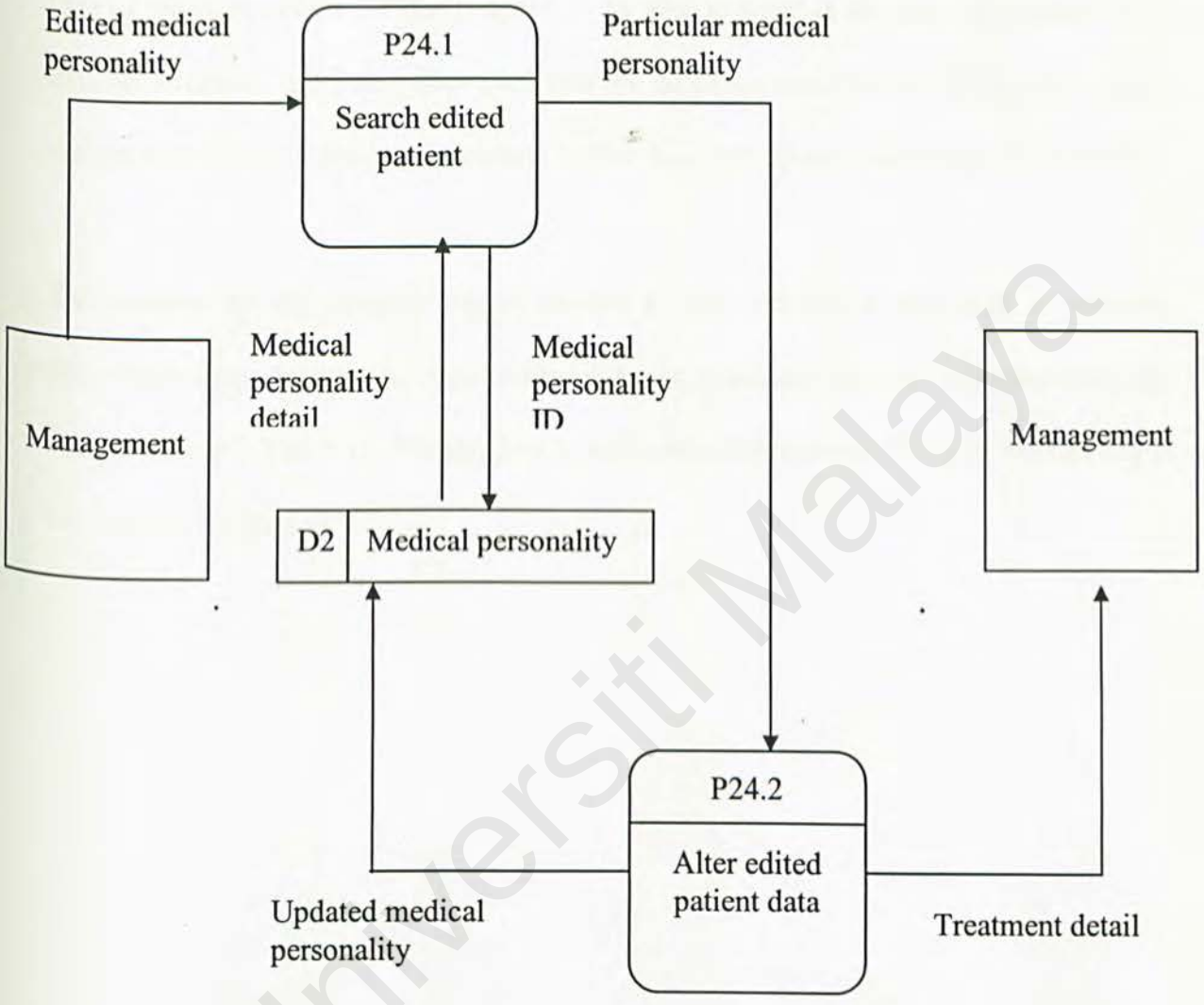


Figure 5-22: First Level Data Flow Diagram for Modified Personality (P24).



5.2 DATABASE DESIGN

One of the components for this program to be able to work is the data requirement. The data requirements are being converted into the database specification during the system design stage. For this project, Microsoft Access has been chosen to develop the database.

The database for this program can be divided to four data stores files, such as security file, medical personality file, appointment file, and pharmacy file. All this data stores are show in Table 5-2 to 5-12. Primary key is represented by symbol ** while foreign key is represented by symbol *.

5.2.1 Security File (D1)

The following table is illustrated security files of this project.

Field Name	Data Type	Size	Description
* Staff_ID	Char	4	Unique code generated by SQL for staff created.
User_name	Char	15	Name used to represent an identity of a person, such as ccis2002.
Password	Char	8	Security code that needed while a user login to system.

Table 5-2: Table of System User File.

5.2.2 Medical Personality File (D2)

The Table 5-3 to 5-7 show the database design for medical personality file.

Filed Name	Data Type	Size	Description
* Patient _ID	Char	6	Unique number generated by SQL for patient created.
Patient_name	Char	50	Patient's name.
Old_IC_no	Char	14	Patient's old identity card number.
New_IC_no	Char	14	Patient's new identity card number.
Birthday	datetime	8	Patient's birthday.
Address	Char	50	Patient's address.
Postcode	Char	6	Post code.
City	Char	20	City.
State	Char	20	State.
Country	Char	20	Country.
Home_Tel	Char	15	Home telephone.
Mobile_Phone	Char	15	Mobile phone number.
Age	Smallint	3	Patient's age.
Race	Char	20	Patient's race.
Sex	Char	6	Gender.
Remark	Char	50	The additional text or remarks for specified patient.

Table 5-3: Table of Patient Personal Information.

Filed Name	Data Type	Size	Description
*Admission_no	Char	6	Unique number for patient's Admission number
*Patient_ID	Char	6	Unique number generated by SQL for patient created.

Table 5-4: Table of Treatment_Xref.

Field Name	DataTyoe	Size	Description
**Admission_no	Char	6	Unique number for patient's Admission number.
Disease	Char	50	Disease infected by patient.
Admission_date	datetime	8	Date or time the patient is given by the clinic.
Infected_date	datetime	8	Date or time the patient is infected by disease.
Remarks	Char	50	The additional remarks.

Table 5-5: Table of Patient's Treatment.

Field Name	Data Type	Size	Description
*Pharmacist	Char	50	Person who given drug to patient.
Drug_name	Char	50	The drug's name.
Unit	Char	15	The unit of measurement of the drug code.
Quantity	Int	4	Quantity used of drug code.
Price	Char	50	The price for the drug code.

Table 5-6: Table of Patient Dispensing.

Field Name	Data Type	Size	Description
**Patient_ID	Char	6	Unique number generated by SQL for patient created.
Date	datetime	8	Date when the allergies is observed.
Reaction	Char	10	Reaction to the drug allergy.

Table 5-7: Table of Patient's drug Allergy.

5.2.3 Appintment File (D3)

Table 5-9 to5-10 illustrated the database design for appointment file.

Flied Name	Data Type	Size	Description
*Patient_ID	Char	6	Unique number generated by SQL for patient created.
*App_no	Char	6	Unique code generated by SQL for application code created.

Table 5-9: Table of Appointment _Xref.

Field Name	Data Type	Size	Description
**App_no	Char	6	Unique code generated by SQL for application code created.
*Staff _ID	Char	4	The staff's ID who making appointment with patient.
Date	datetime	8	Specified the date for appointment.
From_time	datetime	8	From what time the application wish to start.
To_time	datetime	8	To what time the appointment wish to end.
Remarks	Char	30	Any additional remarks.

Table 5-10: Table of Appointment Booking.

5.3 INTERFACE DESIGN

User interface design is an another parts of system design. A good interface design will help user to understand on the system even faster. However, user interface design is tricky things to design because different people have a different style of perception.

In this project, the user interface is designed to have a good and user friendly interface.

5.3.1 User Interface for Clinic Care Information System

The following show some interfaces of the Clinic Care information system (CCS) project from Figure 5-23 to 5-27:

Figure 5-23 shows the main layout of the CCS.

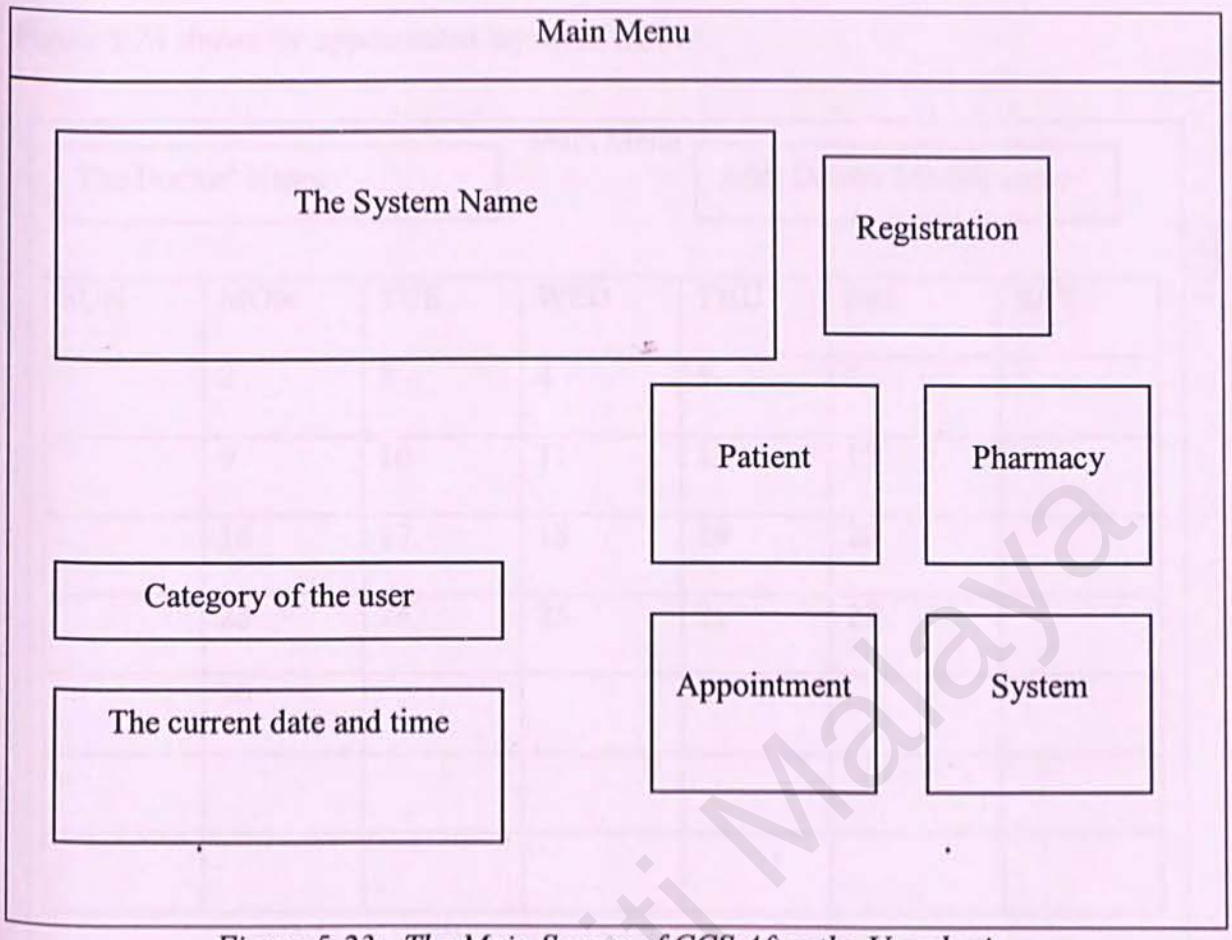


Figure 5-23: The Main Screen of CCS After the User login.

Figure 5-24 shows the appointment layout of CCS.

Main Menu

The Doctor' Name

Add/ Delete/ Modify icons

SUN	MON	TUE	WED	THU	FRI	SAT
	2	3	4	5	6	
	9	10	11	12	13	
	16	17	18	19	20	
	23	24	25	26	27	
	30					

Figure 5-24: The Appointment Layout of CCS.

Figure 5-25 is the layout for staff to carry out the registration for the new patient.

Main Menu: New Patient Information

Name	IC Number
DOB	Gender
Race	Marital Status
Blood Type	Nationality
Occupation	

Home Address:			
State		Telephone No (H)	
City		Telephone No (M)	
Post Code			
Country			

Figure 5-25: The New Patient Registration Layout Of CCIS.

The following layout is the patient history records.

Main Menu

Patient Information

NO	Disease Name	Drug Name	Quantity of Drug	Doctor Name	

Text box

Figure 5-26: The Patient History Record Layout of CCIS.

5.4 CHAPTER SUMMARY

The system design of the project is divided to three parts, such as functionality design, database design, and interface design. In functionality design phase, the structure chart and Data Flow Diagram have been chosen to represent the CCS's functionality requirements. Each of the functionality requirements of the system has been decomposed into simpler modules during the functionality design. It is enabled user or developer to get much more understanding of the system.

In the database design, the data requirements are being converted into database specification.

In the end of this chapter is the interface design. The user interfaces of this project are designed based on the sub modules of the project.

6. SYSTEM IMPLEMENTATION

6.1 Introduction

System Implementation is the continuous stage to the previous system design. The implementation will focus on the process of developing the programs. Programs design and implementation are needed for a numbers of reasons in a system. It will determine the data transfer correctly and to make sire that the system will work accordingly to the users expectation.

Writing program code is one of the big steps in system implementation. In transform the idea from the system design to program code. At the same time, it will implement the idea to the workspace and platform in order to produce what we call the pilot version or prototype of a product.

6.2 Coding Approach

Coding is an iterative process whereby it is done until the programmer obtains the desired output. There are two types of coding approach, one is top-down and the other one is bottom-up.

The top-down approach shows the higher-level modules to be coded first before the lower level modules. The codes in the lower in the lower modules contain only an entry and an exit. A module with such characteristic called a shell. The higher- level modules

will reference the lower ones if they are coded and available. This approach will ensure that type must important modules will be developed and tested. It also gives a preliminary version of the system soon.

The bottom-up coding is based on coding some complete lower lever modules and leaving the high level modules merely as skeletons that are used to call the lower modules, whereas the top-down approach is the reverse.

For this system, coding is done with the bottom-up approach. The advantages of this approach are: testing can be carried out on some of the functions as soon as it is completed, and critical functions can be coded first to test their efficiency.

6.3 Coding Tool

This system is developed using Microsoft Visual Basic 6.0 and Microsoft Access. Chapter 2 provides more explanation of them.

6.4 Program Documentation

Program documentation is a set of written descriptions that explain to a reader what the programs do and how they do it [Pfleeger, 1998].

6.4.1 Naming Convention

Naming convention is an abbreviation of the control name or the object name. This system uses a naming convention to ensure uniformity of the control and object names. The purpose of this naming convention is to increase the readability of the codes. Table 6.1 shows the example of the naming convention.

Control	Control Name	Example
Form	Frm	FrmQueue
Command Button	Cmd	CmdAdd
Text Box	Txt	TxtPatientID

Table 6-1: Table Of Naming Convention

6.4.2 Internal Documentation

Internal Documentation is a description material written directly within the code [Pfleeger, 1998]. It means that internal documentation refers to comments within the codes. This is needed to enhance readability of the code by someone other than the programmer. It will also help the programmer to recall the function or meaning pf the certain codes. For Visual Basic, the comment tag is illustrated below:

' *This is a comment*

Any text after this symbol ' will be ignored during execution time. All the comments are in green colour. Codes are also formatted to enhanced understanding. Spacing or line break in between different section of the codes will enhance readability.

6.5 Coding Specification

The clinic management system is divided into 6 main modules, which are:

i) Security

ii) Pharmacy

iii) Registration

iv) Appointment

v) Consultation

vii) Report

Each module is developed using Visual Basic 6.0.

6.6 Data Validation

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

6.7 User Interface

This system is user friendly with is simple forms and uniformity to enhance readability and ease of use.

6.8 Chapter Summary

The clinic management system is developed using Visual Basic 6.0 and Microsoft Access for Clinic Care System (CCS). Coding is done with the bottom-up approach. While the design model of CCS is being into a workable product. Stages involved are coding and modules implementation. In next chapter, the testing of the final product is highlighted. It will show some testing approaches used. There are coding example shows in *Appendix A*.

7. SYSTEM TESTING

System testing is a essential to ensure the system performs according to its specifications and in line with users' requirements as well as expectations.

Testing is done throughout system development, not just at the end of development stage. It is meant to turn up unknown problems, not to demonstrate the perfection of the programs.

Testing is performed to detected the existence faults and correct it. Therefore, a systematically test produce is needed to make sure the system is tested thoroughly and completely. There are several types of testing. There are Unit Testing, Sub Module and Module Testing, Sub System Testing and System Testing.

7.1 Unit Testing

Individual components are tested to ensure that stand-alone program fixes the bug without side or internal effects. After a new component is developed, it is tested independently, without other systems components. This is to assure that the component is able work accurately and persistently. All function on each button is examine to ensure it perform the entitle output such as connection form page to page, call the right function to execute, display the right message accordingly to the error and eliminates all the syntax faults occurred.

For example, the Resconfirmed(), this function has to test by ordering to see whether the current date is based on Month/ Day/ Year format. The date format that is required from user must match the database format. There is a coding of the example shows in *Appendix A*.

7.2 Module Testing

Module Testing will focus on each sub module in Clinic Care System (CCS). There are totally 14 sub modules in CCS. Therefore, testing was carried out to ensure that the codes under the sub modules work accordingly when all unit of code are integrated.

Each of the sub modules is tested with the specific functions to identify whether they really output the results of desired design requirements. For example, the AddPatient sub module is tested to ensure all the text boxes have been filled up and generated the new PatientID and save all the information to the database.

If an error is occurred from a particular sub module, the part of the sub module could be identified and unit testing is used to examine the errors.

After that, all sub modules group into module to perform the testing. This includes the doctor, staff and pharmacist. This is to test whether all sub modules can work together as a module.

7.3 Integration Testing

When the module testing has achieved certain degree of requirements objectives, the sub modules are combined into a working system. Integration testing is used to test for the combination of the two independent sub modules that are working together.

In this phrase, all the processes are involves in CCS.

7.3.1 Bottom-Up Integration

Bottom-up testing approach was chosen for the Integration Testing. Each component or unit at the lower level of the system is tested individually. Then go to higher level of the system.

7.4 System Testing

System Testing is difference from unit testing and integration system. System testing is ultimate testing procedure. It studies all the concerns issue and behavior that can only be exposed by testing the entire integrated system or major part of it. The testing process is also concerned with validating the system meets its functional and non-functional requirements. The following testing description has been carried out:

- Staff has successfully registered the patient information.

- After registration has been done, the patient will be put in waiting queue in order to get doctor consultation.
- Patient can make making appointment via telephone or face to face.
- Doctor or physician will retrieve patient record from Staff department and waiting queue is based on first in first serve.
- Doctor enters the consultation result of patient to database and add the information, such as drug and price to pharmacy department.
- Pharmacist retrieves the information and dispenses the drug to patient according the doctor's description.
- Finally, payment is made in pharmacy department.

7.5 Chapter Summary

From the testing process that has been carried out, it can summarize the test result as follow:

- **Achieve the main objective of the project**

Generally, the main objectives of the project as described earlier have been achieved. This system can handle and maintain the activities in small clinic.

- **Syntax of language**

The language uses in this system is not too technical. Unlike the programmer, the user of CCS is expected have no computer knowledge, so they should not be able to understand computer syntax as programmer does.

8. SYSTEM EVALUATION

8.1 Introduction

After design and developing processes as well as implementing CCS, the end product of the project is brought up for evaluation. They are many evaluation techniques that can be used to evaluate the final system. The following section will explain in detail about the system strength and its limitation.

8.2 Problem Faced and Solution

The following are the major problems encountered from the beginning of the project through the end of the system.

- **Difficulties in gathering information**

Not much information could be gathered from interview and clinic system is not common in Malaysia.

Solution: Obtain more information and idea through Internet

- **Date format**

The date format of CCS system was DD/MM/YY, but retrieving record from database with this date format will get wrong result.

Solution: Date format has been changed to MM/DD/YY.

- **No end user evaluation**

End user response and feedback of this system cannot be obtained.

Solution: Feedback from course mate.

- **Slow processing time**

Visual Basic 6.0 is a graphical-oriented programming language. It required more memory to compile and execute the application.

Solution: The minimum computer memory requirement in running CCS is 64 MB as it is documented, but a higher memory is much preferred.

8.3 System Strengths

This system is evaluated orderly as following:

- **Security**

This system is implemented with security login. *Userid* required to login before using the system. Only the authorized user can login to the system. System is using valid *userid* and *password*.

For security purpose, users are assigned to different group. There are 4 groups of users: administrator, doctor, staff and pharmacist. Different groups of the users are allocated certain access right to the functions in CCS. Therefore, this system prevent unauthorized accessibility will be reduced.

- **Simple and user friendly interface**

CCS is developed by Visual Basic that provides suitable GUI development tool. Therefore, all forms are kept simple. This is to provide user-friendly system to user for easy to pick it up.

- **Searching capability**

A good information retrieval system is designed. For example, this system allows user to search to database record by suing *PatientID*, or *IC No*.

- **Display process messages**

There are a lot processes between the system and its database. Therefore, it is important to inform user that every action taken will display messages to inform the user.

For example:" Data is successfully added." will be displayed for user knowledge. It can avoid user for adding two similar data into database.

- **Incorporates data validation**

All the fields in form will be checked for null value or invalid data type. Error when inserting record into database will not occur. Besides that, error message will also be prompted to the user.

- **Fast response for information retrieval**

This system provides fast response to record searching and displaying in a report for the user.

- **Implement error handling**

To avoid run time error, this system is developed with error handling. Error message will be displayed when system encounters exceptions and it will not terminate suddenly.

8.4 System Constraints

There are a few system constraints and all of them are discussed below:

- **Slow response time**

If there are a lot of records in the database, the security process will take more time because the system will search the database and retrieve the search result in response to user.

- **Limit categories if user**

There are limit categories of user. The system user can be divided into 4 groups.

- **No drug inventory database provided**

Pharmacist from dispensing department has to find alternative way to manage their drug inventory.

- **No complete report provided**

CCS only can generate a patient medical report.

8.5 Future Enhancements

Some functionality of CCS can be enhanced in order to improve the quality of the system. The following are the suggestions and possible future enhancements:

- **Cover all the functionality to provide end to end service**

To really benefit all the users, CCS shall be modified to covers all functionality to provide better services. For example, intranet mailing module.

- **Generate bar code function**

If the system could generate bar code either for system user or patient. It will be saved time. For example, the medical staff can search patient information by scan the card with the bar code. It is much faster than searching record by entering data.

- **Further classified the user's group**

According to their access right, CCS shall be modified to classify the group of users based on their access right. The system will become more flexible.

- **Further enhancement of interface design**

CCS shall be modified to enhance the interface. So the system become more attractive and user friendly.

- **Provide help function**

If system could provide help instructions, it will assist the users to learn up the system easily.

8.6 Knowledge and Experiences Gained

From the beginning of this project until the final documentation, a number of problems are occurred and experiences are learned from there. There are:

- **The importance of all phases in System Development Cycle Life (SDLC)**

System Analysis is an important phase in System Development Cycle Life. This phase captures user requirements and the goal of the system. If this phase is wrong defined, it will cause faulty to the system development process.

System testing is also an important phase in SDLC. With the procedure, errors and faults in the system can be minimized. The functionality of each module or form can also be tested and confirmed that it meets the user requirements.

- **Development tool knowledge**

This project is using Microsoft Visual Basic 6.0 and Microsoft Access as a development tools.

- **Self Expression**

Developing CCS has really given an opportunity to express myself in designing and developing of the system.

8.7 Chapter Summary

Evaluation of a system is indeed needed to ensure its objectives and intended functions have been achieved. This chapter covers all the aspect of evaluating application software. At the end of evaluation, encloses the conclusion of this thesis project.

8.8 Project Conclusion

Overall, CCS has achieved and fulfilled the objectives and requirements as an Information Management System. This system will help medical staff to simplify day-to-day task in a modern clinic. At the same time, it provides management with the necessary information to make strategic decisions.

There was a lot experiences gained throughout the development of this system. This includes knowledge in using database application, programming and concept as well as manner of project management.

Finally, CCS can be done as it planned earlier. With the first step taken, enhancements can still be made with more features added for future version. Thank you to my supervisor, moderator and everyone have assisted me to the final of CCS.

References

1. Whitten, Jeffrey L. (2000). *System Analysis and Design Methods*. Fifth Edition. McGraw-Hill Irwin.

2. Nieto. T.R. (1999). *Visual Basic 6.0: How To Program*. Prentice-Hall Inc. New Jersey, USA.

3. William Stallings. (1997). *Data and Computer Communication*. Fifth Edition. Prentice-Hall Inc. New Jersey, USA.

4. *Ida M. Flynn. (1996). Understanding* Operating System. Second Edition. PWS Publishing Company.

5. Salami, Joe. (1995*). Guide To Client/ Server Databases*. Second Edition. Ziff-Davis Press, California.

6. Traditional Chinese Medication Clinic Management System.
   Retrieved July 3, 2002 from the World Wide Web.
   http://www.is.cityu.edu.hk/fyp/project2000/chimedic.htm

7. HealthCare Information Management System.
   Retrieved July 3, 2002 from the World Wide Web.
   http://www.zfour.com/Medical_Systems/medical_systems.html

8. e-Clinic Management System.
   Retrieved July 3, 2002 form the World Wide Web.
   http://www.ezyone.com.my/produce_eclinic.htm

9. Clinic Management System 2000 (CMS2000).
   Retrieved July 3, 2002 from the World Wide Web.
   http://www.healthwho.com/cms2000/cms2000.cfm

10. Free Vet – An Animal Clinic Management System.
    Retrieved July 3, 2002 from the World Wide Web.
    http://www-med-chemie.vu-wien.ac.at/mirrors/FreeVet/

11. Database Management System.
Retrieved July 4, 2002 from the World Wide Web.
http://www.microsoft.com/data/oledb/

12. Database Management System (DBMS).
Retrieved July 4, 2002 from the World Wide Web.
http://www.webopedia.com/TERM/d/database_management_system_DBMS.html

13. ODCB
Retrieved July 4, 2002 from the World Wide Web.
http://www.webopedia.com/TERM/O/ODBC.html

14. OLE DB or ODBC
Retrieved July 4, 2002 from the World Wide Web.
http://www.sqlmag.com/Articles/Index.cfm?ArticleID=6197

15. Client/ Server Architecture.
Retrieved July 5,2002 from the World Wide Web.
http://www.webopedia.com/TERM/C/client_server_architecture.html

16. Client/ Server Software Architecture.
Retrieved July 5, 2002 from the World Wide Web.
http://www.sei.cmu.edu/str/descriptions/clientserver_body.html

17. SQL-Server-Performance.
Retrieved July 5, 2002 from the World Wide Web.
http://www.sql-server-performance.com/

18. Microsoft SQL Server.
Retrieved July 5, 2002 from the World Wide Web.
http://databases.about.com/cs/sqlserver/