FACULTY OF COMPUTER SCIENCE AND INFORMATION TECHNOLOGY

SESSION 2004/2005

TITLE : HELP DESK FCSIT

PREPARED BY:

AMINAH BINTI ABD HALIM
WET020004

SUPERVISOR : MRS SURAYA HAMID

MODERATOR : MRS NAZEAN JOMHARI

Abstract

Help Desk FCSIT is a web-based complaint, suggestion and feedback system for Faculty of Computer Science and Information Technology, University of Malaya. The aim of the system is to help the targeted users in doing and managing complaint, suggestion and feedback and therefore replace the massive manual system. Help Desk FCSIT is designed to integrate the strength of the current system with new features that will provide better services to its intended user. Generally, Help Desk FCSIT has three category of target user. Help Desk FCSIT contains eight sections that are Authentication, Complaint, Personalized Information, Reply, Search, Statistics, Committee Maintenance and FAQ. The waterfall model approach is selected for the development of Help Desk FCSIT. This method is adopted because simple and ease of implementing. In addition, testing is performed in every stage and may go back to the previous stage to correct any errors. Through the system development life cycle, system methodology is adopted to understand the current problem situation. Careful analysis and research has been conducted to determine the feasibility of the system and what is required of it. The Help Desk FCSIT will be program using Active Server Pages technology with JavaScript and VBScript as the scripting language and developed on Microsoft Windows XP Professional platform. The development tool is Macromedia Dreamweaver MX, whereas the web server is Microsoft Internet Information Server 5.1 with the back-end database Microsoft SQL Server 2000. Hence, Help Desk FCSIT is designed to be three-tier architecture provide greater overall system flexibility.

Acknowledgement

The success completion of this project is related to the contributions of many people. I would like to take this opportunity to thank some of the key people now.

First and foremost, I would like to express my highest gratitude to my supervisor, Puan Suraya Hamid for her priceless advices, guidance, encouragement and support along the project.

Special thanks are also due to Puan Nazean Jomhari as the project moderator for her precious time to attend the Viva presentations and contributed constructive advice; suggestion and ideas to further enhance value of this project.

Not forgotten also, I would like to take this opportunity to express my admiration and appreciation to my beloved parents, Encik Abd Halim Abd Majid and Puan Asiah Buang and also to my dearest siblings for their valuable supports and encouragements throughout the project.

I would like to express my sincere thanks to my friend, Shaiful Shahreen who have helped me and given support and assistance to me throughout the project. Last but not least, a special word of thanks is reserved to my course mates and friends – Kak Lin, Wan, Rozet, Nor, Su - for their moral support and endless motivation throughout the compilation of this project. Thank you.

Table of Contents

List of Figures

List of Tables

CHAPTER 1 :
INTRODUCTION

CHAPTER 1 : INTRODUCTION

1.1 Project Overview

Essentially, the Help Desk is a central point through which problems or issues are reported and subsequently managed or coordinated. From a wider perspective, it is also seen as core part of the service function, responsible for bringing together multiple resources to address an issue. [4]

Help desks are now a fundamental and core part of good business services and operations. The main reason for using the help desk is to maintain and improve the quality of services of an organization. Subsequently, in order to improve the services provided by the organization, the users must be allowed to give comment, suggestion or feedback. Consequently, the organization will be aware of any difficulties and necessities in their services. Therefore, technology should be adopted to help the organization to achieve their goals. With the growth of Internet, the help desk system can be access online and available 24 hours.

"Help Desk FCSIT" is a web-based complaint, suggestion and feedback management system designed for Faculty of Computer Science and Information Technology (FCSIT), University of Malaya. Every member of FCSIT, from staff to student will be able to use this system. The purpose of this project is to develop and create a web-based system for making and receiving complaint, suggestion and also feedback on the services provided by FCSIT. It will help the administrator and the users, to manage and making the complaint, suggestion and feedback online, respectively.

Moreover, this system will minimize the time to manage and make complaint, suggestion and feedback manually.

This system will be very interactive where the users can check the status of their complaint, suggestion and feedback online. Help Desk FCSIT can be accessed from faculty's web site. This system is more efficient and effective than the manual system.

1.2 Problem Statement

The traditional manual system consists of abundant of weaknesses. Some of the weaknesses faced by the targeted users during making complaint, suggestion and feedback by using the manual system are as below:

i) Paper based or current manual system

The current manual system involves the usage of paper, where the students are required to fill in a form in making complaints, suggestions or feedbacks. The management of the forms and data retrieval requires a lot of time and human resources. Moreover, it requires special places for data storage and usually, it involves a lot of space. More often, the files maybe damaged or lost because of the lack of awareness and these conditions make data retrieval become more difficult.

ii) Complicated manual procedure

Manual procedures are very time consuming to the management, students and staffs. Currently, the management still using filing system. There is no online system to make complaint, suggestion and feedback. They need to fill up the complaint form manually. It takes a lot of time not only for those who wanted to make a complaint, suggestion and feedback but also for the administrator who handle the cases. Many staffs involved in handling a complaint, where they have to pick up the form from the box manually everyday and need to sort it accordingly. It also very difficult and time consuming when they have to write a final report every month.

iii) Lack of human resource

Many staffs are needed to handle the cases. The lack of human resources may cause the manual procedure become worse.

iv) Lost of form

When using the manual system, there is probability to lose the complaint form. This is because of the misplaced, or damages.

v) Less information

There are two suggestion boxes allocated for the students to make complaints, suggestions or feedbacks. One is located at the foyer and the other one is located in front of faculty main office. However, most of the students do not aware of the presence of suggestion boxes around the FCSIT. However, because of lack of information on the right channel to make complaints, suggestions or feedbacks,

the students usually just kept the problems within themselves until the condition get worsen. Therefore, if there is online system, while they surfing the FCSIT website, they will aware of the link to the Help Desk FCSIT.

1.3 Project Aim

The aim of this project is to provide a web-based help desk for FSCIT. The purpose of web-based help desk is to help the targeted users in doing and managing complaint, suggestion and feedback and therefore replace the massive manual system.

1.4 Project Objectives

In order to develop this system, the objectives must be well understood to ensure that the outcome meets the objectives. The main purpose of this project is to develop a web-based system that solves the above manual problem. Others core objectives are:

i) Replaced existing manual system

With this system, the targeted users can use this system online. This system will reduce paper processing, which will also be costly from the perspective of paper wasting or storage of the printed forms and reports. Paper will involve only when the administrator want to print out the reports they needed. This online system will avoid managing the complaint, suggestion and feedback manually.

ii) Increase quality and accuracy

The system is build in with database; therefore the accuracy of the data is very high. In order to increase the quality of management, this system aims at monitoring every activity of complaint, suggestion and feedback.

iii) Easy to use

This system provides a nice and interactive user-friendly interface. The design of user interface is visually pleasing and easy to navigate, which can accommodate all types of users. No command or programming skill is needed to access the system. Users can handle the system very easily as simple as clicking on buttons by using the mouse only.

iv) Help the target user

This system will help the target users to make a complaint, suggestion or feedback online and therefore save their times. They will also receive the confirmation and may look for the status of their complaint, suggestion or feedback. For the administrator, they do not need to collect the form from the suggestion box. They can keep track of the record online and can sort it automatically. This will make the job easier.

v) Reduce human resources

With this system, the number of staff involves to handle the cases can be reduced. This will help the organization to reduce the cost of recruiting on new staffs.

1.5 Project Scope

The Help Desk FCSIT is developed to all members of the faculty. The scope of the project covers two categories that is language and target user.

1.5.1 Language

This system is using English as medium of communication. The usage of English as the communication language will allow vast utilization of the help desk as foreigner students will also be able to access it. Furthermore, the language is also widely use and known.

1.5.2 Target User

This system has three main target users. They are Administrator, Committee and Registered User.

i) Administrator

This type of user has the highest authority to the system. Administrator is the person who will control the e-mail system and also database system. Administrator is responsible to manage the committee and FAQ section.

ii) Committee

Committee is the person or group that involved in settle down the complaint, suggestion or feedback related to the respective category. The committee can reply, update and delete the complaint, suggestion or feedback.

iii) Registered User

This type of user consists of two categories, Staff and Student of FCSIT. All members of FCSIT will receive Log in ID and password generated by the system. They can fill up the complaint form as long as the person is a member of FCSIT. The registered user can send the complaint, suggestion and feedback and receive confirmation and status of their complaint, suggestion and feedback from the system. The user can check about the action taken against their complaint, suggestion and feedback.

1.6 Importance of Project

This project dramatically simplifies the process of organizing and managing web-based help desk. Here are some reasons on why is it important:

i) Keep track of the record

This system allows the administrator to keep track of the record easily. They do not have to collect the complaint forms and sort the records manually. This system is also being able to ensure the administrator receives complaint, suggestion and feedback.

ii) Reducing paper work

The system will allow the user to fill in the form online. There is no more complaint in paper form. Therefore, it ensures that there is no concern about lost of form.

iii) Interactive system

The other important point of this project is to build an interactive system. It means this system provides a two-way electronic communication system that user can send and received response from the system.

iv) The record is store in safe place

This system is using database to store the complaint, suggestion and feedback. As the database also has a backup, so the record is stored in the safe place.

v) Efficient and fast

The complaint, suggestion and feedback are received very fast by the system. The administrator can take the action at the moment they receive the complaint, suggestion and feedback.

1.7 Expected Outcome

The main expected outcome is to provide a web-based helpdesk system for FSCIT. Other expected outcomes from the development of this system are:

i) Improvement of quality in services

As the University Malaya had received the ISO 9001:2000, the quality of the services must always be the priority. It is hoped that with the development of this system, the management of FSCIT can improve their services.

ii) Provide report and statistic

The system is expected to provide the auto generated report and statistic to help the administrator to keep track the record monthly. It is also provides an accurate and consistent report and statistic.

iii) Accuracy and security

For security reasons, this system will only allow authorized user to use the system. The users need to register before making the complaint, suggestion and feedback. So, the accuracy of the record is high and can be trusted.

iv) User-friendly

The system is a user-friendly system, which means that it is easy to learn, use, understand, or deal with.

1.8 Limitation of the system

The limitations for the system are listed as below:

i) The main limitation for this project is it depending on the availability of the Internet services. Without the Internet services, nobody can access the system.

ii) This system is provided for FCSIT user only. Non-member of the faculty cannot make any complaint, suggestion and feedback. The system handles cases within the organization only and has no integration with other faculty.

iii) This system does not provide anonymity function. The users need to provide their true identity and they need to sign up first before doing complaint, suggestion and feedback online.

iv) This system is also do not support multiple languages. All the information is provided in English.

1.9 Project Schedule

Project schedule served as a planned timetable for work management. The Gantt chart is the most commonly used project scheduling and progress evaluation tool. Project schedule act as a guideline to manage the time and tasks that need to be accomplished during the system development. The project schedule for the development of web-based Help Desk FCSIT is shown in Figure 1-1.

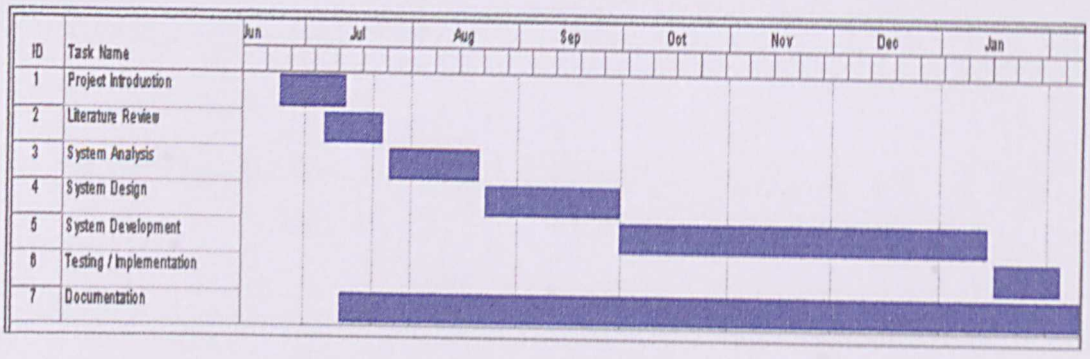


Figure 1-1 : Project Schedule

1.10 Summary

Help Desk FCSIT is a web-based system to enable more efficient and effective way of sending and retrieving complaint, suggestion and feedback information. The main focus of this system is about complaint, suggestion and feedback. This chapter covered the introduction of the project including project definition. This chapter also lists down the problem arising that triggers the development of this system. The goals and objectives are also included. This chapter also covered about the importance of this system and project schedule.

CHAPTER 2 :

LITERATURE REVIEW

2.2.1 Help Desk

In business, a help desk is a place that a consumer can call to get help with a problem. Help desks can range from one person answering a phone in smaller companies to, in larger companies, a group of topic experts using software to help track and analyze the problem. Help desk software is most commonly used in a call center [8].

A help desk is a generic name typically associated with the end user support center. Increasingly, the Help desk is being seen as an integral part of the service function, responsible for bringing multiple resources to bear to solve issues to the client's satisfaction.

Nowadays, many organizations are turning to web-based help desk to automate a variety of tasks and, at the same time, reduce costs by cutting staff and providing more user support with existing staff. The advantage of web-based help desk is that they allow fewer people to deal with higher volume of feedback, complaint and suggestion.

2.2.2 Help Desk Past and Present

The Help Desk is essentially a central point through which problems or issues are reported and subsequently managed and co-coordinated. From a general or wider perspective, it is an integral part of the service function, responsible for bringing resources together to address a problem or other issue.

Call Centers has traditionally used Help desks for customer enquires on pre or post sale issues. Telephone was the main media for providing support until the advent of the Internet. The telephone use has several weaknesses such as:

i. Customers get frustrated waiting on hold or navigating the automated telephone-answering message.

ii. If telephone messages are left then the customer is not sure that their problem will be dealt with or given the priority it deserves.

iii. Additional telephone calls need to be made into the call center for progress reports if telephone calls are not made back to them.

iv. Companies cannot predict the volume of calls, especially during system outages and new product launches. This results in the need to have spare capacity in the call center or risk of losing customers.

v. Customer must explain his or her problem, often to somebody untrained and unable to resolve the problem immediately. This makes them not convenient.

vi. All this adds up to additional costs of providing support through having spare capacity of staff, often under utilized telephone switch systems and associated equipment, phone call charges as well as the potential loss in revenue from unsatisfied customers.

Now, the Internet provides a media for potential and existing customers to communicate directly with a supplier and to review and often buy their services on line. The Internet based help desk complements the buying process and gave the customers a direct route into the help desk and took away the need for continually expanding the telephone call management systems. Few companies have fully exploited this potential. The web-based systems have a number of benefits that overcome the limitations of a telephone support help desk such as:

i. Available 24 hours a day – even if the call center is closed. This is especially important in today's business world where staffs often work late or customers or staffs are based overseas in different time zones.

ii. More comprehensive self-help facilities can be provided such as FAQ's. [8]

2.2.3 Complaint, Suggestion and Feedback

From the Merriam Webster dictionary [15], complaint means expression of grief, pain, or dissatisfaction; something that is the cause or subject of protest or outcry; a bodily ailment or disease; a formal allegation against a party. In this context, complaint can be interpreted as to say something is wrong or dissatisfaction.

Suggestion means to mention an idea, possible plan or action for other people to consider. Meanwhile the feedback means, the return to the input of a part of the output of a machine, system, or process; the partial reversion of the effects of a process to its source or to a preceding stage; the transmission of evaluative or

corrective information to the original or controlling source about an action, event, or process. In the other word, feedback is means any respond from customer to the organizations.

Many industries have long recognized the importance of complaint, suggestions and feedback from their client. Complaints, suggestion or feedback about the services help the organization to recognize rooms for improvement. Therefore, many companies have set up a team to handle the complaint, suggestion and from clients. In fact, by doing this, they welcome complaints, suggestions and feedback about the services provided.

Many industries have the same concepts and procedures in handling complaint, suggestion and feedback cases. The procedure remains unchanged since ages; where the user needs to fill a form manually and handed in back to the organization for further action. They need to fill in their name, address and contact number. Even though some of the organization may handle anonymity cases, but they prefer people to provide the true identity of themselves; especially for government sector. Furthermore, the form should be completed properly because the clear statements can help the organization to solve the cases easier.

2.2.4 Help Desk System

The term help desk usually refers to a place that a consumer could ask for assistant about the services from a service provider via a call. Help Desk System is a system that enables the organization to help manage all complaints, suggestions and

feedback relating to their services. The functionality of help desk system includes data collection, monitoring, reporting, analysis, statistic and quality improvement. Therefore, Help desk are crucial to improve the services of an organization.

There are many reasons for implementing a help desk system. The system will have to improve customer satisfaction about the service they receive. In addition, the data they receive will assist the organization to identify opportunities for improvement and change procedures to optimize service and minimize complaint, suggestion and feedback in the future. Nevertheless, the organization should have a standard checklist to ensure the effectiveness of the help desk system.

2.2.5 Web Concept of Help Desk System

Web-based system describes system that can be accessed through a "web browser" and using the format of a web page. Basically, a web-based helpdesk will have the same concepts of the manual help desk system, but would be different as it utilizes an online system. One needs to have Internet connection in order to get access to it. Therefore, there would involve less paper-based documents as in manually system.

2.3 Fact Finding Techniques

Fact-finding is needed in order to have a better understanding of the system's needs and requirements. In order to identify them, a lot of information is needed. A few techniques have been used to find out what the system needs and users really want.

The requirement elicitation takes quite a long time. This is due to several techniques need to be applied in order to get a complete requirement.

2.3.1 Analysis on existing system

Research has been conducted on both the local and foreign countries' help desk and complaint system to get a better view and understanding of how a real help desk and complaint systems is currently implemented. Besides that, the advantages and disadvantages of these web-based systems are compared as well.

2.3.1.1 Public Complaint System, Department of Irrigation and Drainage (Malaysia)

URL : http://didnet.moa.my/complaint/

Date access: 30 June 2004

The Department of Irrigation and Drainage aspires to be a world-class organization by the year 2010 and is committed to an excellence work culture focusing on continuous improvement towards customer satisfaction. Their mission is to provide excellent services in the development and management of irrigation, drainage, river, coastal zone, hydrology and water resources contributing towards advancement in agriculture, uplifting of quality of life and conservation of environment for national development.

This Public Complaint System provides the following features as shown in Figure 2-1 :

i. Online Complaint Form

ii. Complainer Answer Retrieval

iii. Complaint Guide

iv. Contact Public Relation Officer

v. Contact System Administrator

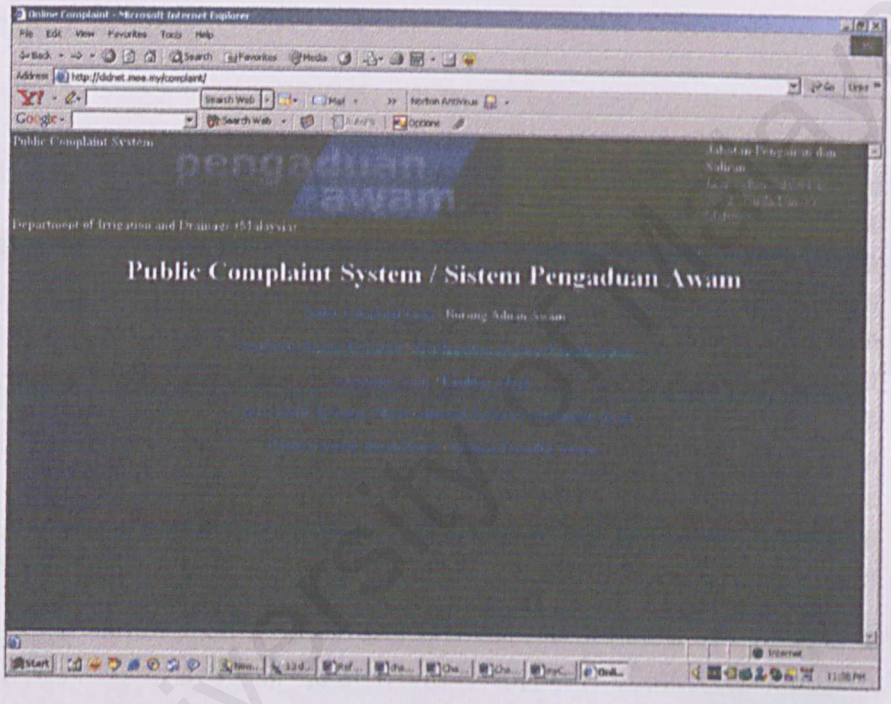


Figure 2-1 : Main page for Public Complaint System

The advantages and disadvantage of the Public Complaint System are as follow:

Advantages:

i. This system has a complaint guide, which helps the user about the procedure and process in doing feedback.

ii. The user can attach image(s) file by pressing the BROWSE button and selecting the file from their computer as shown in Figure 2-2. They can send only 5 JPEG (*.jpg) images with size less than 500k per image.

iii. This system has the Confirmation Page, which allow the user to review the complaint to confirm the information given.

iv. Provide anonymity means the user is allowed to complaint without giving their true identity.

v. Have reference number for each complaint, for example: **Complaint ID No. 62 (DID Ref No. is Bil.(62) dlm. PPS.UPK.17e Jld.1)**

vi. User is allowed to update or change their personal information.

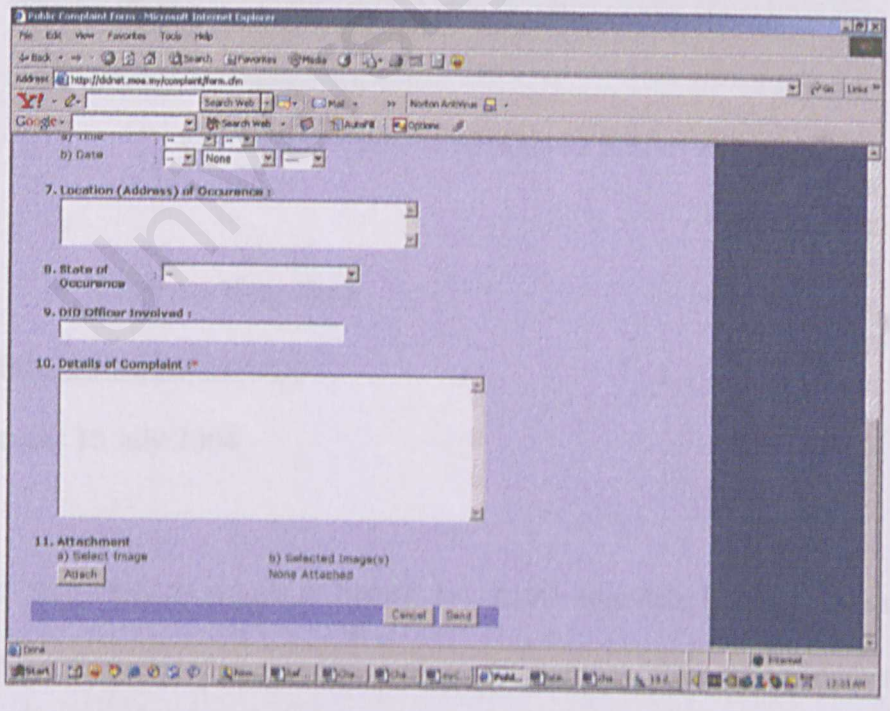


Figure 2-2 : Complaint Form of Public Complaint System

Disadvantage:

i. The user need to memorized or jot down the Complainer ID and also Complaint ID for future referencing as shown in Figure 2-3. These IDs are very important because if the user lost it, they cannot retrieve the feedback from the system.

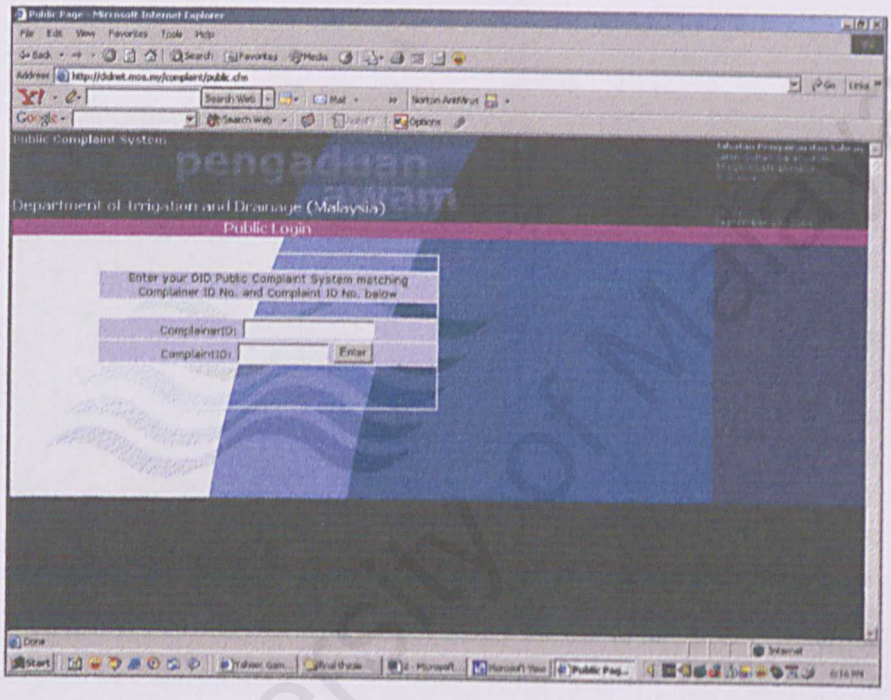


Figure 2-3 : Complainer Answer Retrieval

2.3.1.2 Liberum Help Desk

URL : http://www.liberum.org/

Date access : 15 July 2004

Liberum Help Desk as shown in Figure 2-4, is the complete help desk solution for small to medium sized businesses and organizations. This software provides a

simple, easy to use web interface for managing and tracking technical support problems.

Liberum Help Desk is open sourced under the GPL license and free for use. The help desk software is written in HTML and ASP and is easily modified and customized. All that is required to run Liberum Help Desk is Windows NT/2000/XP running IIS.

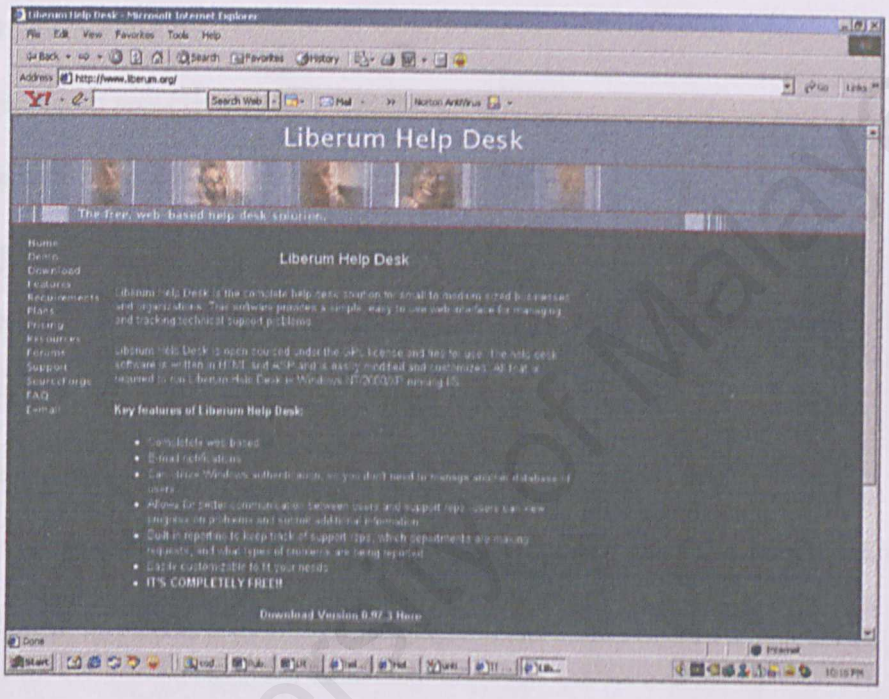


Figure 2-4 : Liberum Help Desk

The features of Liberum Help Desk are as follow depends on types of user:

Help Desk Administrators :

- choice of using NT user accounts or a separate database for granting users access to the help desk
- runs on Microsoft IIS - no proprietary web servers

- easy-to-use menus for managing reps, departments, categories, priorities and statuses
- runs on with MS SQL or an Access database
- built-in reports for keeping track of problems, users and support reps
- configurable email messages
- easily customizable pages to fit your organizations needs

Support Personnel :

- access from any client with a web browser
- e-mail notification of new problems or updates
- pager support
- problem listings with vital details
- ability to view and edit another rep's problems
- problem updates are stamped with date, time and username
- search through problems by username, category, description, solution, dates and more

The Users :

- submit problems via a web-based form
- view the progress of open problems
- update problems with additional information
- e-mail notification
- list previously submitted problems
- user searchable knowledge base

The advantages and disadvantage of the Liberum Help Desk are as follow:

Advantages:

i. Have E-mail notifications for users.

ii. Can utilize Windows authentication, so do not need to manage another database of users.

iii. Allows for better communication between users and support reps: users can view progress on problems and submit additional information

iv. Built-in reporting to keep track of support reps, which departments are making requests, and what types of problems are being reported.

v. The user may see the list of previously submitted problems.

Disadvantages:

i. The user needs to provide the true identity and register before use the help desk system.

2.3.1.3 IT Helpdesk, University of Mississippi

URL : http://www.olemiss.edu/helpdesk/

Date access : 31 July 2004

The mission of the help desk is to assist faculty, staff, and students of the University of Mississippi with their computing needs. The system provides assistance to the University community in the areas of microcomputer applications, hardware and networking. The IT Helpdesk is also support Email, Web pages, and other applications that run on the University's campus-wide systems. The main page of the IT Helpdesk is show in Figure 2-5.

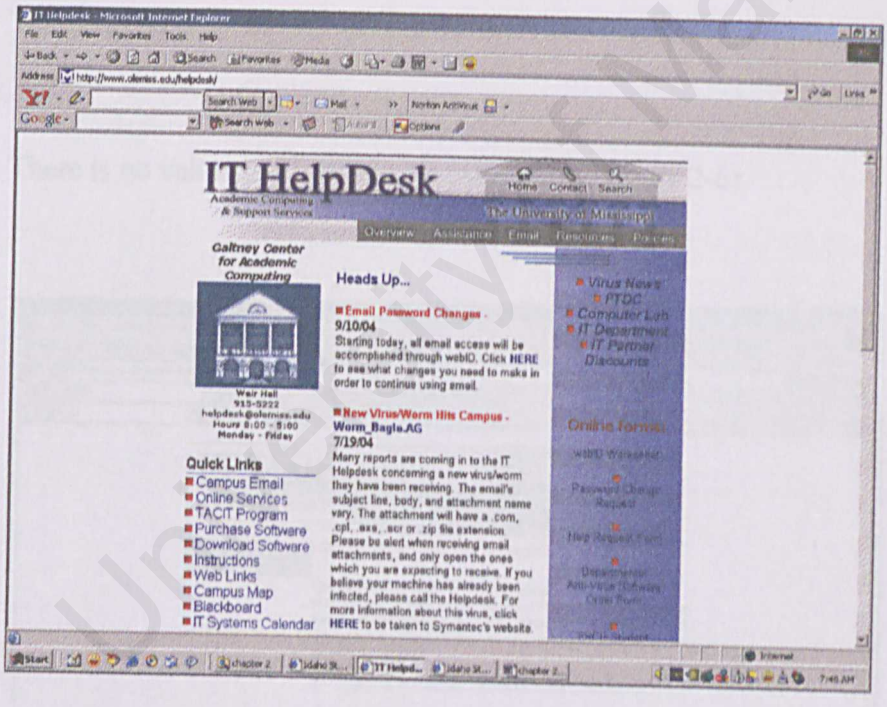


Figure 2-5 : IT HelpDesk, University of Mississippi

The advantages and disadvantage of the IT HelpDesk are as follow:

Advantages:

i. Provide quick links to many sources related to the university.

ii. The Helpdesk provides the following methods of service: Phone, Fax, On-Site, Email & Web, Helpdesk Workroom and Helpdesk Office.

iii. The university's computer users can report computer problems online.

Disadvantages:

i. The users have to choose the appropriate form before they can make a report.

ii. There is no validation for Help Request Form (Figure 2-6).

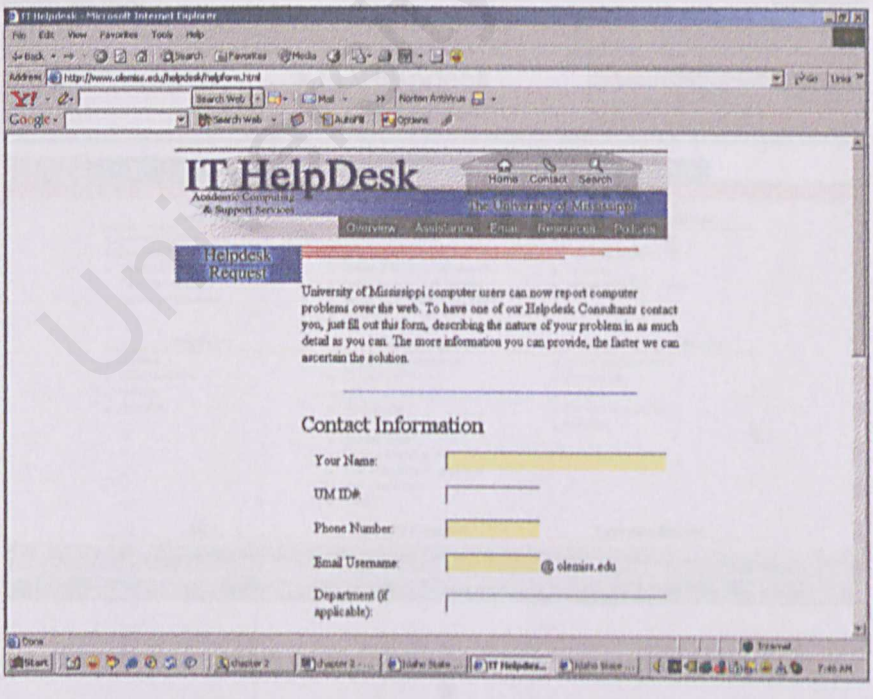


Figure 2-6 : The Help Request Form

2.3.1.4 Idaho State University Helpdesk

URL : http://help.isu.edu/index.php

Date access : 31 July 2004

The purpose of the Helpdesk as shown in Figure 2-7 is to serve as the primary point for ISU students, faculty and staff seeking problem resolution with supported software, hardware and operating systems, including general IT services provided by the Computer Center. The Helpdesk team provides technical support and troubleshooting for Faculty, Staff, and Students, in person and over the telephone. The Helpdesk team creates documents for the Helpdesk Knowledge Base, posts critical news alerts, and submits account requests. The Helpdesk team also creates the Helpdesk Tool Kit CD that is available for Faculty, Staff, and Students to purchase.

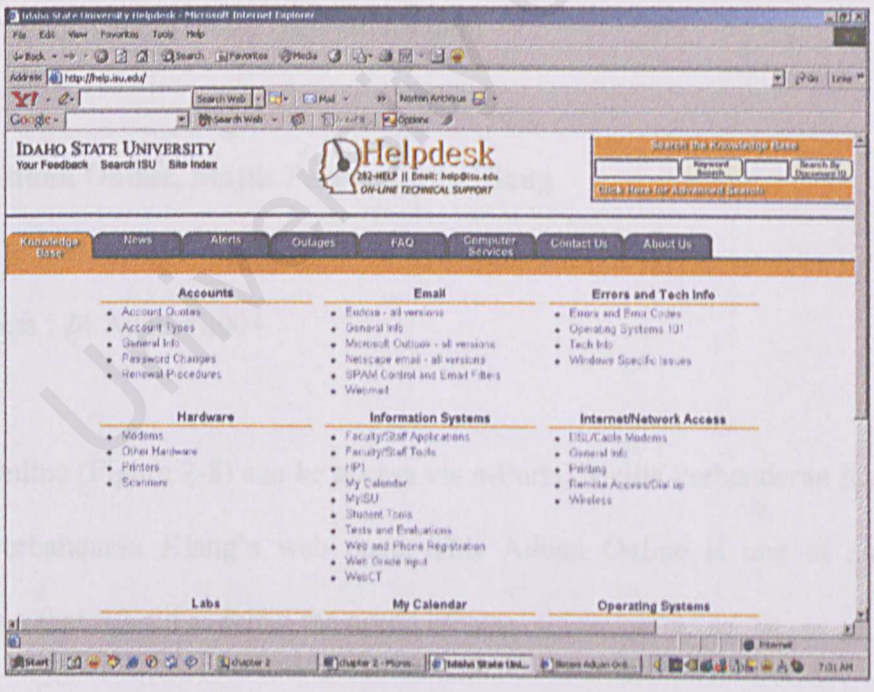


Figure 2-7 : Main page for Idaho State University Helpdesk

The advantages of the Idaho State University Helpdesk are as follow:

Advantages:

i. Provide the support procedures in doing problem report.

ii. Use a Help Desk software application known as Apriori for their Helpdesk.

iii. Provide FAQ.

iv. The on-line form is put in order, so that the user can easily surf it.

v. Provide search function for user to search document they need.

vi. Provide validation for each online form.

2.3.1.5 Aduan Online, Majlis Perbandaran Klang

URL : http://portal.mpklang.gov.my/aduanonline/index.cfm

Date access : 24 August 2004

Aduan Online (Figure 2-8) can be access via e-Portal Majlis Perbandaran Klang and Majlis Perbandaran Klang's web page. This Aduan Online is one of an online application that is built to fulfill the needs of user.

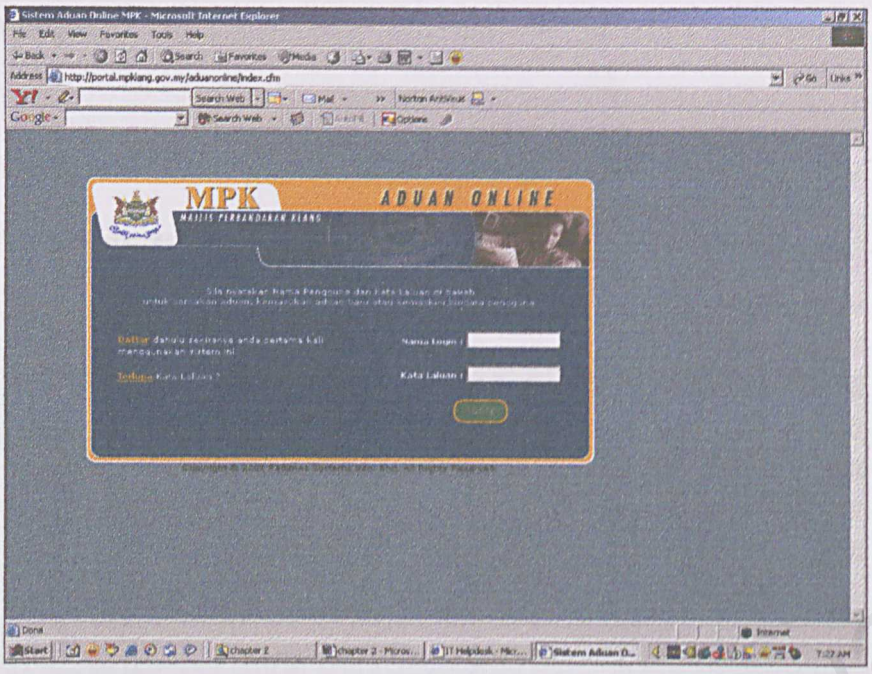


Figure 2-9 : Main page for Aduan Online

The advantages and disadvantage of the Aduan Online Majlis Perbandaran Kelang are as follow:

Advantages:

i. This system has the functionality of forgot the password. The user can retrieve back the password via e-mail.

ii. For each fields in the form, there is pop-up message for validation.

iii. There is Help function for the user in register form. This Help function state the requirement of each field in the registration form.

iv. User is allowed to update their personal information.

Advantage:

i. In Butiran Aduan page as shown in Figure 2-9, the use of text area in "Maklumbalas dari MPKlang" is not suitable. It may confuse the user.

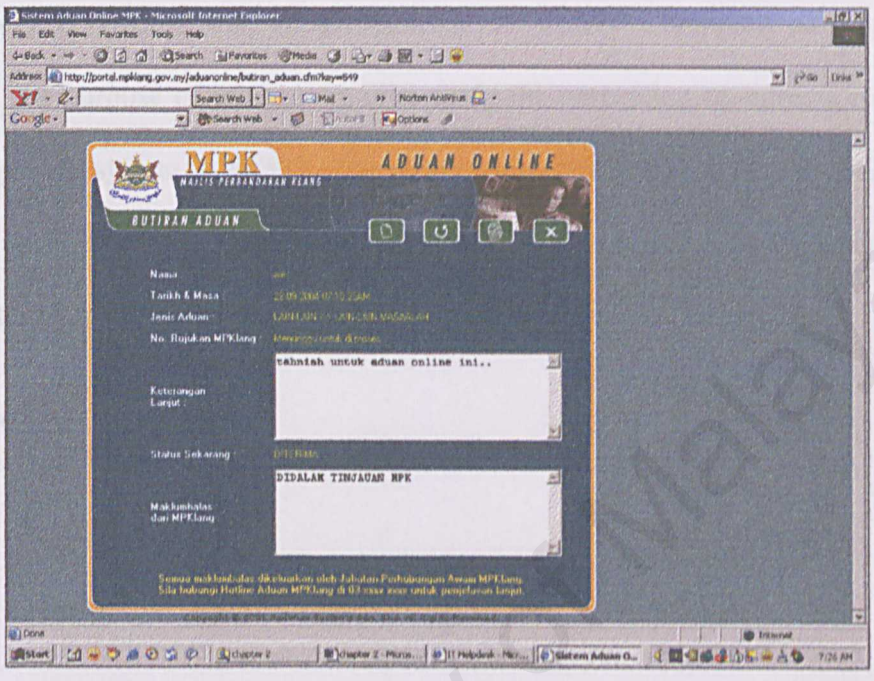


Figure 2-9 : The Butiran Aduan page

2.3.2 Questionnaire

Questionnaire is a set of questions for obtaining statistically useful or personal information from individuals. It is a special purpose document that allows analyst to collect information and opinion from respondents. It can be effective method for gathering facts as it allows the analyst to collect facts from large number of people while maintaining uniform responses.

Different individuals have different responses and this help the analyst to compare

the differences. The use of free format and fixed format questionnaires help to obtain information on requirements and perceptions towards the helpdesk.

2.3.2.1 Results of the questionnaire

The survey conducted was about a survey on FCSIT Complaint & Suggestion System. The purpose of this survey is to study the perception on current complaint and suggestion system in FCSIT. The target audiences for the survey are staffs and students of FCSIT. From 50 questionnaires distributed, only 30 respondents fill out the questionnaires.

The results of the questionnaire are as follow :

i. **Awareness of suggestion box among student**

From the survey as shown in Figure 2-10 below, 42% of the student is aware of the suggestion box, while the 58% said No. The percentages for question, "Do you know the procedure of filling a complaint or suggestion?" is same with the study on awareness of the student.

ii. **Willingness to complaint if online system exist**

83% of the students are willing to complaint if there is an online system as shown in Figure 2-11, while the rest said No.

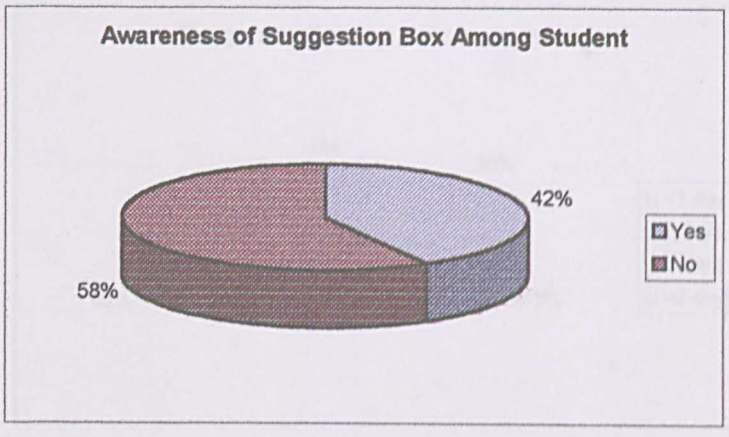


Figure 2-10 : Awareness of suggestion box among student

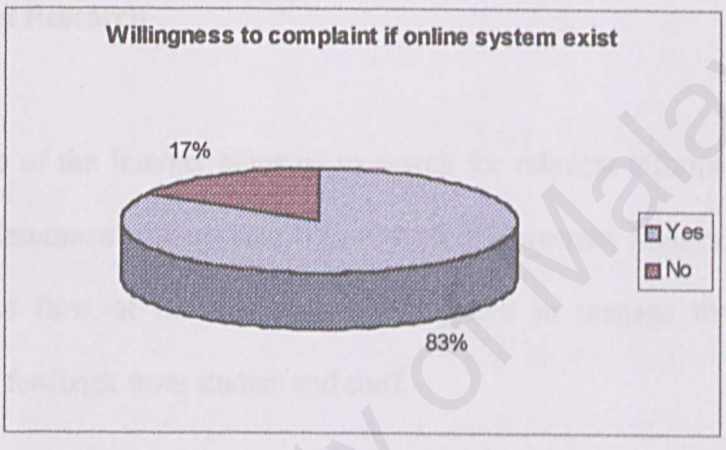


Figure 2-11 : Willingness to complaint among student

iii. Response time on existing system

As shown in Figure 2-12, the days taken to settle down the complaint is highest in more than three days with 62%. The response within three days is 13% and within two days is 25%. While there is no cases is settled down by less than one day.

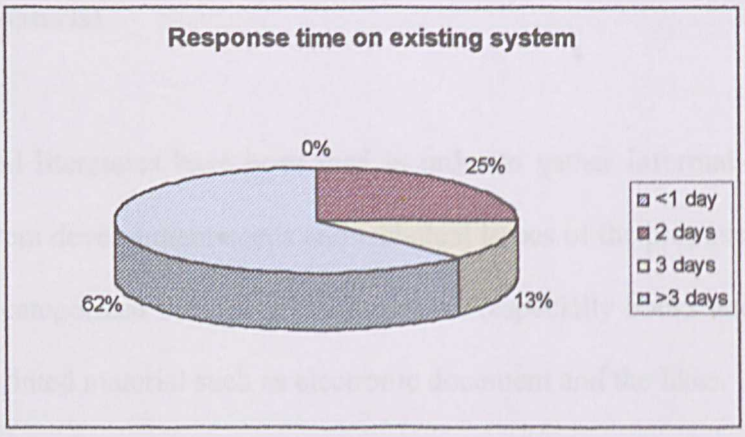


Figure 2-12 : Response time on existing system

2.3.3 Internet Research

The emergence of the Internet helps us to search for relevant information such as from Quality Assurance Management Unit (QAMU), University Malaya. Research is done about the flow of the current system applied in manage the complaint, suggestion and feedback from student and staff.

Internet is also used to find out the existing system and do research about it. Internet is used as the main resource for referring any ambiguities that arise during the entire development period. By analyzing in the similar system documentation has made a big help in giving ideas on the features, functionality as well as the design of the web-based system. Besides that, online tutorials regarding programming language can also be obtained through surfing the Internet.

2.3.4 Printed Material

A lot of published literatures have been read in order to gather information of the users' needs, system development needs and technical issues of the proposed system. All these can be categorized into the printed material (especially books and seniors' thesis) and non-printed material such as electronic document and the likes.

2.4 System Architecture

The term client/server was first used in the 1980s in reference to personal computers (PCs) on a network. The actual client/server model started gaining acceptance in the late 1980s. The client/server software architecture is a versatile, message-based and modular infrastructure that is intended to improve usability, flexibility, interoperability, and scalability as compared to centralized, mainframe, time sharing computing.

A client is defined as a requester of services and a server is defined as the provider of services. A single machine can be both a client and a server depending on the software configuration [9].

2.4.1 Client/server architecture

As a result of the limitations of file sharing architectures, the client/server architecture emerged. This approach introduced a database server to replace the file server. Using a relational database management system (DBMS), user queries could

be answered directly. The client/server architecture reduced network traffic by providing a query response rather than total file transfer. It improves multi-user updating through a GUI front end to a shared database. In client/server architectures, Remote Procedure Calls (RPCs) or standard query language (SQL) statements are typically used to communicate between the client and server [9].

2.4.1.1 Two-tier Architecture

With two tier client/server architectures as shown in Figure 2-13, the user system interface is usually located in the user's desktop environment and the database management services are usually in a server that is a more powerful machine that services many clients. Processing management is split between the user system interface environment and the database management server environment. The database management server provides stored procedures and triggers. There are a number of software vendors that provide tools to simplify development of applications for the two-tier client/server architecture.

The two-tier client/server architecture is a good solution for distributed computing when work groups are defined as a dozen to 100 people interacting on a LAN simultaneously. It does have a number of limitations. When the number of users exceeds 100, performance begins to deteriorate. This limitation is a result of the server maintaining a connection via "keep-alive" messages with each client, even when no work is being done. A second limitation of the two tier architecture is that implementation of processing management services using vendor proprietary database procedures restricts flexibility and choice of DBMS for applications.

Finally, current implementations of the two-tier architecture provide limited flexibility in moving (repartitioning) program functionality from one server to another without manually regenerating procedural code.

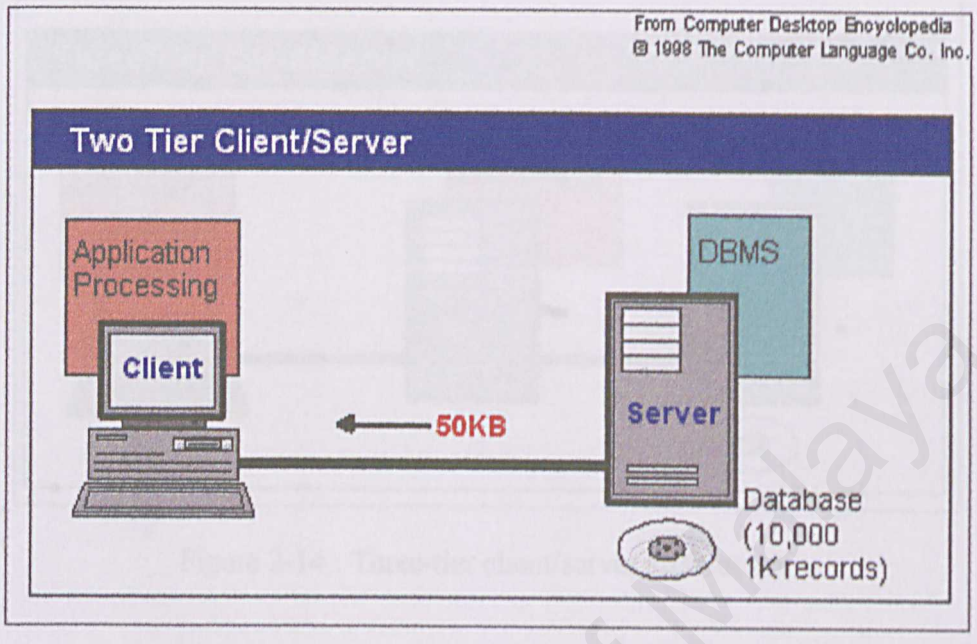


Figure 2-13 : Two-tier client/server architecture

2.4.1.2 Three-tier Architecture

The three- tier architecture (also referred to as the multi-tier architecture) as in Figure 2-14 emerged to overcome the limitations of the two-tier architecture. In the three-tier architecture, a middle tier was added between the user system interface client environment and the database management server environment. There are a variety of ways of implementing this middle tier, such as transaction processing monitors, message servers, or application servers. The middle tier can perform queuing, application execution, and database staging. For example, if the middle tier provides queuing, the client can deliver its request to the middle layer and disengage because

the middle tier will access the data and return the answer to the client. In addition the middle layer adds scheduling and prioritization for work in progress.

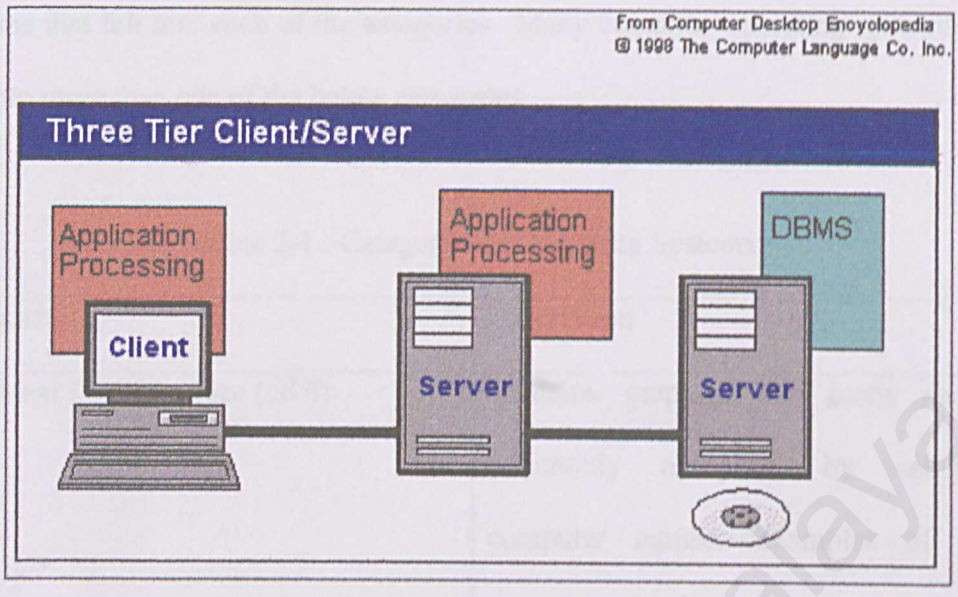


Figure 2-14 : Three-tier client/server architecture

The three-tier client/server architecture has been shown to improve performance for groups with a large number of users (in the thousands) and improves flexibility when compared to the two tier approach. Flexibility in partitioning can be a simple as "dragging and dropping" application code modules onto different computers in some three tier architectures. A limitation with three tier architectures is that the development environment is reportedly more difficult to use than the visually-oriented development of two tier applications [9].

2.5 Operating System

An operating system or OS is a software program that enables the computer hardware to communicate and operate with the computer software. Without a

computer operating system a computer would be useless. As computers have progressed and developed so have the types of operating systems. Below is a basic list of the different types of operating systems and a few examples of operating systems that fall into each of the categories. Many computer operating systems will fall into more then one of the below categories.

Table 2-1 : Categories of Operating Systems

Categories	Description
Graphical User Interface (GUI)	Contains graphics and icons and is commonly navigated by using a computer mouse. Examples of GUI operating systems: i. System 7.x ii. Windows 98 iii. Windows CE
Multi-user	Allows for multiple users to use the same computer at the same time and, or at different times. Examples of multi-user operating systems: i. Linux ii. UNIX iii. Windows 2000

Multitasking	An operating systems that is capable of allowing multiple software processes to be run at the same time. Examples of multitasking operating systems. i. UNIX ii. Windows 2000 iii. Windows XP
Multiprocessing	An operating system capable of supporting and utilizing more than one computer processor. Examples of multiprocessing operating systems. i. Linux ii. UNIX iii. Windows 2000
Multithreading	Operating systems that allow different parts of software program to run concurrently. Operating systems that would fall into this category are: i. Linux ii. UNIX iii. Windows 2000

2.5.1 Windows XP Professional

Windows XP is the operating system release that unifies the Microsoft range, with all the desktop versions now built on the NT/2000 code base rather than the shakier foundation of Windows 95, 98, and Me. That makes XP a great upgrade for users of the now-obsolete 9x and Me line, but for those already on Windows 2000 Professional it is a closer call.

Windows XP certainly looks different, with rounded window corners, larger and more detailed icons, and a clean-look desktop that on first installation shows only the taskbar and Recycle Bin. Windows XP is also more customizable than earlier versions of Windows, and includes visual themes that let you change the whole appearance of Windows in an instant. That is the window-dressing, but underneath are some significant improvements. One of the most interesting is Remote Desktop. A standard XP feature, this uses technology from Microsoft Terminal Server to enable users to access their computer over any connection; for example, by dialing into the office from home. This is not just file access--this technology lets you run applications remotely as if you were sitting at your desk at work. This is mature technology, stable and carefully thought out. So, for example, you can print from a remote word processor to a local printer. A variation on the theme is Remote Assistance, where the user can allow a remote helper to view their desktop, or optionally gain control of the keyboard and mouse, in order to troubleshoot a problem. The feature can also be disabled to ease security concerns.

The XP user interface is not a radical departure from earlier versions of Windows, but there are a number of small changes that together add up to a significant improvement. The Start menu now automatically features the most frequently used programs at the top of the list, and you can add and remove shortcuts by right-clicking the icon and selecting Pin or Unpin from the pop-up menu. Windows online help is integrated into a Help and Support Center that works like an internal Web site, with searchable help, tutorials, and walk-through.

Windows XP Professional has extra functionality such as support for dual processors, encryptable file systems, offline folders, the Remote Desktop, and extra administration features that come into play when connected to a Windows server domain.

Designed for reliability, security and privacy, high performance, and ease of use, the Windows XP operating system provides a host of benefits for business and home users. A clean and simple desktop, and easy-to-use features that take advantage of the digital age all contribute to the value of Windows XP.

The most full-featured edition of Windows XP, the Professional version is designed for businesses of all sizes, as well as home users who demand more from their PCs. Microsoft Windows XP Professional provides the highest levels of performance, productivity, and security, plus all the exciting multimedia features [12].

2.5.2 Linux

Linux has gradually become a popular operating system for Internet and intranet serving purposes. With a host of performance enhancements that will benefit Web sites and Internet sites of all sizes, Linux is a stable and high-performance operating system for Internet usage.

Linux has made progress, primarily in functionality important to Internet infrastructure and Web server capabilities, including a greater selection of drivers, easier installation, GUI-based front ends for Web administration and window management.

2.6 Web Server

A web server is a computer with special software to host web pages and web applications. Web server's traditional function has been to serve static HTML pages. As the Internet has become more functional, that is e-commerce and dynamic sites, increasing emphasis is placed on a servers ability to host web applications.

Many different servers are in used on the Internet. Some of the most popular ones are: Apache and Internet Information Server (IIS) [26].

2.6.1 Apache server

The Apache server is a powerful, flexible, HTTP 1.1 compliant Web Server. It is highly configurable and extensible with third-party modules. It provides full source code and comes with an unrestrictive license. It runs on Windows NT/9x, OS/2 and most version of Unix, as well as several other operating systems. Apache has a built-in search engine and HTML authoring tools and supports File Transfer Protocol (FTP). Apache is actively being developed and encourages user feedback through new ideas, bug reports and patches.

The Apache server allows administrators to easily set up password-protected pages with enormous numbers of authorized users, without slowing down the server. It also permits administrators to set up customized files, or even CGI scripts, which are returned by the server in response to errors and problems. This allows the administrator to perform on-the-fly diagnostics for both users and administrator.

Apache is also flexible enough to perform multiple Directory Index directives, where administrators can instruct the server to either send back index .html or run index .cgi when a directory URL is requested, which ever it finds in the directory. Those come running Apache servers will also find that it has unlimited flexible URL rewriting and aliasing. Apache has no fixed limit on the numbers of Aliases and Redirects that may be declared in its configuration files. In additional, a powerful rewriting engine can be used to solve most URL manipulation problems.

Today the Apache server is the most widely implemented Wed server on the Internet. It offers a powerful and customizable approach for any Unix-based server. It has been shown to be substantially faster, more stable and more feature-full than many other Web servers, including IIS. Apache is run on sites that get millions of hits per day and they have yet experienced any performance difficulties. [27],[28].

2.6.2 Internet Information Services 5.1

Windows XP Professional includes Internet Information Services (IIS) version 5.1, which makes it possible to host Web site on the Internet or the intranet. IIS includes a broad range of administrative features for managing Web sites. With programmatic features like Active Server Pages (ASP), user can create and deploy scalable, flexible Web applications on the Internet or your intranet.

Other features of IIS 5.1 in Windows XP Professional include:

i. Microsoft Management Console (MMC) snap-in for managing IIS. The MMC task pad considerably simplifies the administration of the Web site.

ii. The use of Remote Desktop technology for managing the Web site remotely. The Remote Desktop technology in Windows XP Professional lets administrators remotely administer IIS by using the MMC over a dial-up or PPTP connection

iii. Support for Active Server Pages. IIS support for ASP provides an easy-to-use alternative to Common Gateway Interface (CGI) and Internet Server Application Program Interface (ISAPI) by letting content developers embed any scripting language or server component into their HTML pages. ASP pages provide standards-based database connectivity and the ability to customize content for different browsers. ASP also provides error-handling capabilities for Web-based applications

iv. Up-to-the-minute security features

v. Improved metabase backup and restore functions. User can back up and save metabase settings to make it easy to return to a safe, known state. (A metabase is the structure for storing IIS configuration settings; the metabase performs some of the same functions as the system registry, but uses less disk space.)

vi. More extensive support for Web folders using Web Distributed Authoring and Versioning (WebDAV). WebDAV is an Internet standard that lets multiple people collaborate on a document using an Internet-based shared file system

vii. Web Folders let users maintain a consistent look and feel between navigating the local file system, a networked drive, and an Internet Web site [14].

2.7 Programming Language

Programming language is an artificial language that is used to generate or to express computer programs. The language may be a high-level language, an assembly language, or a machine language.

2.7.1 Cold Fusion

ColdFusion is a popular and sophisticated set of products for building Web sites and serving pages to users which enable a content database to be built by using input templates and combine these with application programs to create a Web site in which pages are developed dynamically as they are served. ColdFusion consists of ColdFusion Studio, which is used to build a site, and ColdFusion Server, which serves the pages to users. ColdFusion Studio is described as "a complete integrated development environment (IDE)" and ColdFusion Server as "a deployment platform."

The most valuable feature for ColdFusion is the ability to build Web sites as "piece parts" that can be stored in a database and then reassembled for Web pages, e-mail newsletters, and other uses. ColdFusion provides a visual interface for building Web pages directly or for building the "piece parts." ColdFusion is also a popular tool for building E-Commerce sites.

ColdFusion has its own page markup language, called ColdFusion Markup Language (CFML). CFML encompasses the Web's Hypertext Markup Language (HTML) and Extensible Markup Language (XML). A just-in-time (JIT) compiler turns the CFML

into the pages that get served. ColdFusion is open and "extensible". Applications can access databases using Microsoft's OLE DB, Open Database Connectivity (ODBC), or drivers that access Oracle and Sybase databases. ColdFusion can be coordinated with distributed applications that use Common Object Request Broker Architecture (CORBA) or Microsoft's Distributed Component Object Model (DCOM) to interact with other network applications. ColdFusion is also scalable, allowing both the size of a database and the number of users that can be served to grow. For large Web sites, multiple ColdFusion servers can be run together as a cluster. [34]

2.7.2 Active Server Pages

Active Server Pages (ASP) is the server-based technology from Microsoft. The basic of Active Server Pages is the Internet Information Server(IIS). Active Server Pages is used to create dynamic and interactive web pages and it allows ActiveX controls and Java Applet to be included. It has pre-built Active Server components that provide plug-in objects that will perform specific tasks. Active Database object is one of the Active Server components allows easy but powerful connections to be made to almost any database system. It is suitable for publishing and collecting data on the web. It provides a way for building secure transactions, server-based application and web sites.

2.8 Scripting Language

Scripting language is a programming language supported by and specific to a particular program. A scripting program is normally used to automate complex or advanced features or procedures within the program.

2.8.1 VBScript

VBScript, or by its full name, the Microsoft Visual Basic Scripting Edition Language, is a simplified version of the Visual Basic and Visual Basic for Applications family of programming languages. It is closely related to the BASIC programming language. While it does not offer the functionality of Visual Basic, it does provide a powerful, easy to learn tool that can be used to add interaction to Web pages.

VBScript is a scripting language, or more precisely a scripting environment, which can enhance HTML Web pages by making them active, as compared to a simple static display. It is the default language of ASP and is event-driven. VBScript provides a small but sufficient set of error handling capabilities. Handling multi-dimensional arrays in VBScript is also a breeze. VBScript arrays can easily be dimensioned and re-dimensioned, as required by the developer.

VBScript talks to host applications using ActiveX Scripting. With ActiveX Scripting, browsers and other host applications don't require special integration code for each scripting component. ActiveX Scripting enables a host to compile scripts, obtain and

call entry points, and manage the namespace available to the developer. With ActiveX Scripting, language vendors can create standard language run times for scripting. Microsoft will provide run-time support for VBScript. Microsoft is working with various Internet groups to define the ActiveX Scripting standard so that scripting engines can be interchangeable.

Specifically, VBScript was created by Microsoft to use either as a client-side scripting language for the Microsoft Internet Explorer (version 3.0 and later), or as a server-side scripting language with the Microsoft Internet Information Server (version 3.0 and later). However, VBScript is more often used as a server-side scripting language because of its potent processing capabilities on the server-side. [29], [30]

2.8.2 JavaScript

JavaScript is a platform-independent, event-driven, interpreted programming language developed by the Netscape Communications Corporation and Sun Microsystems. Originally called Live Script. JavaScript is affiliated with Sun's object-oriented programming language Java primarily as a marketing convenience. They interoperate well but are technically, functionally and behaviorally very different. A popular misconception about JavaScript is that many people believe that JavaScript is Java because of their similar names. This is definitely not true as JavaScript is not Java.

User can develop server applications or client applications with Java Script. The term "server" is referring to the computer where the Web page resides. The term "client" is referring to the browser application that loads and displays Web page. Java Script is an extension to HTML that lets users create more sophisticated Web pages than they ever could with HTML alone.

The Netscape Navigator 2.01 and its later releases support Java Script. There are several version of Java Script supported by certain browsers and browser version. Unfortunately, this can often lead to confusion in compatibilities. Since Netscape originally introduced Java Script, Java Script 1.0 was the language specification supported in Netscape Navigator 2.0. Subsequently, Navigator 3.0 supported new enhancements that comprised Java Script 1.1. At present, Navigator 4.0 support Java Script 1.2.

Java Script offers much more expressive power that HTML alone. Java Script can do thing such as create multipart documents, build dynamic documents that take users through a Web site from one document to another, and generate documents that interact with the user. [31], [32], [33].

2.9 Web Authoring Tool

Also known as *authorware,* a program that helps you write hypertext or multimedia applications. Authoring tools usually enable you to create a final application merely by linking together objects, such as a paragraph of text, an illustration, or a song. By defining the objects' relationships to each other, and by sequencing them in an

page layout tool, the control of an HTML text, editor and support for new web technologies, all in one software packing.

Developers can use it to create web sites visually, with confidences that HTML being generated is concise and always editable. It includes advanced features that takes advantage of the latest innovations on the web, such as dynamic HTML and CSS, while still ensuring that web pages work well in a variety of web browsers. All of the code generated by it is carefully created to work on as many platforms and browsers as possible.

Others features include easy integration of Active X components, Java applets, Plug-ins for improved web page interactivity. It also integrates seamlessly with other components of Macromedia, such as Flash Movies, Shockwave, and Fireworks, which are essential for the development of interactive web pages.

2.10 Database Management System (DBMS)

A database is a structured collection of data. To add, access, and process data stored in a computer database, a database server is needed. There are several database management system available currently: MS Access 2000 and MS SQL Server 2000.

2.10.1 MS Access 2000

Microsoft Access 2000 is a Window-based database management system. It is a member of the Microsoft Office 2000 family and it runs under the Windows

appropriate order, authors (those who use authoring tools) can produce attractive and useful graphics applications. Most authoring systems also support a scripting language for more sophisticated applications.

The distinction between authoring tools and programming tools is not clear-cut. Typically, though, authoring tools require less technical knowledge to master and are used exclusively for applications that present a mixture of textual, graphical, and audio data [35].

2.9.1 Microsoft FrontPage 2000

FrontPage 2000 is aimed at letting non-programmers build no-nonsense pages for their web sites quickly and easily. It features built-in tools to facilitate the creation of forms, tables, banners, Java Applets, and much more. Developers, writers and designers may also work on the same web project by using Microsoft FrontPage and Microsoft Visual Interdev in conjunction. In this case, Visual Interdev provides developers with a robust set of tools for developing web applications, while FrontPage provides a WYSIWYG environment for editing pages that does not require any programming knowledge.

2.9.2 Macromedia Dreamweaver MX

Macromedia Dreamweaver MX is a professional visual editor for creating and managing web sites and pages. It gives developers the productivity of a visual web

95/98/NT/2000 operating system. Due to the fact that Access is part of the Office 2000 suite, it interoperates well with the other components of the Office 2000 family.

Access is easily the world's most popular relational database management software (RDBMS). Access 2000 brings not only the traditional broad range of easy data management tools but also adds increased integration with the Web for easier sharing of data across a variety of platforms and user levels and additional ease-of-use enhancements to assist with personal productivity. With Access, the database administrator can design and use databases very quickly, as it has a very user-friendly interface. Also, Stat/Transfer can be used to convert data between Microsoft Access and your favorite spreadsheet, database or statistical package. Besides that, data in Microsoft Access can be migrated to the Microsoft SQL Server.

Access 2000 continues to offer an easy-to-use tool for easily finding information that provides consistency and integration with the other applications in the office suite. It also allows easily sharing information via the corporate Intranet and the ability to easily host a database within the browser. This combines the power of a desktop database with the power of the Web. Furthermore, tables, forms, queries, and reports can be generated just at a snap of a finger, just by using the set of wizards that come with this software. All this makes Access an excellent all-in-one database tool for creating standalone database application [25].

2.10.2 MS SQL Server 2000

Microsoft SQL Server2000 extends the performance, reliability, quality, and ease-of-use of Microsoft SQL Server version 7.0. Microsoft SQL Server 2000 includes several new features that make it an excellent database platform for large-scale online transactional processing (OLTP), data warehousing, and e-commerce applications.

The OLAP Services feature available in SQL Server version 7.0 is now called SQL Server 2000 Analysis Services. The term OLAP Services has been replaced with the term Analysis Services. Analysis Services also includes a new data mining component.

The Repository component available in SQL Server version 7.0 is now called Microsoft SQL Server 2000 Meta Data Services. References to the component now use the term Meta Data Services. The term repository is used only in reference to the repository engine within Meta Data Services.

Microsoft SQL Server2000 introduces several server improvements and new features:

i. **XML Support**

   The relational database engine can return data as Extensible Markup Language (XML) documents. Additionally, XML can also be used to insert, update, and delete values in the database.

ii. **Federate database server**

SQL Server 2000 supports enhancements to distributed partitioned views that allow you to partition tables horizontally across multiple servers. This allows you to scale out one database server to a group of database servers that cooperate to provide the same performance levels as a cluster of database servers. This group, or federation, of database servers can support the data storage requirements of the largest Web sites and enterprise data processing systems.

iii. **User-defined function**

The programmability of Transact-SQL can be extended by creating own Transact-SQL functions. A user-defined function can return either a scalar value or a table.

iv. **Indexed Views**

Indexed views can significantly improve the performance of an application where queries frequently perform certain joins or aggregations. An indexed view allows indexes to be created on views, where the result set of the view is stored and indexed in the database. Existing applications do not need to be modified to take advantage of the performance improvements with indexed views.

v. **Scalability Enhancements for Utility Operations**

SQL Server 2000 enhancements for utility operations include faster differential backups, parallel Database Console Command (DBCC) checking,

and parallel scanning. Differential backups can now be completed in a time that is proportional to the amount of data changed since the last full backup. DBCC can be run without taking shared table locks while scanning tables, thereby enabling them to be run concurrently with update activity on tables. Additionally, DBCC now takes advantage of multiple processors, thus enabling near-linear gain in performance in relation to the number of CPUs (provided that I/O is not a bottleneck).

vi. 64-GB Memory Support

Microsoft SQL Server 2000 Enterprise Edition can use the Microsoft Windows 2000 Advanced Windows Extension (AWE) API to support up to 64 GB of physical memory (RAM) on a computer.

2.11 Summary

The literature review is doing for analysis to consider the requirements for the system. After discusses the literature reviews regarding the system architecture, system platform, database system, programming and also development tools that are done, the next chapter will discuss on the methodology of the project.

CHAPTER 3 :

METHODOLOGY

CHAPTER 3 : METHODOLOGY

3.1 Introduction

The previous chapters have introduced the overview and information gathered in the literature review about the Help desk. In this chapter, the justifications for the chosen project methodology would further be elaborated.

In order to produce a correct and functional system, the approach in handling the system development process is the most essential component. Different types of system development methodologies can be used in designing information system. System development methodology can be described as a systematic description of the sequence of activities required to develop an information system. Methodology is used to make the project development more manageable and well structured.

3.2 System Development Methodology

The methodology used for Help Desk FCSIT is the Waterfall life-cycle model. The Waterfall life cycle model was introduced by Winston Royce in 1970 and is currently the most commonly used model for system development. The Waterfall model is also known as the traditional method that divides the project into well-defined sequential stages with intermediate milestones. The final product is not delivered until all phases are finished [16]. "Waterfall" method has many advantages, as it is the most direct way to the objective within the shortest development time and cost possible.

The Waterfall model in Figure 3-1 is also called the "classic life cycle" or the "linear sequential model". The stages of the waterfall model are performed sequentially, as in a systematic fashion. Only after a stage is completed, then only the next stage begins. The waterfall model also allows feedback loops for maintenance while the system is being developed. Therefore, if errors are detected during execution of any stage, one can return to the previous stage for corrections.

3.3 Justification of Methodology Selected

It is easier to perform a sequence of smaller tasks than one large task. That is why the overall life-cycle model is broken into a series of smaller steps called phases. The justification of each phases in the Waterfall life cycle are as follow:

3.3.1 Requirements

During the requirements phase, the concept of Help Desk FCSIT is explored and refined. Through observations and surveys conducted on the problems, the user's requirements have been elicited. Investigation on the current manual system is also done to understand how it is works, the problems raised and the existing constrains.

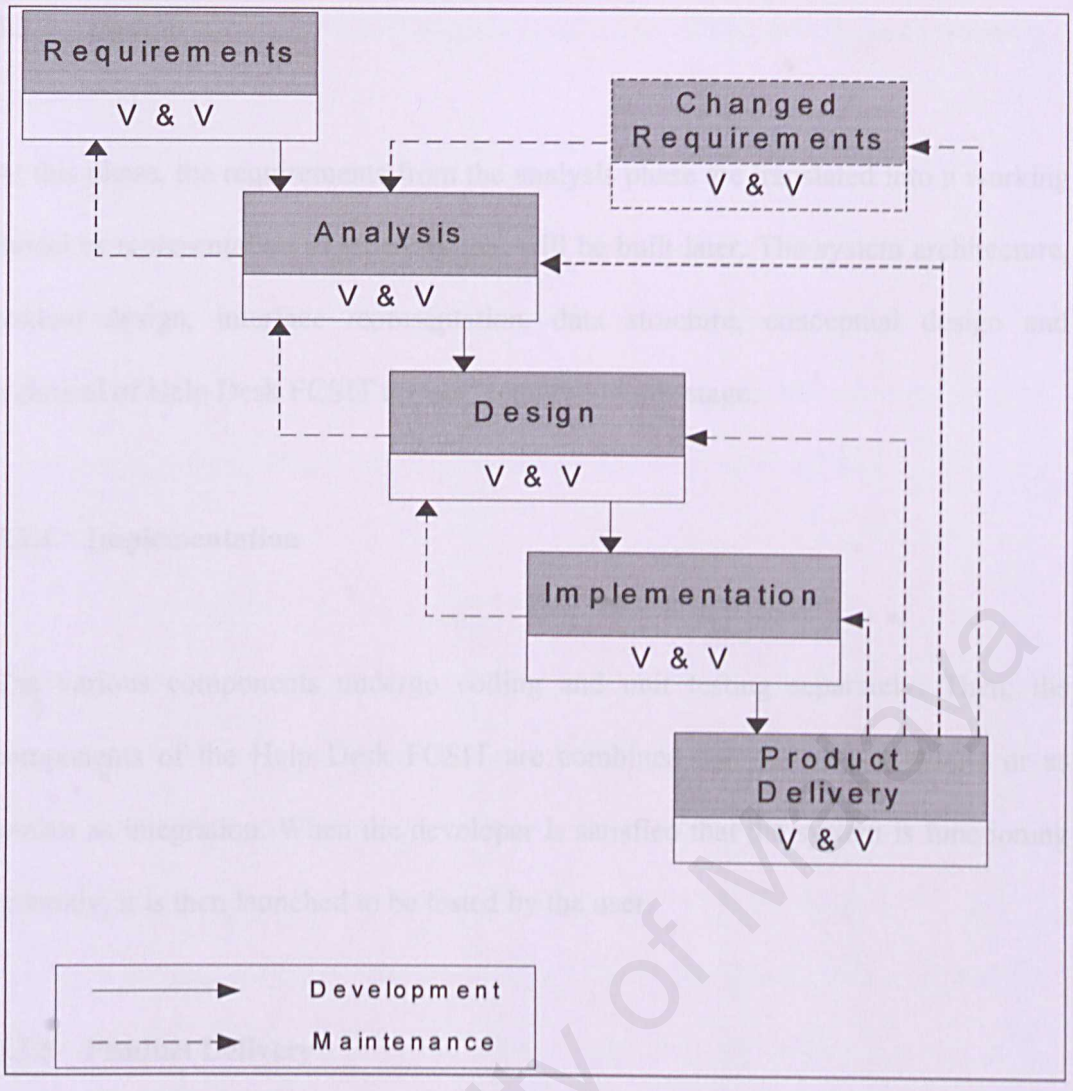


Figure 3-1 : The Waterfall Life-Cycle Model

3.3.2 Analysis

In this phase, the user's requirements are analyzed and it covers the area of functional and non-functional requirements of the Help Desk FCSIT. Besides that, software and hardware requirements during the development stage are also need to be identified.

3.3.3 Design

At this phase, the requirements from the analysis phase are translated into a working model or representation of an entity that will be built later. The system architecture, content design, interface representation, data structure, conceptual design and technical of Help Desk FCSIT are also required in this stage.

3.3.4 Implementation

The various components undergo coding and unit testing separately. Then, the components of the Help Desk FCSIT are combined and tested as a whole or as known as integration. When the developer is satisfied that the system is functioning correctly, it is then launched to be tested by the user.

3.3.5 Product Delivery

The system is delivered to the user only after the acceptance testing is accomplished. Upon acceptance, the Help Desk FCSIT is installed and used.

In addition to the five phases that make up the Waterfall life-cycle model, the other important development activity represented in this model is testing. According to Schach, "inherent in every phase of the Waterfall Model is testing. Testing is not a separate phase to be performed only after the product has been constructed, nor is it not to be performed only at the end of each phase. Instead, testing should proceed continuously throughout the software process." [17]

Testing in the Help Desk FCSIT actually involves two steps, which are verification and validation.

Verification is substantiating that the system has been transformed from one form into another as intended with sufficient accuracy. Verification, in this case is to determine whether the system modules are the actual entities that the design document describes. Verification answers the question such as "Are we building the system *right*?"

Validation is substantiating that the system functions with sufficient accuracy with respect to its requirements specification. Validation answers the question "Are we building the *right* system?"

Testing involving verification and validation should be conducted in parallel with the development of the system. In order to represent the testing activity as an ongoing process, the V&V boxes appear under each product of the Waterfall Model [18].

3.4 Rational for Proposed Methodology

The Waterfall life-cycle model has many advantages. This life cycle is selected to be used in the development of the Help Desk FCSIT because of the following reasons:

i. **Testing is performed in every phase**

Checking a product once it is ready to be delivered to the user is far too late. The Waterfall model performed testing in every phase. Therefore, it is reliable that any fault in the system can be detected earlier.

ii. **Document driven**

Each phase is provided with documentation. The developer will have a guide and reference in developing the system. The documentation must be complete, correct and up to date in parallel with the system development.

iii. **Well structured model**

The Waterfall life-cycle model is well structured. The developer only needs to develop the system step by step according to the structure outlined.

iv. **Ease to use**

This model is ease to use. There are no special skills required in using this approach. This is because all the outlined of the system is already draft out. What the developer needs to do is completed according to what was already structured out.

3.5 Summary

In this chapter, study has been carried out on the proposed Help Desk FCSIT to find out the suitable methodology used. Waterfall Methodology has been chosen in the development of the Help Desk FCSIT.

CHAPTER 4 :
SYSTEM ANALYSIS

CHAPTER 4 : SYSTEM ANALYSIS

4.1 Introduction

The previous chapter has elaborated the justifications for the chosen project methodology; where as in this chapter would further explain the system requirement analysis including the functional requirements, non-functional requirements, and hardware and software requirements on different developing tools.

System analysis is the most critical phase of a system development. System analysis is a problem-solving technique that decomposes a system into its component pieces for the purpose of studying how well those component parts works and interacts to accomplish their purpose [20].

4.2 Functional Requirements

Functional requirement specifies an action that the target product must be able to perform. Functional requirements are often expressed in terms of inputs and outputs [17].

Help Desk FCSIT is divided into three major target users (Figure 4-1). The target users consist of Administrator, Committee and Registered User. The registered user is divided into two categories that are for staff and student. Each target user has different functional requirements. Following are the functional requirements for this system:

4.2.1 Authentication

The initial form that begins the system is Log In form. This form asks the user to log in as an Administrator, Committee or Registered User. Depending on the reply, the system will send the user the respective Main Menus. Different user will receive different main Menus according to their status of User.

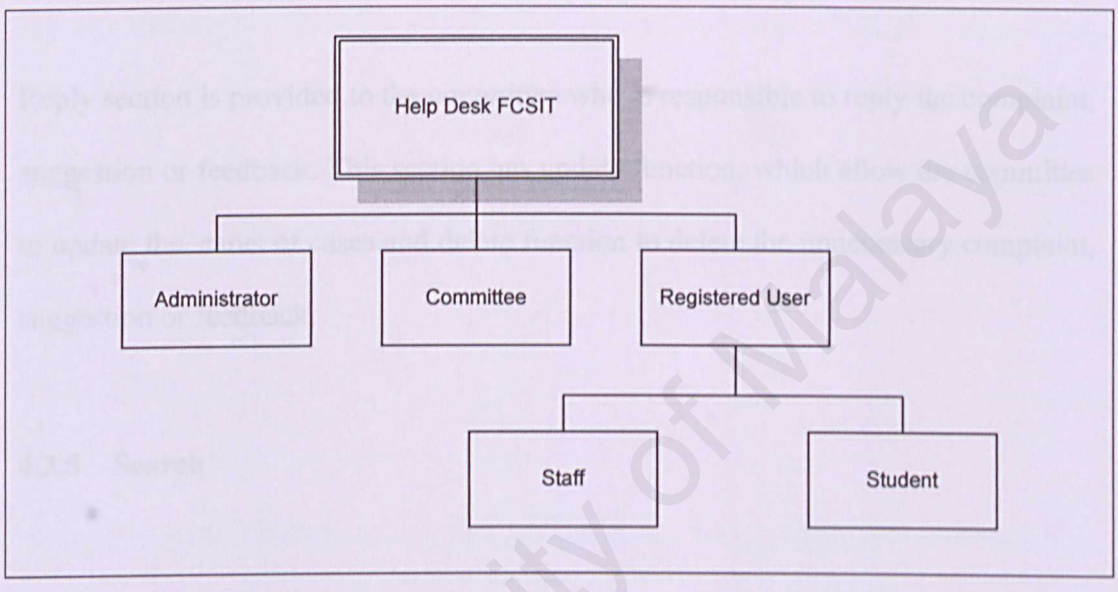


Figure 4-1 : Structure of Help Desk FCSIT

4.2.2 Complaint

The complaint section consists of a form to make a complaint, suggestion or feedback. This module is developed for the registered user only. They need to fill up the form and submit it to the system's administrator.

4.2.3 Personalized Information

This section allows all types of user to update their personal information such as password and address. The new updated information will replace the existing data in the database.

4.2.4 Reply

Reply section is provided to the committee who is responsible to reply the complaint, suggestion or feedback. This section has update function, which allow the committee to update the status of cases and delete function to delete the unnecessary complaint, suggestion or feedback.

4.2.5 Search

This Search section allows the committee to search the complaint, suggestion or feedback by an e-mail address of the user. For administrator, the Search function is provided to search information about the committee.

4.2.6 Statistics

For the administrator, he or she can view the statistic by type or by category of the complaint, suggestion and feedback. The administrator can also choose to display the statistic according to the months. This system is also providing the overall report of the complaint, suggestion and feedback.

For committee, they can only view the statistics by type of complaint, suggestion or feedback. They cannot view statistics and report of others committee.

4.2.7 Committee Maintenance

Committee maintenance is meant to administrator to add, delete and update information about the committee. If there is a new category required to be added to the complaint form, the administrator need to add new committee also. It is same with delete function where if the committee is not appropriate in the system anymore, the administrator has the authority to delete the committee. The information about the committee can also be updated by the administrator.

4.2.8 FAQ

Frequently Asked Questions (FAQ) section is a section where the questions and answers about the use of the Help Desk FCSIT are placed. The administrator is the person who will add, delete and update the FAQ section. While for committee and registered user, they can view and print out the FAQ.

4.3 Non-Functional Requirement

Non-functional requirement is conversed with functional requirement. It specifies properties of the target product itself [17]. The new Help Desk FCSIT must ensure certain web application qualities like user-friendliness, correctness, functionality, reliability, efficiency, security as well as manageability.

4.3.1 User-Friendliness

User interfaces design creates an effective communication medium between a human and a computer. Therefore, it is very important to make sure that the interfaces fulfill user-friendliness so that it would not cause trouble to users. The Golden Rules [21] coins three rules:

i. **Place the user in control**

This will define interaction modes in a way that does not force a user into unnecessary or undesired actions. Besides, it also provides flexible interaction for different users for instance via mouse movement and keyboard commands.

ii. **Reduce the user's memory load**

One of the principles that enable an interface to reduce the user's memory load is by reducing demand on short-term memory. The interface should be designed to reduce the requirements to remember past actions and results.

iii. **Make the interface consistent**

The interface design should apply to consistent fashion where all visual information must be organized according to a design standard that is maintained throughout all screen displays. Apart from that, inputs mechanisms are constrained to a limited set that are used consistently throughout the application. Lastly, mechanisms for navigating from task to task are consistently defined and implemented.

4.3.2 Correctness

A program or system must operate correctly or it provides little value to its users. Correctness is the degree to which the software performs its required function. To ensure this application quality, lots of testing and trial-and-errors will be carried out.

4.3.3 Functionality

The functionalities stressed here are the searching and retrieving capability, which is very important in any web application that deals with data retrieval from existing database. Besides, navigation and browsing features as well as application domain-related features will be taken into account.

4.3.4 Reliability

Reliability is the extent to which a program can be expected to perform its intended function with required precision. It is closely related to correct link processing, error recovery and user input validation and recovery. This quality is essential as it indicates how far users will be confident in the implementation of the new computerized system.

4.3.5 Efficiency

Undeniable, efficiency is the main key for implementing the new help desk management system. Efficiency is understood as the ability of a process procedure to

be called or accessed unlimitedly to produce similar performance outcomes at an acceptable or credible speed [22]. Efficiency is measured based on response time performance, page generation speed and graphics generation speed.

4.3.6 Security

The proposed system has also security measures to minimize the risk of data exposure to unauthorized people. All of the staff and student of FCSIT will receive an e-mail contains of login ID and password from the administrator. The login ID is staff ID or student's matric number. While the password is auto generated from the database. This ID and password is used for log in. By this way, the system can avoid an unauthorized user log in to the Help Desk FCSIT.

4.3.7 Manageability

The modules within the system should be easy to manage. This will make the maintenance and enhancement works simpler and does not time consuming.

4.4 Development Tools And Technologies

The main task in this section is to choose suitable development technologies and programming languages that used to develop the Help Desk FCSIT. After all the requirements have been reviewed and analyzed, the most suitable and appropriate tools for developing the system are identified and selected. The tools to be selected include the development software as well as the entire platform on which the

development of the project is occurred. The major criteria to be considered are not only the suitability of the tools perhaps; the tools to be used must be able to interact with each other. The following programming technology and languages are chose in order to develop the Help Desk FCSIT.

4.4.1 Platform

Microsoft Windows XP is short for Windows Experienced and is the convergence of the two major Microsoft operating systems into one. Windows XP Professional is built on the core software code used in Windows 2000 and Windows NT Workstation. This code, known as the NT kernel, makes Windows XP more powerful, secure, and stable than Windows Me, Windows 98, or Windows 95.

Windows XP is designed more for users who may not be familiar with all of Windows features and has several new abilities to make the Windows experience easier for those users.

Windows XP includes various new features not found in previous versions of Microsoft Windows. Below is a listing of some of these new features:

i. New interface - a completely new look and ability to change the look.

ii. Updates - new feature that automatically obtains updates fro the Internet.

iii. Internet Explorer 6 - Includes Internet Explorer 6 and new IM.

iv. Multilingual support - added support for different languages.

In addition to the above features, Windows XP does increase reliability when compared to previous versions of Microsoft Windows. [23]

4.4.2 Web Server

Microsoft Internet Information Server 5.1 comes together with Windows XP Professional platform. It is stable and is well design for Windows XP. Its friendly GUI features provide easy management and administrative tasks.

IIS 5.1 builds on the features and capabilities needed to deliver Web sites required in an increasingly Internet-centric business environment. And it makes it even easier to use the technologies delivered in prior versions.

4.4.3 Web Browser

Internet Explorer 6.0 comes with Microsoft Windows XP Professional pack. Internet Explorer 6.0 acts as a platform for the development and deployment of Web-based applications. The platform has increased its level of extensibility, supporting script-based behaviors, binary behaviors, as well as layout and custom rendering behaviors. The unique level of extensibility provided by Internet Explorer enables developers to develop rich applications, as well as extend the core functionality provided by Internet Explorer in a reusable manner. IE 6.0 also protected by the latest security improvements.

4.4.4 Programming Language

Active Server Pages (ASP) technology is selected to develop Help Desk FCSIT. For ease of implementation and simplicity, Active Server Pages have been implanted because it acts as a bridge between the server housing the database and the client. Socket connection can be opened on the web server, which then starts the Active Server Pages. The Active Server Pages then connects to the database to handle query and then returns data to the client in a predictable form that can be passed and understood.

Executable scripts can be included directly into HTML files by using Active Server Pages. HTML development and scripting development become the same process, enabling developer to focus directly on the look and feel of the web site, weaving dynamic elements into the pages as appropriate. Active Server Pages applications are:

i. Completed integrated with the HTML files

ii. Easy to create with no manual compiling or linking of programs required

iii. Object-oriented and extensible with ActiveX Server components

The Active Server Pages files are written in VBScript or Jscript that enables the Active Server Pages to receive high-level commands as input. Sophisticated Functionality using ActiveX Server components to process data and generate useful information can be incorporated. This allows the server to deal with larger number of

queries at the same time, as opposed to merely acting as a pass-through query device. Another advantage of Active Server Pages is that it connects to database portability. These Active Server Pages also offer the same functionality as a CGI programs but are much more efficient because of increasing speed time and native ODBC functionality.

4.4.5 Scripting Language

Both VBScript and JavaScript can be used as a client-side programming language and server programming language in Help Desk FCSIT. Client-side programming language is a language that can be interpreted and executed by a browser. While server-side programming language is a language that executes on the browser being used.

Hence, VBScript only best viewed in Microsoft Internet Explorer; therefore it is adopted as a server-side programming language in this project. In addition, VBScript is the default language of Active Server Pages. For the opponent, JavaScript will be used as a client-side programming language because it can work best in the Microsoft Internet Explorer and Netscape Navigator and communicator.

4.4.6 Authoring Tool

Macromedia Dreamweaver MX was really the first full-featured HTML editor that could truly be called a Web page layout program. It combined features familiar to print designers with the functionality and site management capabilities fundamental

for making complex sites work. Dreamweaver MX has evolved to meet the growing needs of Web designers. Besides that, it is very easy to use.

4.4.7 Database Management System

MS Access 2000 and MS SQL Server 2000 are excellent database management software from Microsoft. Both of them are easy to use and are efficient at handling databases. As such, both of them are worthy of consideration for this project.

Access 2000 is suitable for use with small to medium-sized database. However, the size of a database may grow rather huge at times, and this is when Access 2000 starts to reveal its deficiencies. Access 2000 has the tendency of slowing down when the tables in a database gets too huge. This may not be practical for Web sites running huge databases, as the opportunity costs arising from the slowing down of the database may be significant. As such, Access 2000 is unsuitable for the management of huge databases.

SQL Server 2000, on the other hand, works well with databases of any size. It contains all the user-friendly features of Access 2000, yet it works so many times more efficiently than Access 2000. It has the ability of handling hundreds of transactions simultaneously without affecting performance.

SQL Server 2000 will therefore be chosen to act as the database management software for the development of Help Desk FCSIT.

4.5 Hardware and Software Requirements

The recommended hardware configurations for Help Desk FCSIT are as below in Table 4-1 :

Table 4-1 : Hardware Requirements

Component	Description
Microprocessor	Intel Pentium 300 MHz and above
RAM	128 MB RAM and above (64 MB minimum supported; may limit performance and some features)
Storage	1.5 gigabytes (GB) of available hard disk space

The software requirements for installing Help Desk FCSIT are summarized in Table 4-2.

Table 4-2 : Software Requirements

Description	Technologies/Software
Operating system/ Platform	Windows XP Professional
Web server	Internet Information Server 5.1
Web development technology	Active Server Pages (ASP)
Web authoring tool	Macromedia Dreamweaver MX
Markup and scripting language	HTML, VBScript, JavaScript
Database	Microsoft SQL Server 2000

4.6 Summary

In this chapter, system requirements such as functional requirements and non-functional requirements have been investigated and analysed. The development programming and technologies tools have been rectified as well. As the result, Windows XP Professional has been chosen as the system platform, IIS as the web server, ASP as the development technology, Macromedia Dreamweaver MX as the web-authoring tool, HTML, VBScript and JavaScript as the scripting language and the Microsoft SQL Server 2000 as the database management system.

CHAPTER 5 :

SYSTEM DESIGN

CHAPTER 5 : SYSTEM DESIGN

5.1 Introduction

The system design will only take place after the system analysis phase. This chapter will explain the system design of this project. System design is defined as those tasks that focus on the specification of a detailed computer-based solution. It is also called physical design [20]. Systems design focuses on the technical or implementation concerns of the system. System design includes the following issues such as flow of the system, database design and user interface design.

5.2 System Architecture

System architecture refers to the architecture of a specific construction or system. System architecture corresponds to "architecture as a product." It is the result of a design process for a specific system and specifies the functions of components, their interfaces, their interactions, and constraints. This specification is the basis for detailed design and implementation steps.

The Help Desk FCSIT is based on the three-tier client/server architecture. Three-tier is the new growth area for client/server computing because it meets the requirements of large-scale Internet and intranet client/server applications.

5.3 Process Flow Diagram

Process flow diagram depicts how the process is done in this system. Figure 5-1 shows the process flow of the Help Desk FCSIT.

Firstly, the system requires the user to fill up the complaint form. By clicking on the submit button, the respective committee will receive the complaint, suggestion or feedback automatically based on their category and assign a team for investigation. The user will then receive an e-mail from the system as an acknowledgement that the system had received the complaint, suggestion or feedback.

After the committee received the complaint, suggestion or feedback, they will reply to the cases using the reply form. They will also update the status of the cases received and action taken by the committee. Then, the committee will proceed with the investigation on the cases reported. After the investigation is completed and necessary steps are taken to reconcile the problems, the committee will again update the status of cases. Finally, the system will generate a report and also statistics for the committee and administrator.

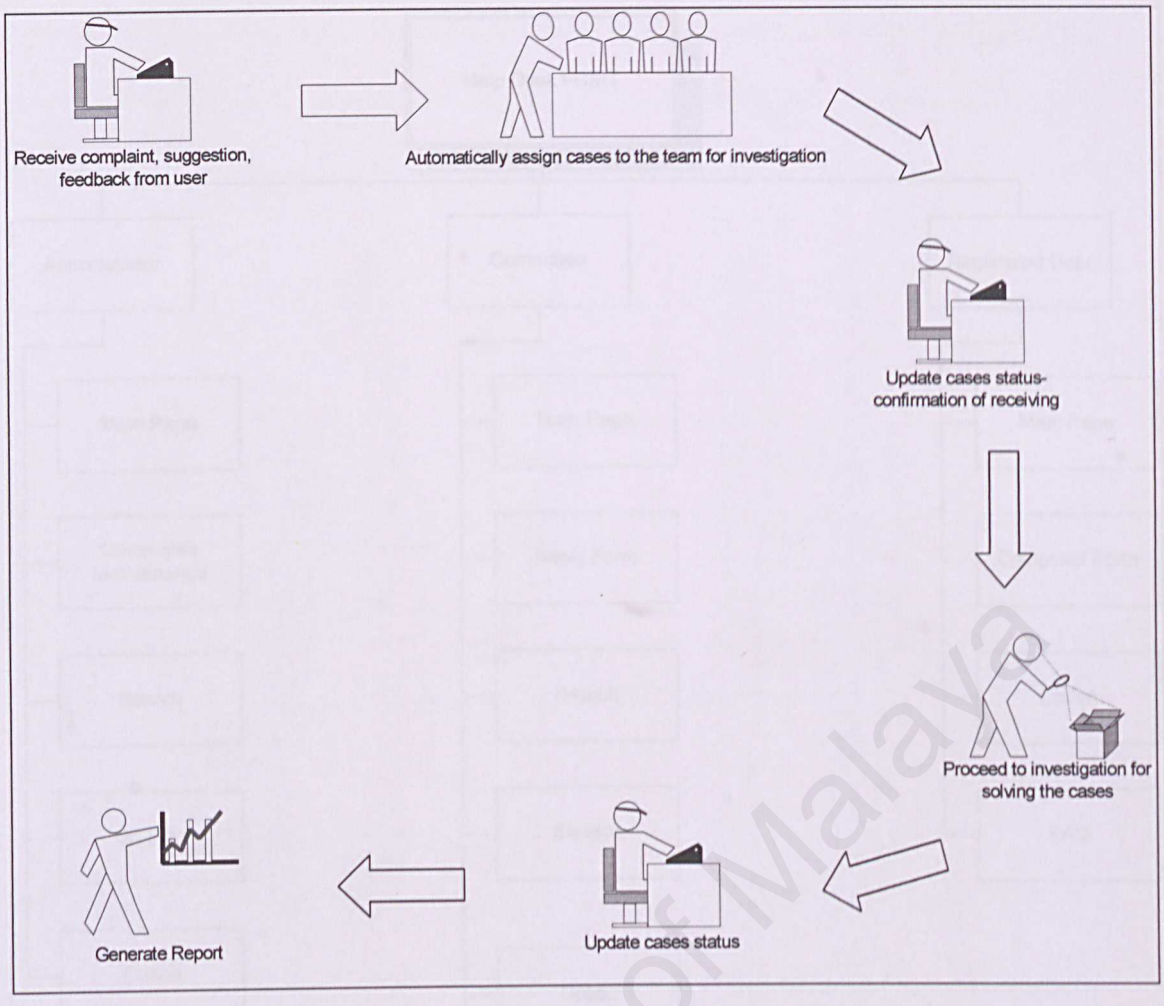


Figure 5-1 : Process flow of Help Desk FCSIT

5.4 System Structure Chart

The Help Desk FCSIT consists of three types of users, which are the Administrator, the Committee and the registered user. Meanwhile for the registered users, they are divided into two categories - the Staff and the Student. The system structure for the Help Desk FCSIT is shown in Figure 5-2 below.

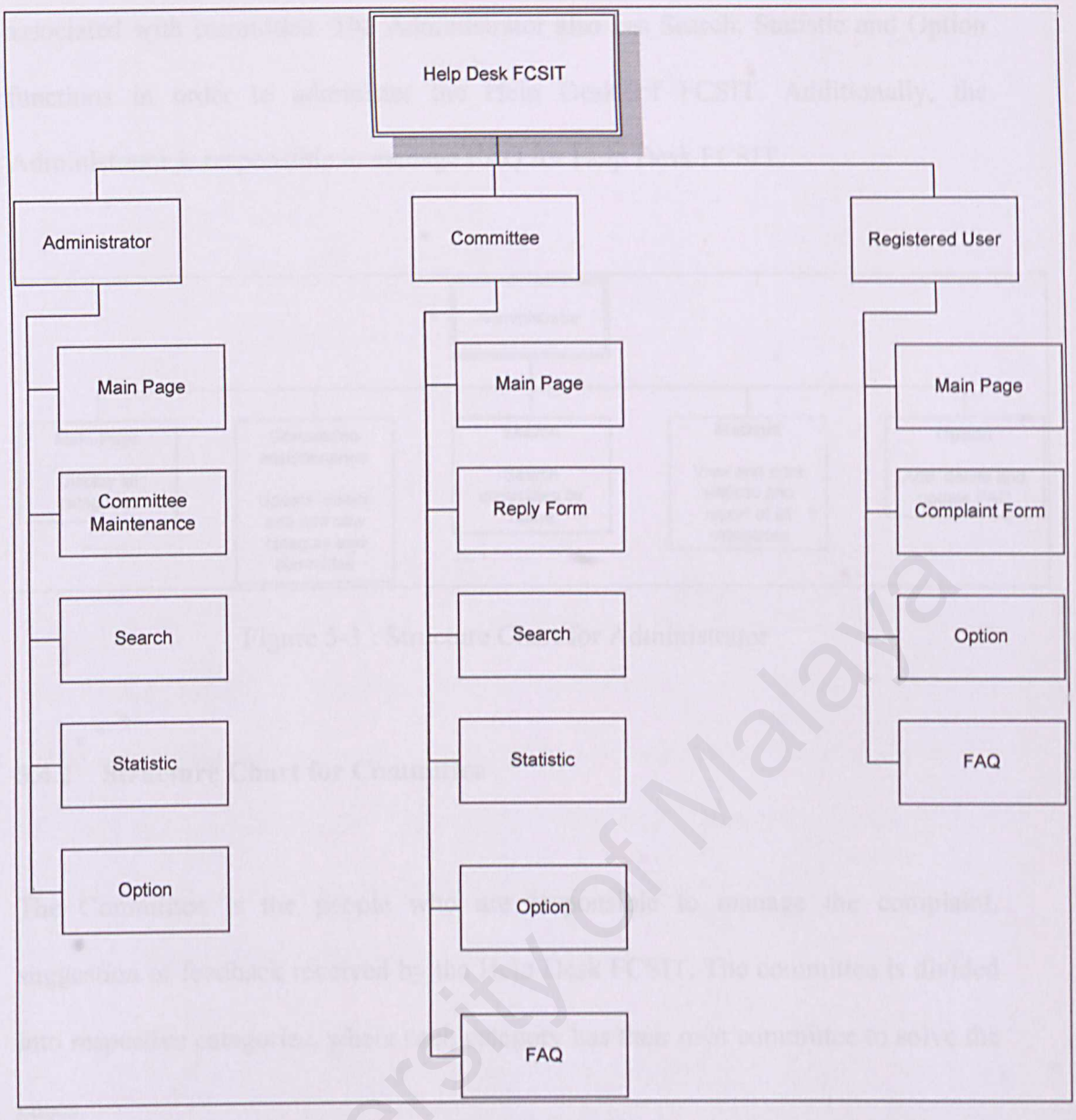


Figure 5-2 : Structure Chart for Help Desk FCSIT

5.4.1 Structure Chart for Administrator

Basically, the administrator's main task is to manage the Help Desk FCSIT, in terms of the database and e-mail systems. The Administrator has five major tasks as shown in Figure 5-3. The first task is to manage the main page where all the categories in the web page should be displayed. The Administrator is allowed to update, delete and add new category and committee in the web page. However, the Administrator is not

associated with committee. The Administrator also has Search, Statistic and Option functions in order to administer the Help Desk of FCSIT. Additionally, the Administrator is responsible to manage FAQ for Help Desk FCSIT.

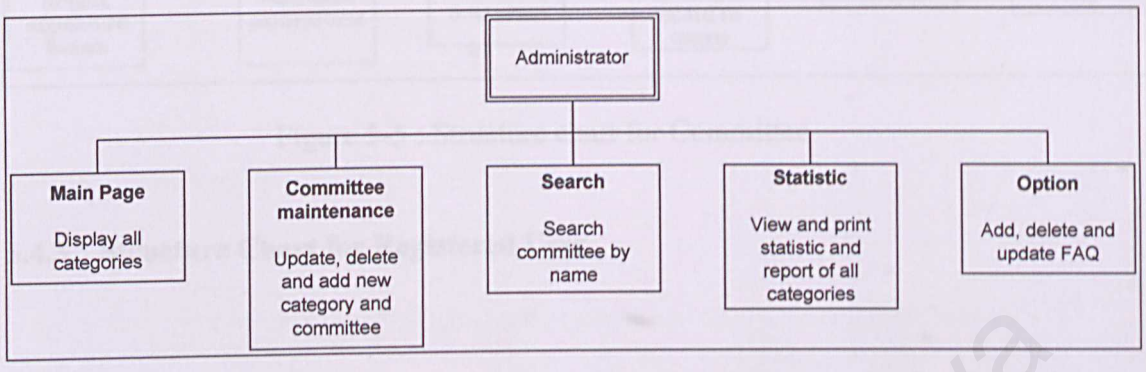


Figure 5-3 : Structure Chart for Administrator

5.4.2 Structure Chart for Committee

The Committee is the people who are responsible to manage the complaint, suggestion or feedback received by the Help Desk FCSIT. The committee is divided into respective categories, where each category has their own committee to solve the cases.

As shown in Figure 5-4, the committee has six menus: Main page, Reply Form, Search function, Statistic, Option and FAQ.

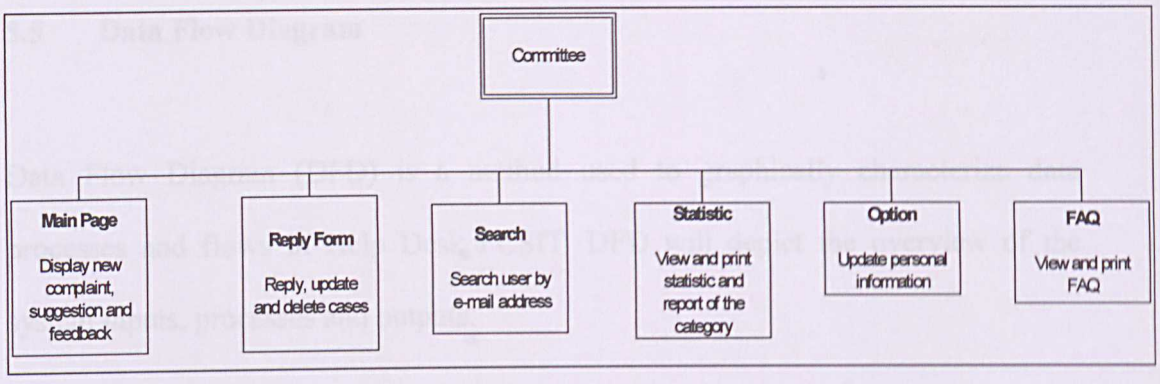


Figure 5-5 : Structure chart for Committee

5.4.3 Structure Chart for Registered User

Help Desk FCSIT has two types of registered user, which are the Staff and the Student. All the Staff and Student of FCSIT will receive an e-mail consists of log in id and password. The password is generated by the database system. The menus for the user, as shown in Figure 5-5, are Main Page that displays the status of their complaint, suggestion or feedback, form to submit complaint, Option for personalized or update personal information and FAQ.

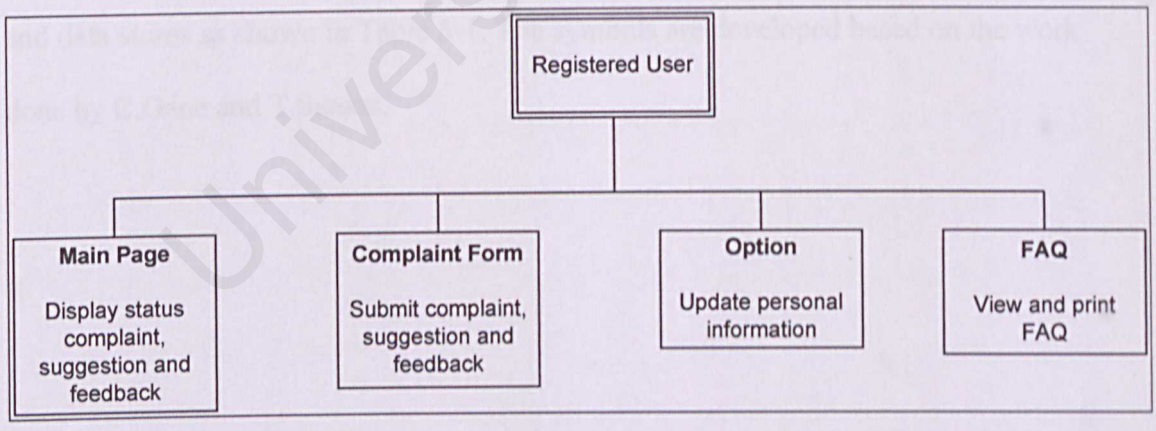


Figure 5-6 : Structure Chart for Registered User

5.5 Data Flow Diagram

Data Flow Diagram (DFD) is a method used to graphically characterize data processes and flows in Help Desk FCSIT. DFD will depict the overview of the system inputs, processes and outputs.

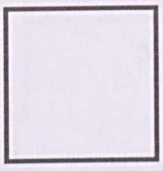
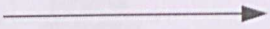
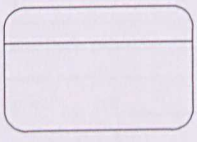
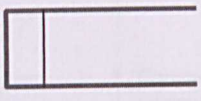
The advantages of using DFD are:

i. Further understanding of the interrelation of modules and sub modules of Help Desk FCSIT.

ii. Analysis of a proposed system to determine if the necessary data and processes have been defined.

DFD is easy to be understood as it has symbols that specify the physical aspects of implementation. There are four basic symbols in DFD: entity, flow of data, process and data stores as shown in Table 5-1. The symbols are developed based on the work done by C.Gane and T.Sarson.

Table 5-1 : DFD Symbols

Symbols	Meaning
	External Agent
	Data Flow
	Process
	Data Store

5.5.1 Context Data Flow Diagram

Top-down approach is adopted in diagramming DFD. DFD drawing is started from general to specific. Initially, a context level diagram of the proposed Help Desk FCSIT is drawn, as show in Figure 5-6. The context diagram in Figure 5-6 is the most general diagram. The context data flow diagram contains only one process. The data flows define the interactions of Help Desk FCSIT within the boundaries and with the external data stores.

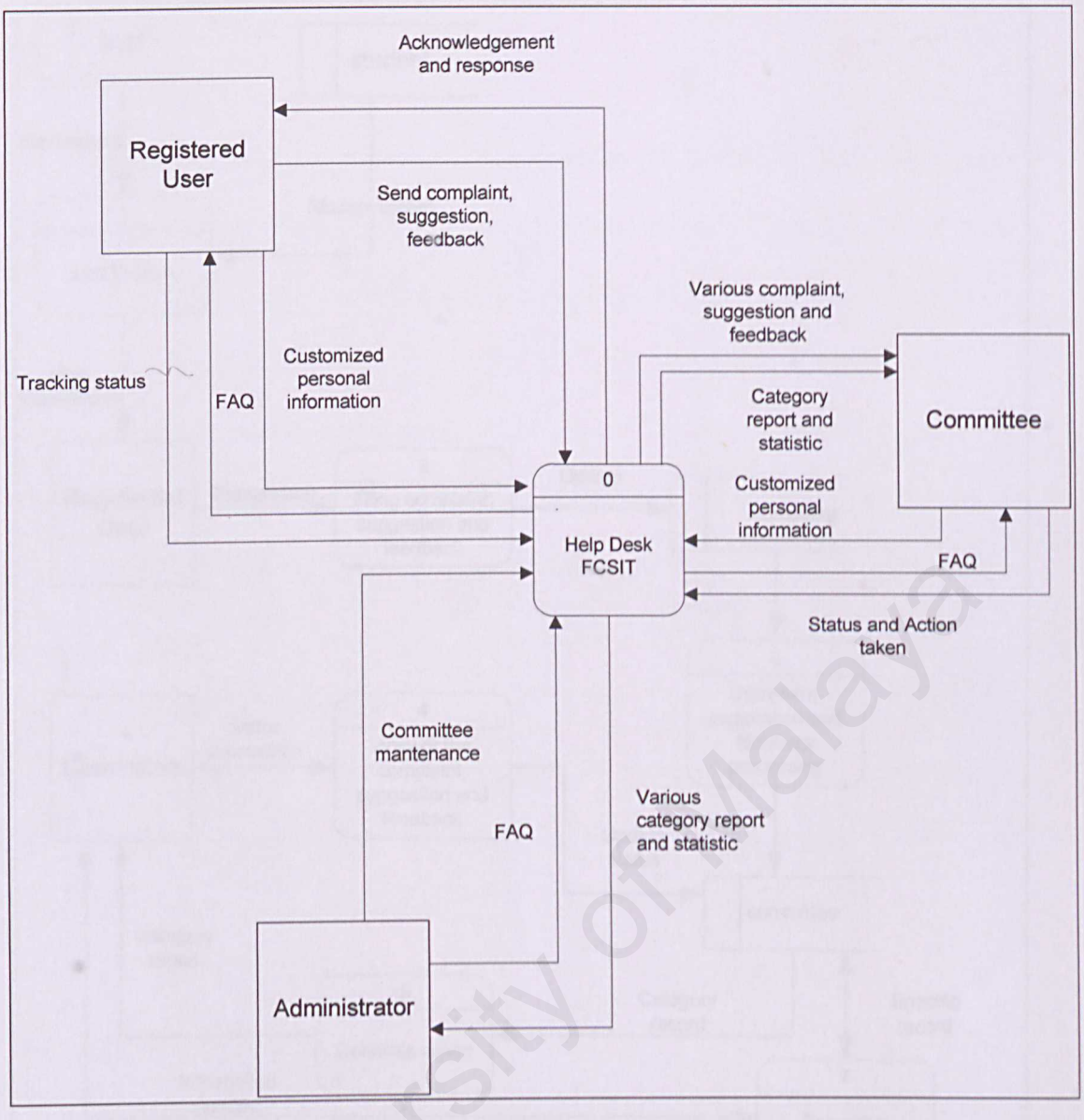


Figure 5-6 : Context Data Flow Diagram for Help Desk FCSIT

5.5.2 Diagram 0

Diagram 0 is an overview process of all the major modules in Help Desk FCSIT that includes all the data stores, external agents and processes involved as shown in Figure 5-7.

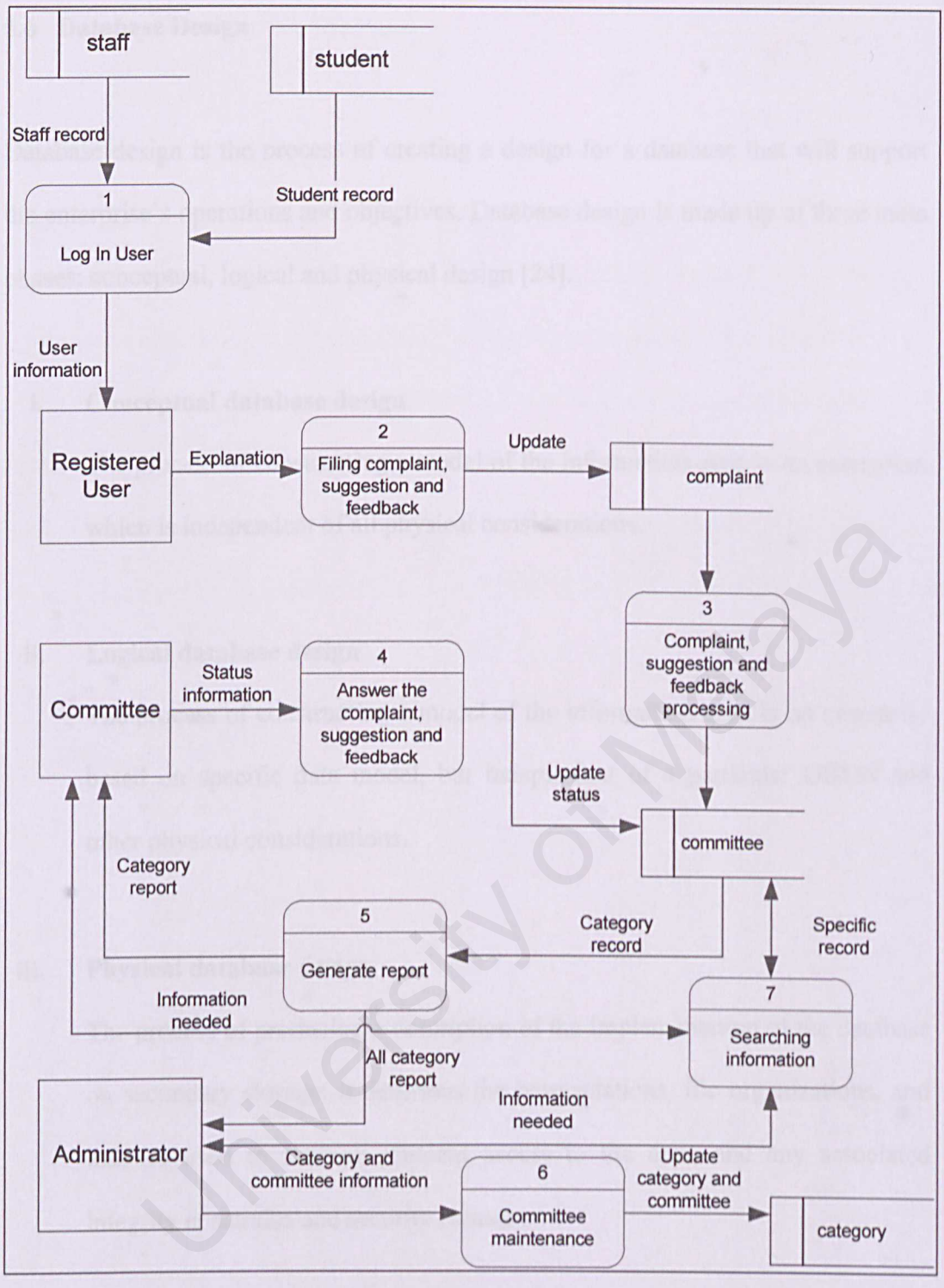


Figure 5-7 : Diagram 0 for Help Desk FCSIT

5.6 Database Design

Database design is the process of creating a design for a database that will support the enterprise's operations and objectives. Database design is made up of three main phases: conceptual, logical and physical design [24].

i. **Conceptual database design**

The process of constructing a model of the information used in an enterprise, which is independent of all physical considerations.

ii. **Logical database design**

The process of constructing a model of the information used in an enterprise based on specific data model, but independent of a particular DBMS and other physical considerations.

iii. **Physical database design**

The process of producing a description of the implementation of the database on secondary storage; it describes the base relations, file organizations, and indexes used to achieve efficient access to the data, and any associated integrity constrains and security measures.

5.6.1 Entity-Relationship Diagram

One of the most difficult aspects of database design is the fact that designers, programmers, and end-users tend to view data and its use in different ways. To ensure getting a precise understanding of the nature of the data and how it is used by enterprise, a model for communication that is non-technical and free ambiguities is needed. The Entity-Relationship (ER) model is one such example. ER modeling is a top-down approach to database design that begins by identifying the important data called entities and relationships between the data that must be represented in the model [24]. The ER diagram for Help Desk FCSIT is shown below in Figure 5-8.

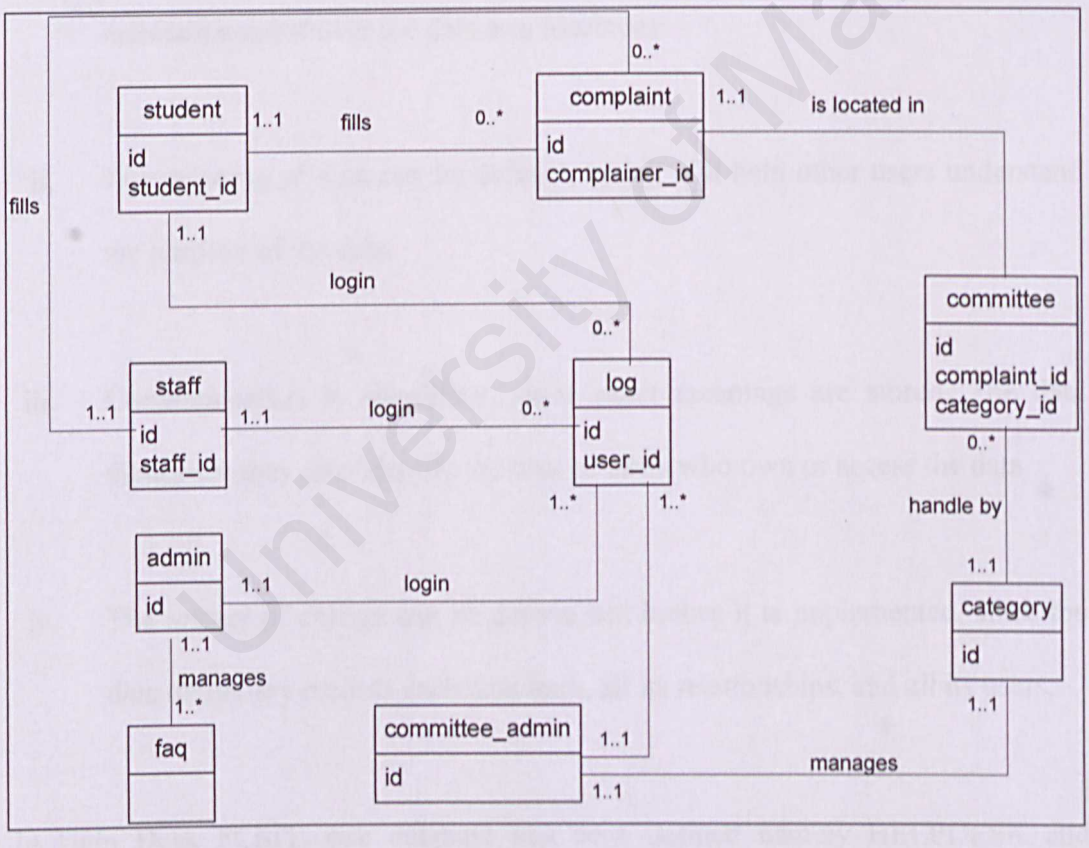


Figure 5-8 : An Entity-Relationship (ER) diagram of the Help Desk FCSIT

5.6.2 Data dictionary

A database must furnish a catalog in which description of data items are stored and which is accessible to users. Data dictionary or system catalog is a repository of information describing the data in the database: it is the 'data about data' or meta-data. The amount of information and the way the information is used vary with the database management system (DBMS) [24].

Some benefits of a data dictionary are [24] :

i. Information about data can be collected and stored centrally. This helps to maintain control over the data as a resources

ii. The meaning of data can be defined, which will help other users understand the purpose of the data.

iii. Communication is simplified, since exact meanings are stored. The data dictionary may also identify the user or users who own or access the data

iv. The impact of change can be determined before it is implemented, since the data dictionary records each data item, all its relationships, and all its users.

In Help Desk FCSIT, one database has been defined namely HELPDESK and contained ten tables. The tables are student, staff, complaint, admin, committee_admin, committee, category, log, faq and check_register.

i. Table name : **student**

Table 5-2 : Table of student

Field Name	Data Type	Length	Description
id	int	4	The number of student
student_id	varchar	50	Student's matric number
name	text	16	Student's name
address	text	16	Student's address
telephone	text	16	Student's telephone number (optional)
email	text	16	Student's e-mail address
password	varchar	50	Student's password
status	varchar	50	Active or not active

ii. Table name : **staff**

Table 5-3 : Table of staff

Field Name	Data Type	Length	Description
id	int	4	The number of staff
staff_id	varchar	50	Staff's id
name	text	16	Staff's name
address	text	16	Staff's address
telephone	text	16	Staff's telephone number (optional)
email	text	16	Staff's e-mail address
password	varchar	40	Staff's password
status	varchar	50	Active or not active

iii. Table name : **complaint**

Table 5-4 : Table of complaint

Field Name	Data Type	Length	Description
id	int	4	Number of complaint, suggestion or feedback
date	varchar	50	Date the complaint made
time	varchar	50	Time the complaint made
complainer_id	int	4	staff_id or student_id
complainer_status	varchar	10	Staff or student
category	varchar	50	Category for complaint, suggestion or feedback
explanation	text	16	Explanation of the complaint, suggestion or feedback

iv. Table name : **admin**

Table 5-5 : Table of admin

Field Name	Data Type	Length	Description
id	int	4	ID for administrator
login_id	varchar	50	Login ID for administrator
password	varchar	50	Password for administrator

v. Table name : **committee_admin**

Table 5-6 : Table of committee_admin

Field Name	Data Type	Length	Description
id	int	4	ID for administrator for each committee
login_id	varchar	50	Login ID for administrator for each committee
password	varchar	50	Password for administrator for each committee
category	varchar	50	Category for respective committee

vi. Table name : **committee**

Table 5-7 : Table of committee

Field Name	Data Type	Length	Description
id	int	4	Number of cases in investigation
category_id	int	4	ID for the respective category
date_received	varchar	50	The date of receiving the cases
time_received	varchar	50	The time of receiving the cases
complaint_id	int	4	ID of the complaint, suggestion and feedback
result	text	16	The action taken by the committee
status	varchar	50	Complete or incomplete
date_approved	varchar	50	Date the cases complete
time_approved	varchar	50	Time the cases complete
approved_by	text	16	The person in charge

vii. Table name : **category**

Table 5-8 : Table of category

Field Name	Data Type	Length	Description
id	int	4	The ID for category
category_name	varchar	50	The name of category

viii. Table name : **log**

Table 5-9 : Table of log

Field Name	Data Type	Length	Description
id	int	4	The number of users
user_id	varchar	50	User ID
user_status	varchar	50	Student, staff, administrator or committee
ip_address	ntext	16	IP address
start_date	varchar	50	The date the user log in
stop_date	varchar	50	The date the user log out
start_time	varchar	50	The time the user log in
stop_time	varchar	50	The time the user log out
online_status	varchar	50	The status of user whether Online or Offline

ix. Table name : **faq**

Table 5-10 : Table of faq

Field Name	Data Type	Length	Description
id	int	4	ID for each question and answer
question	text	16	Questions for FAQ
answer	text	16	Answers for FAQ

x. Table name : **check_register**

Table 5-10 : Table of check_register

Field Name	Data Type	Length	Description
id	int	4	ID for each user ID
check_id	varchar	50	Check user ID

5.7 User Interface Design

The Help Desk FCSIT is a web-based system. Thus the Graphical User Interfaces (GUIs) play an increasingly important role of the system. The web pages will be designed so that it is easy to use, and a user does not have to require training and minimal support.

Basically, the user interface designs for Help Desk FCSIT are as follow in Figure 5-9, Figure 5-10 and Figure 5-11. Figure 5-9 shows the page where the user chooses their category and click on OK button. While Figure 5-10 shows the next interface where the user need to type their log in ID and password before they can enter into the system. Lastly, in Figure 5-11, the page for each type of user will be display.

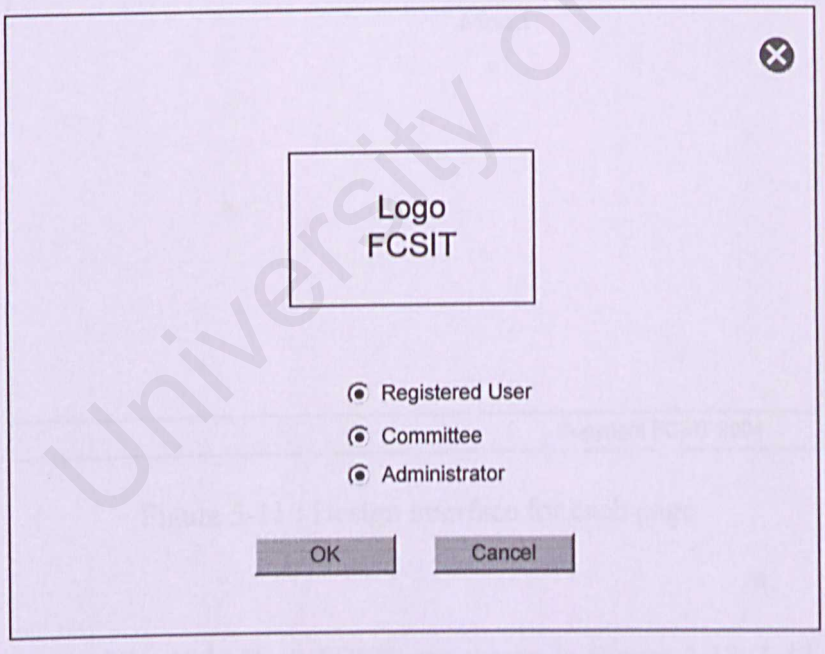


Figure 5-9 : Design interface for choosing the type of user

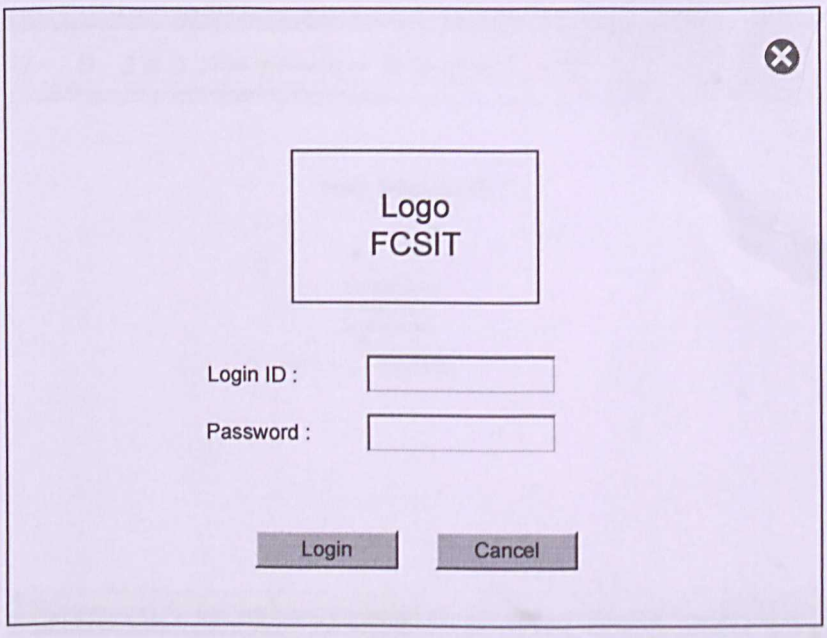


Figure 5-10 : Design interface for login page

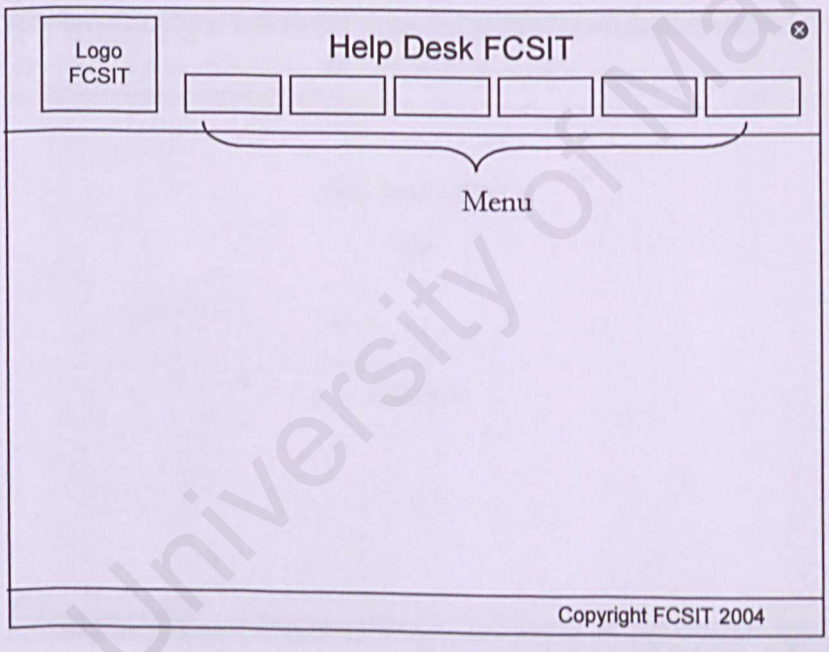


Figure 5-11 : Design interface for each page

The prototypes of the Help Desk FCSIT are shown in Figure 5-12, 5-13, 5-14,5-15 and 5-16 below.

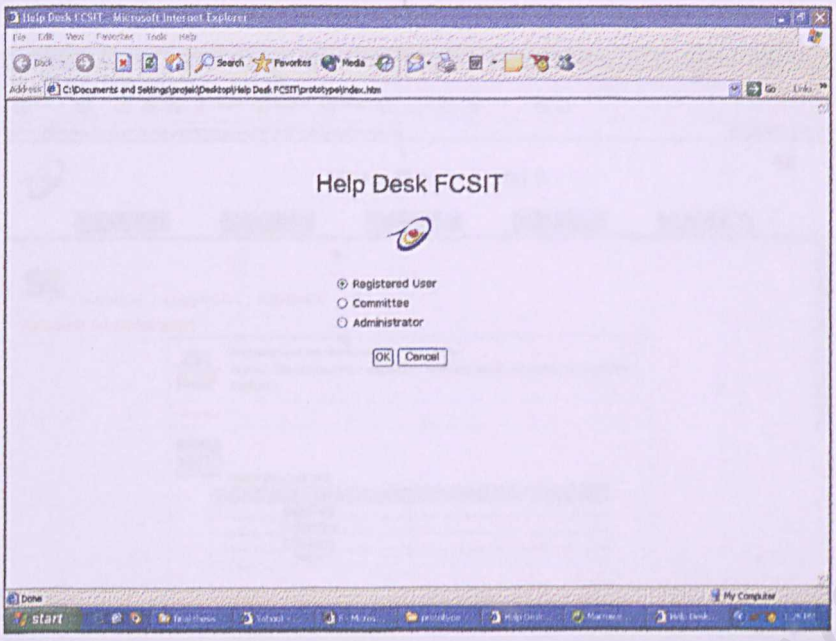


Figure 5-12 : First interface for select type of user

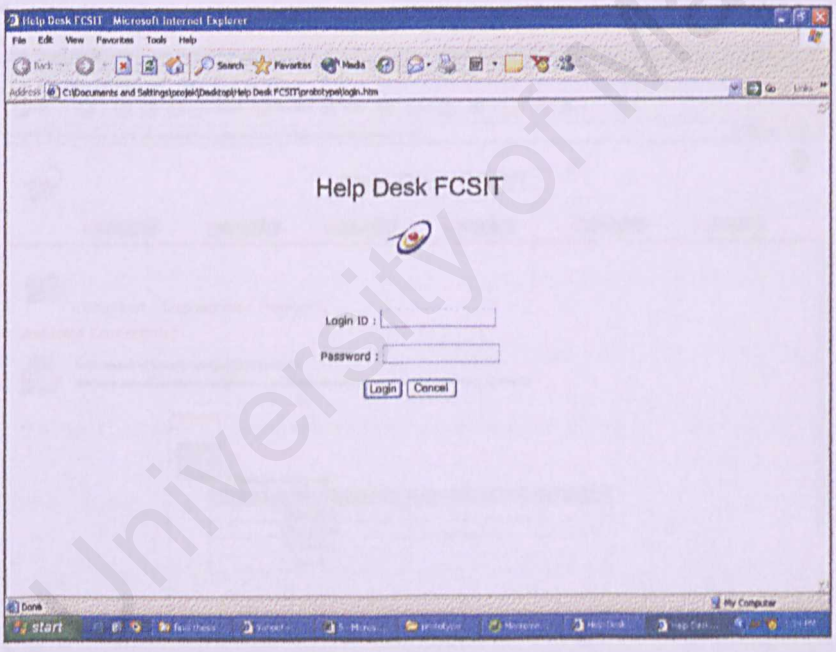


Figure 5-13 : Log in Interface

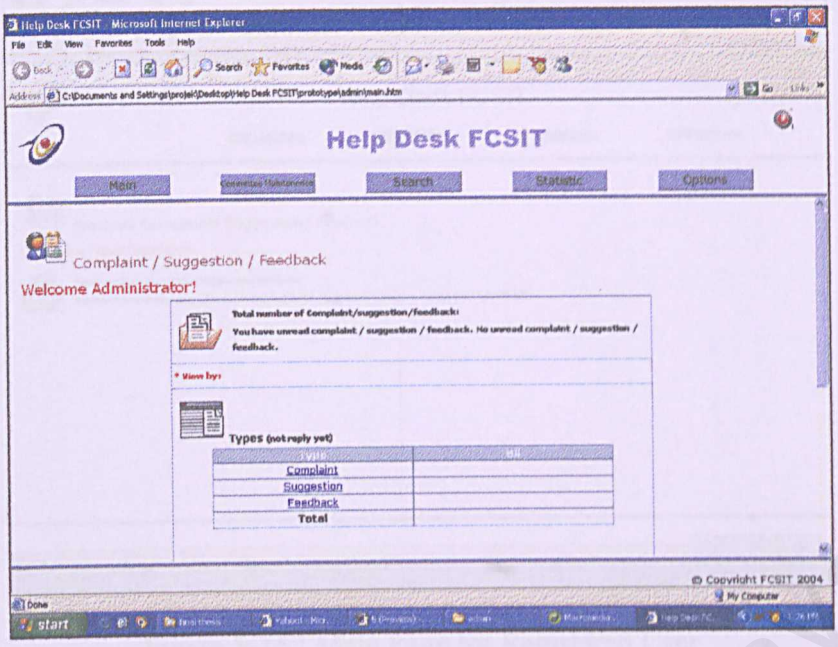


Figure 5-14 : Main page for Administrator

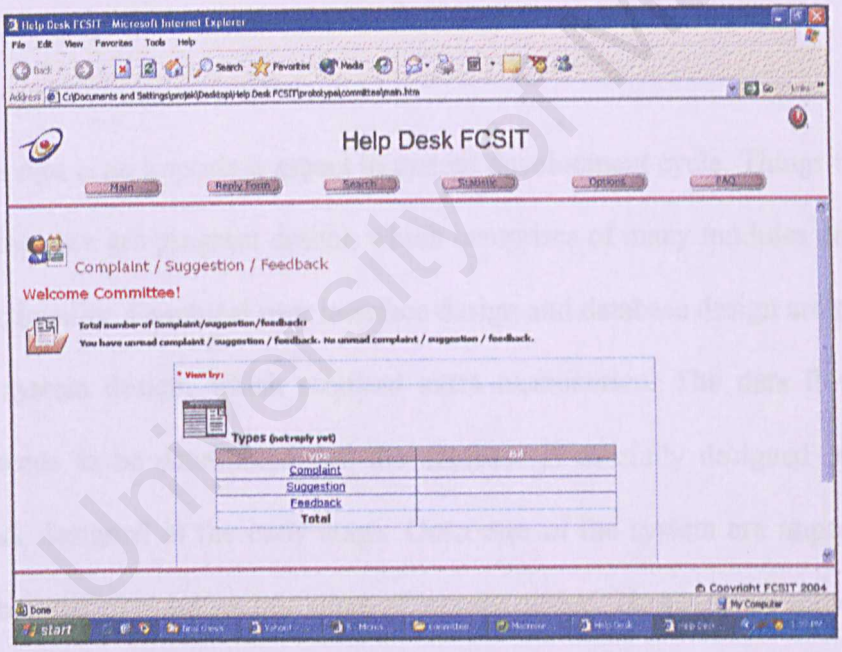


Figure 5-15 : Main Page for Committee

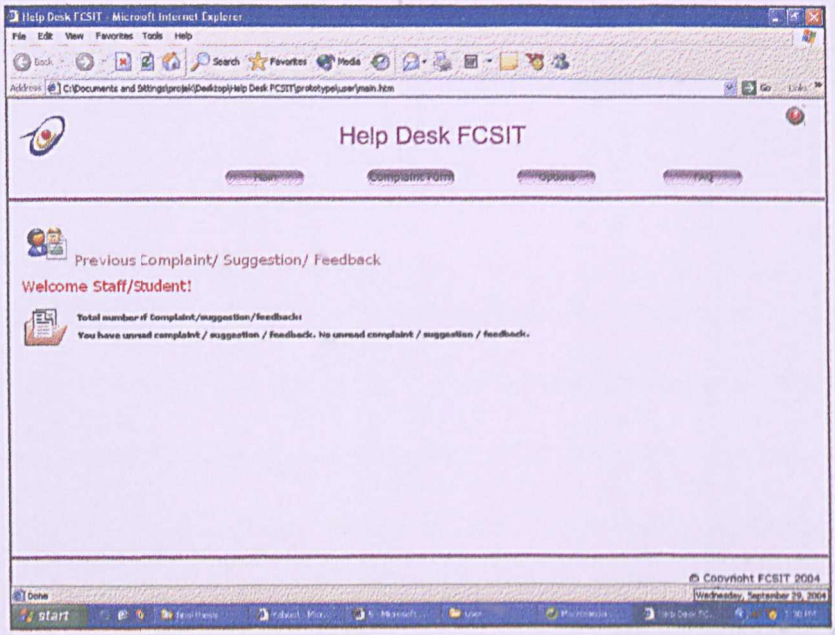


Figure 5-16 : Main Page for Registered User

5.8 Summary

System design is an important aspect in system development cycle. Things that need to be taking care are program design, which comprises of many modules defined by their functionality. Graphical user interface design and database design are two other parts in system design, which required extra examination. The data flow of the system needs to be determined and the database is carefully designed out of the framework designed in the early stage. Outcomes of the system are important and need to be predicted during this stage of system design. However, all these designs might need minor or major changes, as there is no promise that all the primary designs are good and perfect.

CHAPTER 6 :

SYSTEM IMPLEMENTATION

CHAPTER 6 : SYSTEM IMPLEMENTATION

6.1 Introduction

System implementation in a system development is the process of translating the detailed design into code. The aim of the system implementation is to implement the target system product in the chosen implementation language. The initial stage of system implementation involves setting up the development environment. This includes setting up development tools to facilitate the system implementation.

6.2 Development Environment

During the Help Desk FCSIT system development, a vast array of hardware and software tools was used. Table 6-1 and 6-2 below depicts the hardware and software used to develop the system.

Table 6-1 : Hardware Requirements

Component	Description
Microprocessor	Intel Pentium 300 MHz and above
RAM	128 MB RAM and above
Storage	1.5 gigabytes (GB) of available hard disk space

Table 6-2 : Software Requirements

Technologies/Software	Description	Purpose
Windows XP Professional	Operating system/ Platform	System requirement
Internet Information Server 5.1	Web server	Web server host
Active Server Pages	Web development technology	Programming language to build dynamic page
Macromedia Dreamweaver MX	Web authoring tool	Setup and design layout.
Notepad	Web authoring tool	Edit code
Macromedia Fireworks MX 2004	Web designing tool	Design and create animation
Swift 3D	Graphics animation tool	Design logo and icon
HTML, VBScript, JavaScript	Markup and scripting language	Coding the web pages
Microsoft SQL Server 2000	Database Management System	Build the database to store and manipulate data

6.3 Development Of The System

The development of the system includes setting up the database system, developing the application and connecting the application to the database.

6.3.1 Database development

The database for the Help Desk FCSIT is created using the SQL Server Enterprise Manager that comes with the installation of MS SQL Server 2000. The database was developed according to the logical data model created during the system design phase. However, as the development of the system progresed, changes were also made to the database design to sui the need of the proposed system.

An empty database was created first, named **helpdesk.** All the tables were then created by specifiying the field names, data type and length. Primary key was set for each table in the database.

6.3.2 Application development

Active Server Pages (ASP) is a *server-side scripting* environment that can be used to create and run dynamic, interactive Web server applications. With ASP, HTML pages, script commands, and COM components can be combined to create interactive Web pages or powerful Web-based applications, which are easy to develop and modify.

A server-side script begins to run when a browser requests an .asp file from the Web server. The Web server then calls ASP, which processes the requested file from top to bottom, executes any script commands, and sends a Web page to the browser.

Because of the scripts run on the server rather than on the client, the Web server does all the work involved in generating the HTML pages sent to browsers. Server-side scripts cannot be readily copied because only the result of the script is returned to the browser. Users cannot view the script commands that created the page they are viewing.

6.3.3 Macromedia Fireworks MX 2004

Macromedia Fireworks MX 2004 is the definitive solution for professional web graphics design and production. It is the first production environment to address and solve the special challenges facing web graphics designers and developers.

Fireworks can be used to create, edit, and animate web graphics, add advanced interactivity, and optimize images in a professional environment. In Fireworks, create and edit bitmap and vector graphics can be done in a single application. Everything is editable, all the time. And the workflow can be utomate to meet the demands of time-consuming updates and changes. Below in Figure 6-1 is the layout design in grid using Macromedia Fireworks MX 2004 for the interface of Help Desk FCSIT.

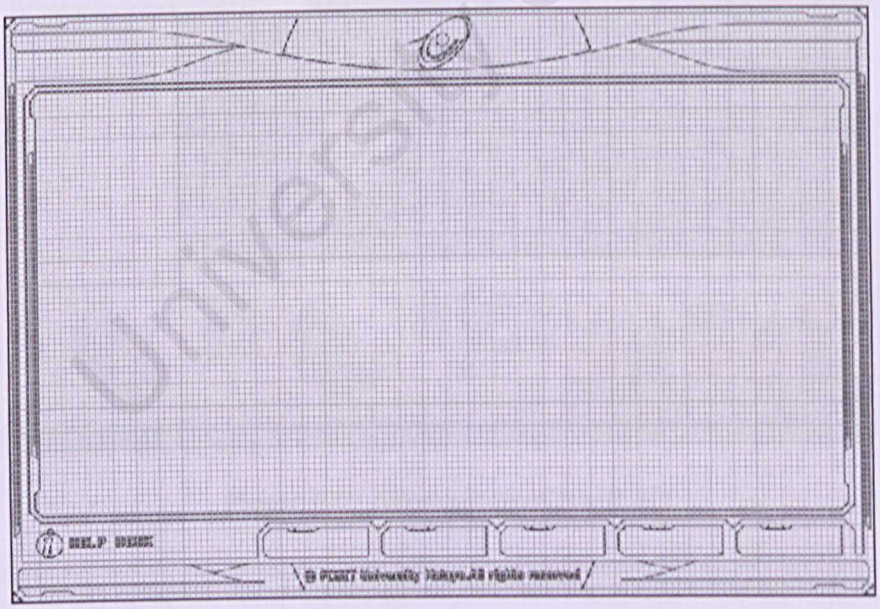


Figure 6-1 : Layout design for Help Desk FCSIT

6.3.4 Swift 3D

In order to make this project more interactive, 3D elements were included. The development tool is Swift 3D. Swift 3D provides a toolset and interface that allows anyone to quickly learn the basics of 3D modeling and animation while providing plenty of room to grow into a full set of advanced 3D tools. But it's Swift 3D's remarkable vector and raster rendering capabilities that have made it the tool of choice for beginners and experts alike. Figure 6-2 below show the workspace in Swift 3D and Figure 6-3 shows the process of generating all frames.

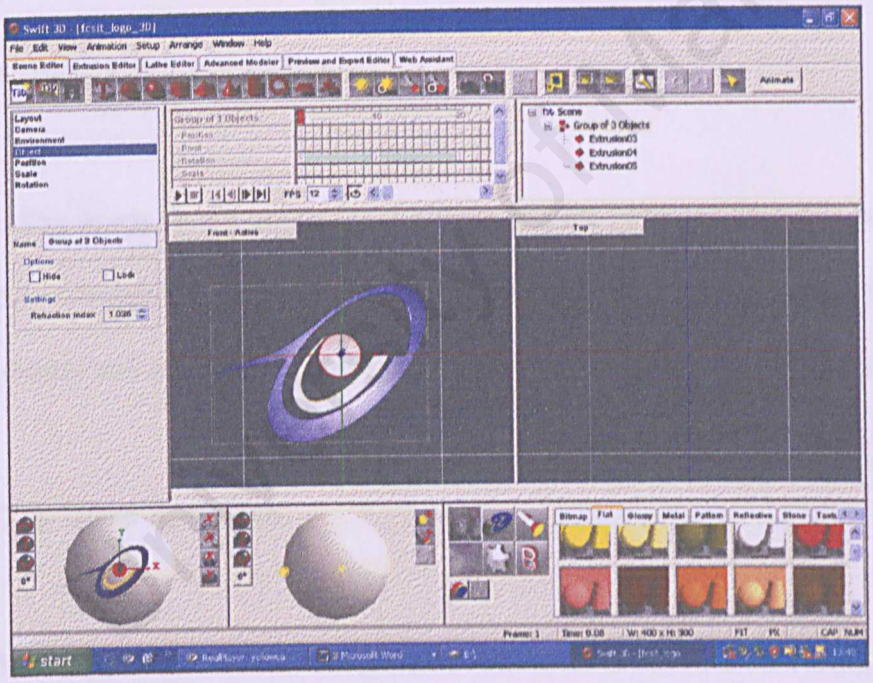


Figure 6-2: Workspace of Swift 3D

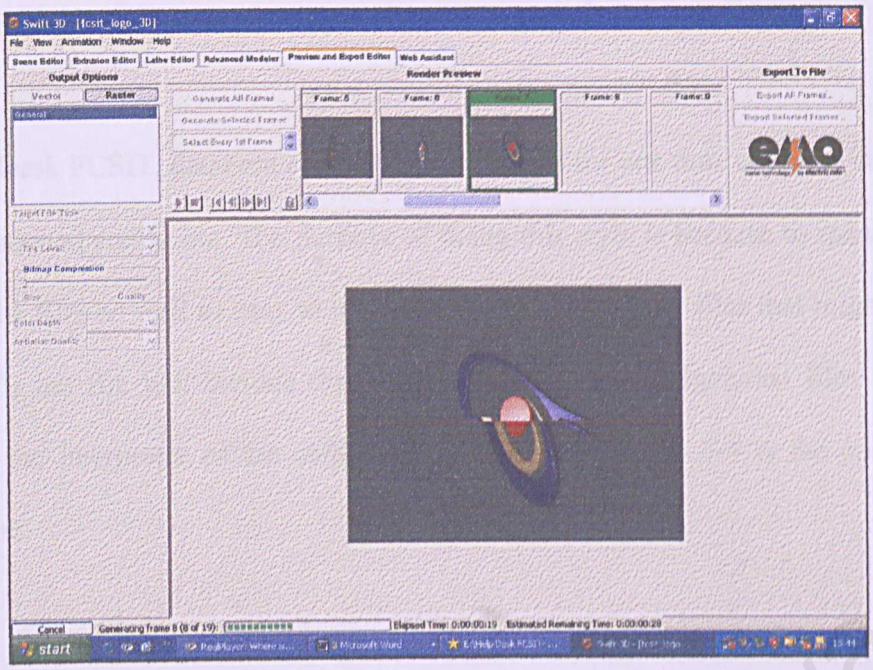


Figure 6-3 : Process of generating all frames

6.4 Coding Approach and Style

Good coding practices are important to ensure that the proposed system's coding is consistent, maintainable and easy to read. This will not only make the program written clearer and more understandable, but also easy to maintain in the future.

6.4.1 Top Down Approach

Top-down approach is chosen to break the big modules of Help Desk FCSIT into functions and procedures. All these small modules or functions are built and developed separately.

6.4.2 Database Connection

In Help Desk FCSIT, database connection is declared for one time only and stored in one file named config.inc. The purpose of doing this style is because to reduced the size of the files as well as easy to modify that page. The other files that might using the config.inc file will include that file by using #include. Included files will go through the interpreter of the calling page. Below in Figure 6-4 is the coding of config.inc.

```
<%
Dim nameOfFile, objConn, objRS
Const db = "Provider=SQLOLEDB.1;server=AMINAH\MEEN; Database=helpdesk;
UID=aminah; PWD=helpdesk"
Const adminEmail = admin@fcsit.um.edu.my
Const numRowsFaq = 3
Set objConn = Server.CreateObject("ADODB.Connection")
Set objRS = Server.CreateObject("ADODB.Recordset")
Set nameOfFile = Request.ServerVariables("script_name")
%>
```

Figure 6-4 : Database connection

6.4.3 Including Files

Server-side include directives give a way to insert the content of another file into a file before the Web server processes it. ASP implements only the **#include** directive of this mechanism. To insert a file into an .asp file, the following syntax is used:

<!-- #include file ="*filename*" -->

The **file** keywords indicate the type of path that are using to include the file, and *filename* is the path and file name of the file that wanted to include. Included files do not require a special file name extension; however, it is considered good programming practice to give included files an .inc extension to distinguish them from other types of files.

In Help Desk FCSIT, the syntax that have been used to include database connection to all files is :

<!--#include file="../config.inc"-->

By using this mechanism, a shorter coding of .asp file can be produced. Meanwhile, programmers also do not have to do duplicate work, for written the same function, which is shared with several .asp files. The use of this mechanism is important when a large amount of ASP pages has been created.

6.4.4 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) are a collection of formatting rules which control the appearance of content in a web page. With CSS styles, the developer will have great flexibility and control of the exact page appearance, from precise positioning of layout to specific fonts and styles.

CSS styles let the developer control many properties that cannot be controlled using HTML alone. By using CSS styles and setting font sizes in pixels, a more consistent treatment of page layout and appearance in multiple browsers can be ensure. In

addition to text formatting, the format and positioning of a block-level elements in a web page can be control. For example, the developer can set margins, borders, float text around other text, and so on. Below in Figure 6-5 is one of CSS that has been used in Help Desk FCSIT.

```
A {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 10pt;
        color: #FFFFFF;
        text-decoration: none;
}
a:link {
        color: #FFFFFF;
}
a:hover {
        color: #FFCC00;
        text-decoration: underline;
}
body {
        margin-left: 15px;
        margin-top: 0px;
        margin-right: 0px;
        margin-bottom: 15px;
        background-attachment: fixed;
        background-color: #404d68;
        background-image: url(images/bg.gif);

        scrollbar-face-color: #545E6F;
        scrollbar-highlight-color: #999999;
        scrollbar-shadow-color: #333333;
        scrollbar-3dlight-color: #545E6F;
        scrollbar-arrow-color: #FFFFFF;
        scrollbar-track-color: #404D68;
        scrollbar-darkshadow-color: #999999;
}
.textTitle {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 11pt;
        color: #FFCC00;
        font-weight: bold;
}
.textNormal {
        font-family: Verdana, Arial, Helvetica, sans-serif;
        font-size: 10pt;
        color: #FFFFFF;
}
```

Figure 6-5 : Example of CSS file

A CSS style rule consists of two parts—the selector and the declaration. The selector is the name of the style and the declaration defines what the style elements are. The declaration consists of two parts, the property (such as `font-family`), and value (such as `Helvetica`). The term cascading refers to the ability to apply multiple style sheets to the same web page. For example, one style sheet can be created to apply color and another to apply margins, and apply them both to the same page to create the wanted design.

A major advantage of CSS styles is that they provide easy update capability; when update a CSS style, the formatting of all the documents that use that style are automatically updated to the new style.

6.4.5 Server Variables

The ServerVariables collection retrieves the values of predetermined environment variables. The Syntax is Request.ServerVariables (*server environment variable*). Parameter *server environment variable* specifies the name of the server environment variable to retrieve. In this Help Desk FCSIT project, example of Server Variables that has been used is REMOTE_ADDR. The function of this parameter is to get the IP address of the remote host making the request. Below in Figure 6-6 is the code for request IP address.

```
If objRS.EOF Then
        objRS.close
        objRS.Open "log", objConn, adOpenKeyset , adLockOptimistic
        objRS.AddNew
        objRS("user_id") = □login_Id
        objRS("user_status") = Ucase(status)
        objRS("ip_address") = Request.ServerVariables("REMOTE_ADDR")
        objRS("start_date") = Date()
        objRS("start_time") = Time()
        objRS.Update
        Session("ID") = objRS("id")
        objRS.Close
    Else
        Session("ID") = objRS("id")
        objRS("ip_address") = Request.ServerVariables("REMOTE_ADDR")
        objRS("start_date") = Date()
        objRS("start_time") = Time()
        objRS.Update
        objRS.Close
    End If
```

Figure 6-6 : The use of REMOTE_ADDR

6.4.6 Code layout

It is relatively simple to make a code artifact easy to read [17]. Indentation is perhaps the most important technique for increasing readability. Indentation also had been used to assist in understanding the code. Indentation also shows which statements belong in a given block.

Another useful aid is blank lines. Methods should be separated by blank lines; in addition, it often is helpful to break up large blocks of code with blank lines. The extra 'white space' makes the code easier to read and, hence, comprehend.

6.4.7 Use of Consistent and Meaningful Variable Names

The term meaningful variable names means "meaningful from the viewpoint of future maintenance programmers." In addition to the use of meaningful variable

names, it is equally essential that consistent variable names be chosen. Consistency in ordering of the components of variable names is also important. The example of variable names used in Help Desk FCSIT as below in Figure 6-7:

```
<%
Dim strSQL, userName, userStatus

If Session("HelpDesk") = False OR IsNull(Session("HelpDesk")) = True Then
%>
        <script>
                window.top.location = "../../login/logout.asp"
        </script>
<%
End If

If Session("ID") <> "" Then

        strSQL = "SELECT id, user_id, user_status, online_status FROM log WHERE id = '"
        & Session("ID") & "'  AND online_status = 'ONLINE'"

        objConn.Open db
        objRS.Open strSQL, objConn , 3

        If NOT objRS.EOF Then

                userStatus = objRS("user_status")

        End If

        If userStatus <> "ADMINISTRATOR" Then
        objRS.Close
%>
                <script>
                        window.top.location = "../../login/logout.asp"
                </script>
<%
        Else

                userName = "ADMINISTRATOR"

        End If

        objConn.Close

Else
%>
        <script>
                window.top.location = "../../login/logout.asp"
        </script>
<%
End If
%>
```

Figure 6-7 : Variable Names used in check_admin.inc

6.5 Summary

The implementation assures that the system being developed is operational and then allowing the users to take over its operation for use. After the detail explanation of the implementation phase, the next chapter will discuss about the testing phase. This is also a very important stage whereby testing is essential to assure quality of the system.

CHAPTER 7 :
SYSTEM TESTING

CHAPTER 7 : SYSTEM TESTING

7.1 Introduction

The main function of testing is to establish the presence of defects in a program and to judge whether the program is usable in real application. Nevertheless, testing can only demonstrate the presence of errors. It cannot show that there is no error in the program. Therefore, a more suitable approach must be chosen to reduce the possibility of errors in a program.

Bottom-up approach is adopted in system testing for Help Desk FCSIT. Each module at the lowest level of the system hierarchy is tested individually. Then, all the tested modules would be related to the next module testing. This approach is repeated until all the modules are tested successfully.

7.2 Testing Process

In general, the testing process of Help Desk FCSIT can be shown in the following Figure 7-1. All the details will be further explained in sub-sections.

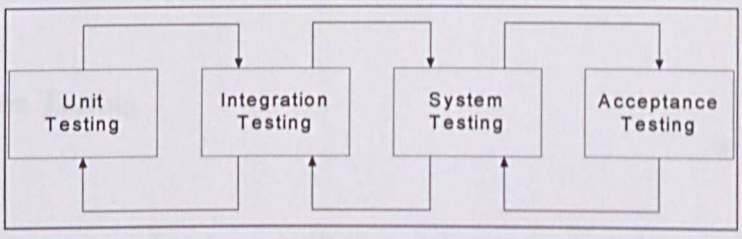


Figure 7-1 : Testing Process

7.2.1 Unit Testing

Starting from the bottom the first test level is "Unit Testing", sometimes called Component Testing. It involves checking that each feature specified in the functional has been implemented in the unit.

The problem with a component is that it performs only a small part of the functionality of a system, and it relies on co-operating with other parts of the system, which may not have been built yet. To overcome this, the developer either builds, or uses special software to trick the component into believing it is working in a fully functional system.

7.2.2 Integration Testing

Once individual program components have been tested, they must be integrated to create a partial or complete system. This integration process involves building the system and testing resultant system for problems hat arise from component interactions. Integration tests should be developed from the system specification and integration testing should begin as soon as usable versions of some of the system components are available [22].

7.2.3 System Testing

Once the entire system has been built then it has to be tested against the "System Specification" to check if it delivers the features required.

In essence System Testing is not about checking the individual parts of the design, but about checking the system as a whole. In effect it is one giant component.

System testing can involve a number of specialist types of test to see if all the functional and non-functional requirements have been met. In addition to functional requirements these may include the following types of testing for the non-functional requirements:

i. Performance - Are the performance criteria met?

ii. Volume - Can large volumes of information be handled?

iii. Stress - Can peak volumes of information be handled?

iv. Documentation - Is the documentation usable for the system?

v. Robustness - Does the system remain stable under adverse circumstances?

7.2.4 Acceptance Testing

Acceptance Testing checks the system against the "Requirements". It is similar to systems testing in that the whole system is checked but the important difference is the change in focus:

- Systems Testing checks that the system that was specified has been delivered.
- Acceptance Testing checks that the system delivers what was requested.

The target user should always do acceptance testing. The target user knows what is required from the system to achieve value in the service operation as well as help desk management and is the only person qualified to make that judgment.

7.3 Summary

The entire process of testing is the key to recognize flaws that exist in a system. Testing is one of the important steps in developing a system. Precision and accuracy of output data is considered during this process. Unit, integration and system testing has been carried out for the Help Desk FCSIT system. These testing approaches lead to delivering a quality system to users. The objective of a system will only achieve after all the thorough testing done by different user with different aspects.

CHAPTER 8 :

SYSTEM EVALUATION

CHAPTER 8 : SYSTEM EVALUATION

8.1 Introduction

Evaluation is the ultimate phase of developing a system and an important phase before delivery the system to the end users. Evaluation was related to user environment, attitudes, information priorities and several other concerns that are to be considered carefully before effectiveness can be concluded. At all phases of the system approaches, evaluation is a process that occurs continuously, drawing on a variety of sources and information.

8.2 Problem Encountered and Solutions

There have been many problems encountered throughout the development of the proposed system. Highlights the problems faced and solutions taken:

i) Understanding the error message

Although thee are many error messages from the Internet Information Server 5.1 that are meant to show where the coding went wrong, they are often unclear and confusing. Sometimes it was indeed annoying o be informed that there is an error occurred somewhere else instead. To correct the wrong coding, a lot of time was spent to look for he solution of the error.

However, the problem was solved by writing out the program in a more careful way to avoid unnecessary errors. Back-ups for ASP pages were created when trying to put new coding that might cause errors.

ii)Module Integration

Since the system contained three major modules, most of the problem faced during integration of all modules.

Some system design needs to be change to integrate the module based on logic thinking and design.

8.3 System Strength

The strengths of Help Desk FCSIT are as follow:

i) Simple and easy to use interface

The interface of the proposed system is both simple and easy to use, where it relies heavily on browsing through the bottom menu. The interactive button created using Macromedia Fireworks MX 2004 are appealing, which give a pleasant effect every time the users point on the buttons, if compare to the normal dull text hyperlinks.

All the forms are ensured to design in user-friendly mode, where radio button and combo list are used to minimize the users' actions while performing certain tasks.

ii) Authorization and Authentication

A custom password authentication system is created to prevent unauthorized users from accessing the page if they do not have any permission to view the data. The users will be redirected to the login page if they attempt to enter the system without permission. This is important to assure security features are included in the system to give the users more confidence on the system.

iii) Significant validation on input data

Check for the validation of every data input in the field and prompt the user of invalid data being input and ask for valid data. Data field that disallow data to duplicates will also prompt the user about the error.

iv) Expandable for Future Use

The system is designed in such a way that it can support additional changes in the future. Administrator can just add the new committee as well student and staff.

8.4 Weakness and Future Enhancement

Like many other system, despite its strengths, Help Desk FCSIT also has several limitations. Thus, no doubt the Help Desk FCSIT is a big improvement from manual system; it is undeniable that the proposed system has the potential to be future enhanced. These constraints can be reduced and addressed in the future development and system enhancement. Listed below are some of these constraints as well as possible enhancement to be incorporated into the system.

i) Browser Limitation

The Internet browser for this Help Desk FCSIT is Microsoft Internet Explorer. All development products that were used are Microsoft products including Microsoft Internet Explorer. Therefore the best view for the system is with Internet Explorer 6.0 as properties in one browser might not recognize in other such as Netscape Navigator.

ii) No E-mail System

Currently, there is no e-mail function offered in the Help Desk FCSIT. But this system provides a printer friendly version to the users.

An e-mail system can be added in the system so that the user will be notified also by e-mail besides accessing to the system only. This will ease the job. Therefore, the e-mail system should be developed in the future.

8.5 Knowledge and Experience Gained

Throughout the development period of this system, a lot of exciting and valuable experience was gained. There has been improvement in searching information and solving problems.

The benefit that gained throughout his project is the chance to understand the concept of system development process. There was a golden opportunity to learn additional programming language, which are not familiar before such as JavaScript and VBScript. Moreover, various development tools such as Macromedia Dreamweaver,

Fireworks and Swift 3D had been learnt in order to design the interface of the system. On the other hand, knowledge of how to configure and manage a database such as Microsoft SQL Server was acquired too.

Skills in time management were also improvised when the system was compulsory to finish before deadline. The way to handle a project under time constraints was learnt. Indirectly, the experience to plan a system and solve a problem had been enriched.

Lastly, there has been a progress in communicating skills when the survey session was carried out to obtain information of the system. The methods of produce a good questionnaire and later analyzing the result of the question, as well as writing a good report were gained.

8.6 Conclusion

Overall, a system which contains the entire proposed module is successfully developed. All the modules and sub-modules have been successfully integrated and have fulfilled both the functional and non-functional requirement. Also, the objectives of the Help Desk FCSIT stated in the proposal has successfully been achieved and obtained.

The system has a high potential to be furthered enhanced and to be made more effective and efficient in the future.

This project has been excellent practical testing to see the capabilities of handling and developing a project. Knowledge and experience gained will be of value and technical skills such as database maintenance, web based programming and design technique can be applied to the real working environment.

It is hoped that this Help Desk FCSIT will be useful to the target users to simplify and ease management of complaint, suggestion and feedback.

REFERENCES

REFERENCES

[1] Mohd Saud Ayutollah Abdul Manan (2001). *Sistem Aduan Kerosakan di FSKTM Berasaskan Web.* Laporan Teknikal. Universiti Malaya.

[2] Tan Pei Yee(2003). *FSCIT Online Complaint & Suggestion.*Laporan Teknikal. Universiti Malaya.

[3] Anusha Devi Rajaratnam. *Student Online Feedback System.* Laporan Teknikal. Universiti Malaya.

[4] http://www.help-desk-world.com/help-desk.htm Date access: 21 July 2004

[5] Ragumaran s/o Govindaraju(2003).*FSKTM Helpdesk Website.* Laporan Teknikal. Universiti Malaya

[6] http://www.wisc.edu/writing/Handbook/ReviewofLiterature.html

[7] http://sbc.webopedia.com/TERM/H/help_desk.html

[8] http://www.helpdesksurvival.com/helpdesk.html

[9] http://www.sei.cmu.edu/str/descriptions/clientserver_body.html

[10] http://www.webopedia.com/TERM/C/client_server_architecture.html\

[11] http://www.computerencyclopedia.com/webexamples.htm

[12] http://www.amazon.com/exec/obidos/tg/feature/-/213782/104-4645935-3172758

[13] http://www.oasisinfoway.com/features_windows.htm

[14] http://www.microsoft.com/windowsxp/evaluation/features/iis.mspx

[15] http://www.m-w.com/cgi-bin/dictionary?book=Dictionary&va=complaint&x=8&y=12

[16] http://www.iqx.com/methodologies.htm

[17] Schach, S. R. 2005, Object-oriented & Classical Software Engineering, 6th Edition, McGraw Hill.

[18] http://courses.cs.vt.edu/~csonline/SE/Lessons/Waterfall/

[19] http://www.hennymarske.homestead.com/

[20] Whitten J. L, System Analysis and Design Methods, 5th Edition, McGraw Hill

[21] Mandel, T. (1997) The Elements of User Interface Design, Wiley.

[22] Sommerwille, I. (1995). Software Engineering. 5th edition. Addison-Wesley Ltd.

[23] http://www.computerhope.com/winxp.htm

[24] Connoly T, Begg C, Database System : A Practical Approach to Design, Implementation and Management, 3rd Edition,Addison Wisley

[25] Sellapan P. (1999). Access 2000 Through Examples.Shah Alam: Times, Federal Publications Sdn. Bhd.

[26] http://www.leveltendesign.com/kb_terms/web_server.html

[27] Schneider Gary P. & Perry James T. (2000). Electronic Commerce. Cambridge: Course Technology.

[28] http://searchsecurity.techtarget.com/sDefinition/0,,sid14_gci211576,00.html

[29] http://msdn.microsoft.com/scripting/default.htm?/scripting/vbscript

[30] http://searchwin2000.techtarget.com/sDefinition/0,,sid1_gci213279,00.html

[31] http://www.intranetjournal.com/faqs/jsfaq/gen1.html

[32] http://msdn.microsoft.com/scripting/default.htm?/scripting/jscript

[33] http://developer.netscape.com/docs/manuals/communicator/jsguide4/index.htm

[34] http://searchdatabase.techtarget.com/sDefinition/0,,sid13_gci211812,00.html

[35] http://www.webopedia.com/TERM/a/authoring_tool.html

APPENDIX A

Serial No: _____

Section A : Personal Details

Choose only ONE answer

1. Gender ☐ Male ☐ Female

2. Status ☐ Staff ☐ Student

3. If Staff,

☐ Support ☐ Academic ☐ Administration

4. If Student, (If Staff, proceed to Section B)

Level ☐ Beginner ☐ Intermediate ☐ Final

Current Level ☐ Sarjana Muda ☐ Sarjana

☐ PhD ☐ Others :_____(please state)

5. Department _____

(please state)

Section B: Helpdesk FSCIT

1. Do you aware of the FSCIT suggestion box?

☐ Yes ☐ No

2. Have you field a complaint or give suggestion before regarding the FSCIT?

☐ Yes ☐ No

3. If No, please state your reason. (You can choose more than one answer)

☐ No complaint/suggestion to make

☐ Manual procedure

☐ Wasting time

☐ Others:_____

(please state)

4. Do you know the procedure of filing a complaint / suggestion?

☐ Yes ☐ No

5. What do you think of the current system of complaint / suggestion in FSCIT?

_____

_____

_____

6. If there is an online system, would you complaint?

☐ Yes ☐ No

7. Any recommended features to include if there is an online complaint / suggestion in FSCIT?

_____

_____

_____

_____

_____

_____

Section C: Assessment on current complaint/suggestion system (For those who had complaint / suggestion before)

1. Why did you make complaint/suggestion?
   - ☐ Having Damages
   - ☐ To increase quality of services
   - ☐ Having Difficulty
   - ☐ Others:_____ (please state)

2. How much Complaint / Suggestion per semester?
   - ☐ 1 – 2
   - ☐ 3 – 4
   - ☐ 5 – 7
   - ☐ > 7

3. Your complaint / suggestion are about? (You can choose more than ONE)
   - ☐ Program Ijazah Sarjana Muda
   - ☐ Program Ijazah Sarjana
   - ☐ Research and Development
   - ☐ Infrastructure / Facilities (eg: labs, air-condition etc)
   - ☐ Others: _____ (please state)

4. Do you satisfied with the action taken by the FSCIT about your complaint/suggestion?
   - ☐ Yes
   - ☐ No

5. How long the respond time against your complaint / suggestion?
   - ☐ <1 day
   - ☐ 2 days
   - ☐ 3 days
   - ☐ > 3 days

6. Would you prefer online system?
   - ☐ Yes
   - ☐ No

##########

Thank you for your cooperation to fill out this questionnaire.

APPENDIX B

UNIVERSITI MALAYA

UNIT PENGURUSAN PENJAMINAN KUALITI

PENGURUSAN ADUAN / CADANGAN / MAKLUM BALAS DARIPADA STAF / PELAJAR

NO. DOKUMEN	UM-PT00-PK08	TARIKH KUATKUASA	27 / 02 / 2004
NO. SEMAKAN	4	JUMLAH HALAMAN	7

Tugas	Tanggungjawab	Tandatangan	Tarikh
Disedia	QAMU		12 / 02 / 2004
Disemak	Wakil Pengurusan		27 / 02 / 2004
Dilulus	Naib Canselor		27 / 02 / 2004

Nombor Salinan	

Dokumen ini dokumen terkawal.
Tidak dibenarkan mengeluar ulang tanpa kelulusan Universiti Malaya.

1.0 TUJUAN

Dokumen ini menerangkan cara mengurus aduan / cadangan / maklum balas daripada staf / pelajar.

2.0 SKOP

Dokumen ini merangkumi urusan menerima dan merekodkan aduan / cadangan / maklum balas, mengambil dan merekodkan tindakan pembetulan dan mengambil tindakan susulan. Dokumen ini akan diterima pakai oleh PTj / Jabatan / Bahagian / Unit di Universiti Malaya.

3.0 TANGGUNGJAWAB

Naib Canselor dan Ketua PTj bertanggungjawab menentukan prosedur ini diikuti dalam pengurusan aduan / cadangan / maklum balas daripada staf / pelajar.

Sesiapa yang terlibat dalam pengurusan aduan / cadangan / maklum balas daripada staf / pelajar perlu mematuhi prosedur ini.

4.0 DOKUMEN RUJUKAN

	Nama Dokumen Rujukan	Nombor Rujukan Dokumen
4.1	PK Penyelenggaraan Aset	UM-PT04-PK02

5.0 DAFTAR KATA

5.1 Definisi

(a) Aduan merujuk laporan / catatan tidak puas hati berkenaan sesuatu perkhidmatan / produk.

(b) Aduan SERIUS merujuk aduan yang boleh mengakibatkan tindakan disiplin atau gangguan keselamatan.

5.2 Singkatan

JPPHB	Jabatan Pembangunan dan Penyelenggaraan Harta Benda
PGKS	Pengarah Keselamatan
PP	Penolong Pendaftar
PTj	Pusat Tanggungjawab

TD	Timbalan Dekan
TP	Timbalan Pengarah
QAMU	Unit Pengurusan Penjaminan Kualiti
PTPO	Pegawai Tadbir (Perkeranian / Operasi)

6.0 PROSES KERJA

Tindakan	Tanggungjawab
Penerimaan Aduan / Cadangan / Maklum Balas	
6.1 Kutip Borang Aduan / Cadangan / Maklum Balas (UM-PT00-PK02-BR001) daripada peti cadangan PTj secara berkala.	Pegawai Keselamatan / QAMU
6.2 Terima dan sahkan bilangan borang aduan yang di terima.	QAMU
6.3 *Asing aduan berkenaan penyelenggaraan untuk diunjurkan kepada JPPHB.*	*QAMU*
6.4 Beri kod bagi setiap borang aduan, rekod ke dalam pangkalan data dan sahkan bilangan yang dimasukkan ke pangkalan data setara dengan bilangan kutipan.	QAMU
6.5 *Hantar borang aduan kepada PTj berkenaan dan sahkan penerimaan hantaran. Bagi aduan penyelenggaraan yang dihantar ke JPPHB, unjur satu salinan ke PTj yang berkenaan untuk makluman.*	*QAMU*
Pemprosesan Aduan / Cadangan / Maklum Balas	
6.6 Terima Borang Aduan / Cadangan / Maklum Balas (UM-PT00-PK02-BR001) daripada QAMU dan rekod penerimaan. *Ambil salah satu tindakan berikut:* (a) *Bagi aduan baru, terus ke langkah 6.7.* (b) *Jika aduan berkaitan penyelenggaraan, simpan borang sebagai rekod dan pantau tindakan oleh JPPHB.*	Ketua PTj / *Pengurus Kualiti PTj / Ketua Jabatan / Bahagian / Unit*

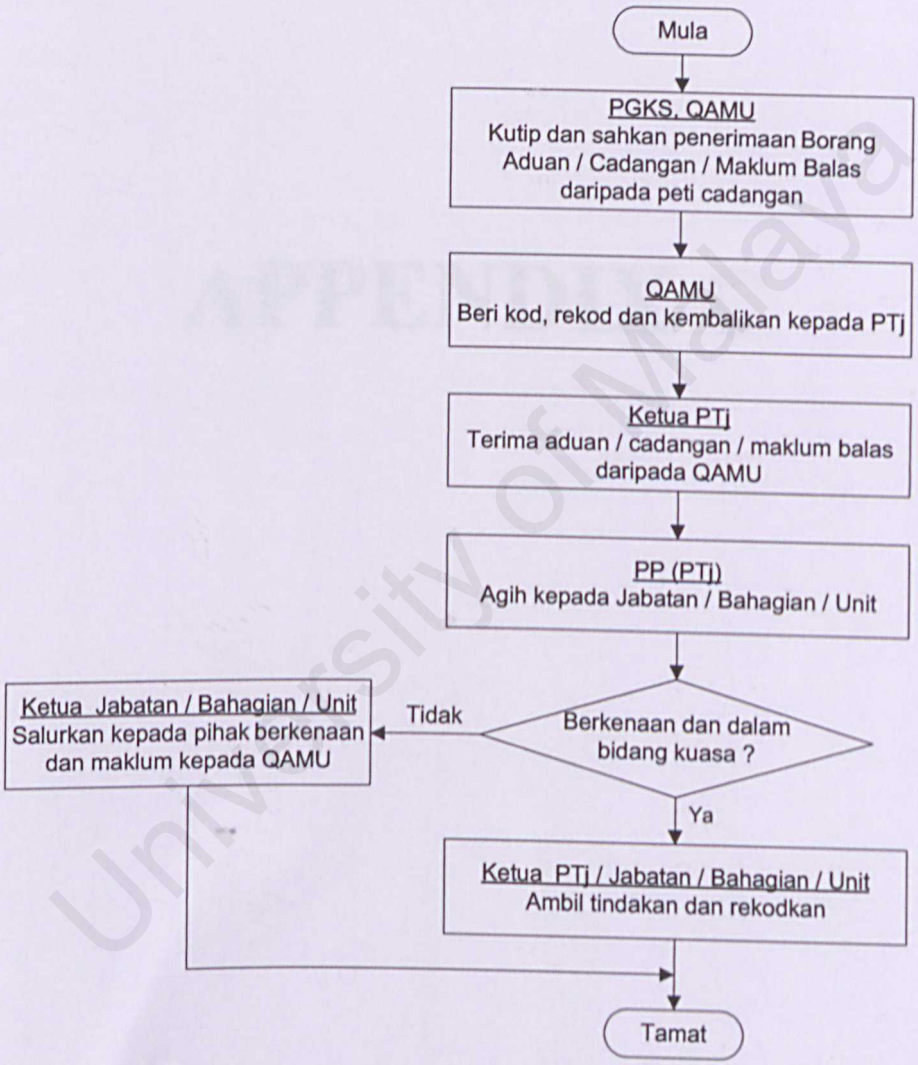
Tindakan	Tanggungjawab
(c) *Bagi aduan yang dikembalikan semula oleh QAMU, beri cadangan tindakan yang baru dan kembalikan salinan borang ke QAMU selewat-lewatnya 7 hari daripada tarikh penerimaan borang daripada QAMU.*	
6.7 Semak semua aduan / cadangan /maklum balas yang diterima.	*Ketua PTj* / Pengurus Kualiti PTj
6.8 Buat siasatan punca aduan.	Ketua PTj / Jabatan / Bahagian / Unit
6.9 Pastikan sama ada aduan / cadangan / maklum balas adalah berkenaan dan dalam bidang kuasa.	Ketua PTj / Jabatan / Bahagian / Unit
6.9.1 Sekiranya tidak, salurkan aduan / cadangan / maklum balas kepada pihak atasan seterusnya untuk tindakan. Rekod tarikh pengunjuran aduan *dan hantar satu salinan borang yang telah dilengkapkan kepada QAMU.*	
6.9.2 Sekiranya ya, ambil salah satu tindakan berikut:	
(a) Bagi aduan / cadangan / maklum balas yang boleh diselesaikan, ambil tindakan yang sewajarnya dan rekodkan. Pantau keberkesanan tindakan yang telah diambil (jika perlu).	
(b) Bagi aduan / cadangan / maklum balas yang tidak dapat diselesaikan, laporkan masalah kepada PTj yang berkenaan.	
6.10 Maklum tindakan pembetulan yang diambil kepada QAMU dengan mengembalikan borang yang telah dilengkapkan *selewat-lewatnya satu (1) bulan dari tarikh penerimaan aduan.*	Ketua PTj
6.11 Teliti tindakan pembetulan yang diambil dan pastikan status aduan.	QAMU

Tindakan	Tanggungjawab
6.11.1 Sekiranya tindakan yang diambil tidak memuaskan, kembalikan aduan kepada PTj/Wakil Pengurusan untuk pemprosesan lanjut.	
6.11.2 Sekiranya tindakan yang diambil memuaskan, kemas kini status aduan dalam pangkalan data(elektronik) dan maklum kepada PTj.	
6.12 *Maklum kepada pengadu (melalui surat / e-mel / notis) tentang tindakan yang telah di ambil dan status aduan berkenaan.*	*Ketua PTj / Pengurus Kualiti PTj / Ketua PTj / Jabatan / Bahagian / Unit*
6.13 *Hantar satu salinan makluman kepada pengadu kepada QAMU.*	*Ketua PTj / Pengurus Kualiti PTj / Ketua PTj / Jabatan / Bahagian / Unit*
Pengurusan Kaji Selidik Maklum Balas Pelanggan	
6.14 Buat kaji selidik maklum balas berkenaan kualiti di Universiti Malaya sama ada dari kalangan staf atau pelajar sebagai tindakan pencegahan sekurang-kurangnya sekali setahun.	QAMU
6.15 Sedia laporan hasil kaji selidik dan serahkan kepada pihak pengurusan universiti.	QAMU

7.0 REKOD KUALITI

Bil.	Rekod	Lokasi	Tanggungjawab	Tempoh simpanan
7.1	Borang Aduan / Cadangan / Maklum Balas (UM-PT00-PK02-BR001)	Pejabat PTj / Jabatan	TD (Pembangunan)	3 tahun (minimum)
7.2	Rekod pengesahan bilangan kutipan	QAMU	Pengurus Dokumen	3 tahun (minimum)
7.3	Rekod pengesahan kemasukan rekod borang aduan ke dalam pangkalan data	QAMU	Pengurus Dokumen	3 tahun (minimum)
7.4	Laporan Kaji Selidik Kualiti di Universiti Malaya	QAMU	Pengurus Dokumen	3 tahun (minimum)

8.0 CARTA ALIR



9.0 LAMPIRAN

Tiada

APPENDIX C

Abstract

The purpose of this user manual is to provide some helpful guideline and usage about this Help Desk FCSIT to the user.

Help Desk FCSIT can be divided into three main modules – Administrator Module, Committee Module and Registered User Module.

Administrator has full access rights to the system. They can add, modify, update and delete records of category. Administrator is responsibility to create and maintain the website of organization. They can view the summary reports through this system too.

Committee has limited access to this system. The assign committee can access their part only. They can not view other committee profile.

Registered User can send complaint, suggestion and feedback. Their profile can be also update through the system.

1.0 Welcome Page

When the hyperlink to the Help Desk FCSIT is clicked by the user, the following interface below (Figure 1) will appear. To continue using the Help Desk FCSIT, click on **Enter** button.

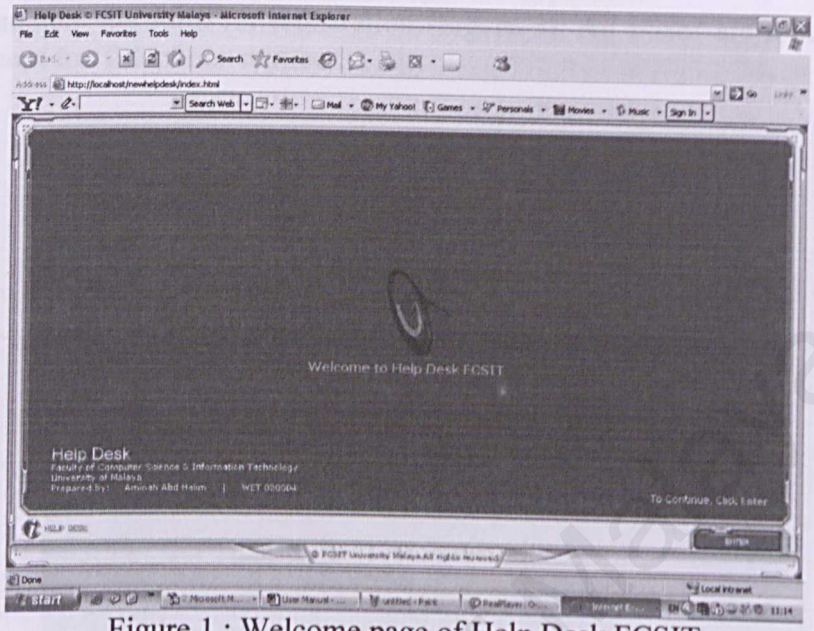


Figure 1 : Welcome page of Help Desk FCSIT

User needs to choose the appropriate category of users by clicking the buttons as shown in Figure 2 to login.

Note:

- **Registered User** (for staff and student), **Committee** (person in charge the committee) and **Administrator** (the administrator of the system).

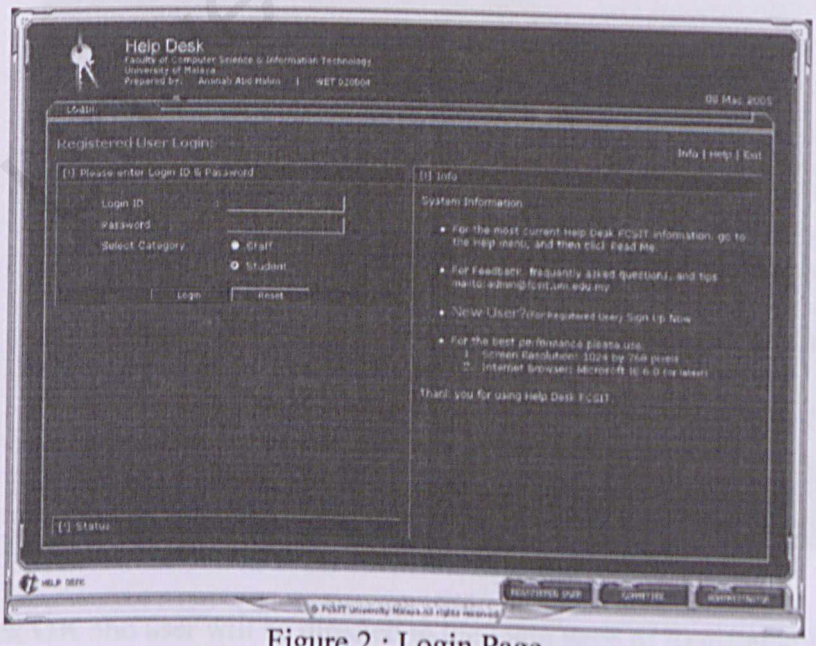


Figure 2 : Login Page

2.0 Registered User

2.1 Registration

To use the system, user (staff and student) must have an account and need to sign up.

1. In the **Info** column, click on **Sign Up Now** hyperlink.
2. Complete the form for registration (Figure 3). All boxes are required to fill except for Telephone number.

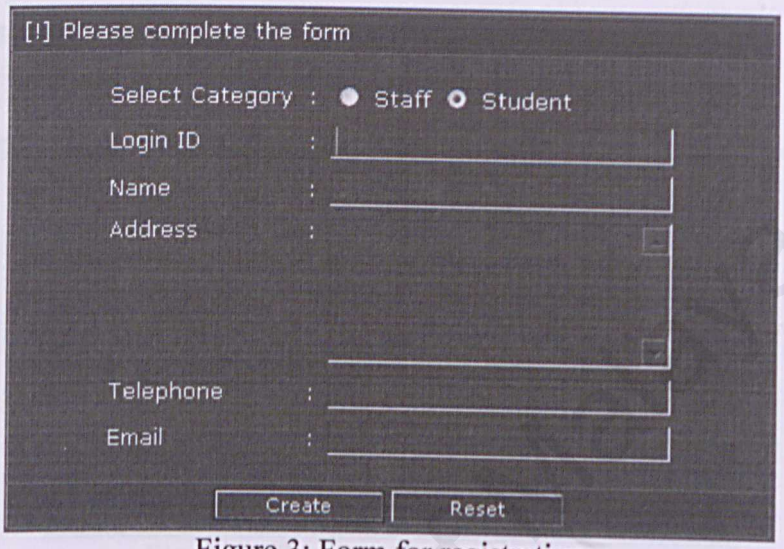


Figure 3: Form for registration

3. Click **Create** to register and **Reset** to clear the form.
4. The account is successfully created (Figure 4). Write down the password for future login.

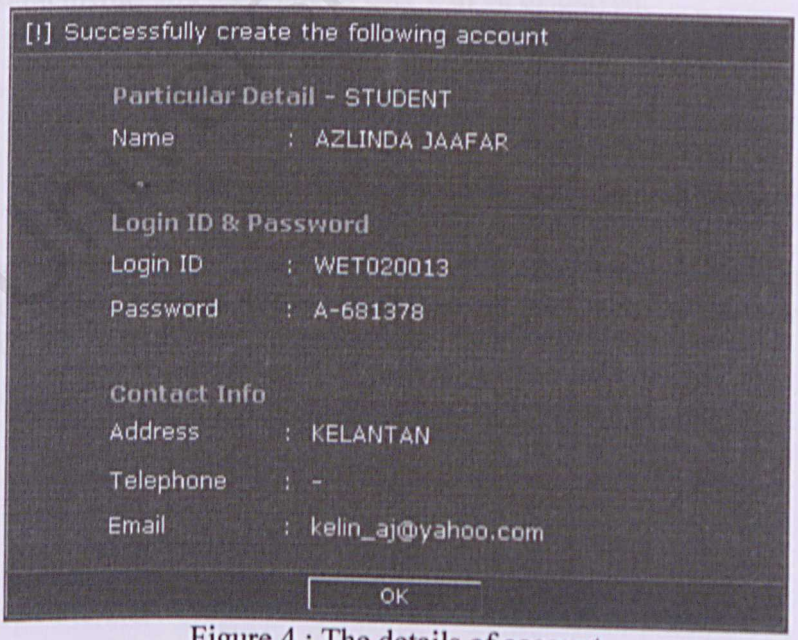


Figure 4 : The details of account

5. Click **OK** and user will be directed to the login page of Registered User.

Notes:

- If user account is existed in database, user will prompt with the message **User account is already in database.**

- **Please check your Login ID** means that user is not authorized to use the system because the login ID does not exist in the database. Please refer to administrator to add your Login ID to the database.

2.2 Registered User Login

In order to use the system, registered user needs to login.

1. In Registered User Login form (Figure 5) , enter Login ID and Password

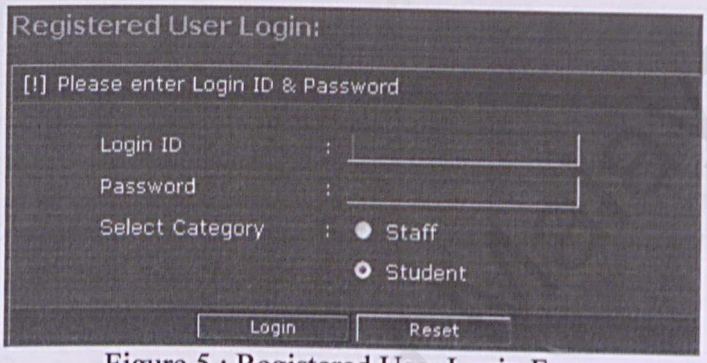


Figure 5 : Registered User Login Form

2. Please select the category (Staff or Student).
3. Click on **Login** button to enter the Help Desk FCSIT or **Reset** to clear the login form.
4. The main menus and the name of user will be displayed.

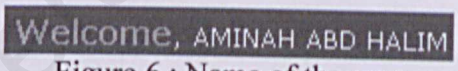


Figure 6 : Name of the user

Note:

Invalid Login ID and Password (Case Sensitive) means that user enter invalid Login ID or password. Case sensitive means that user needs to enter exactly as in registration form. Login ID must in upper case letters.

2.3 View User Profile

1. Click the **Option** button and select **User Account** to View User Profile.
2. In the User Account page, click **View Account** (Figure 7).

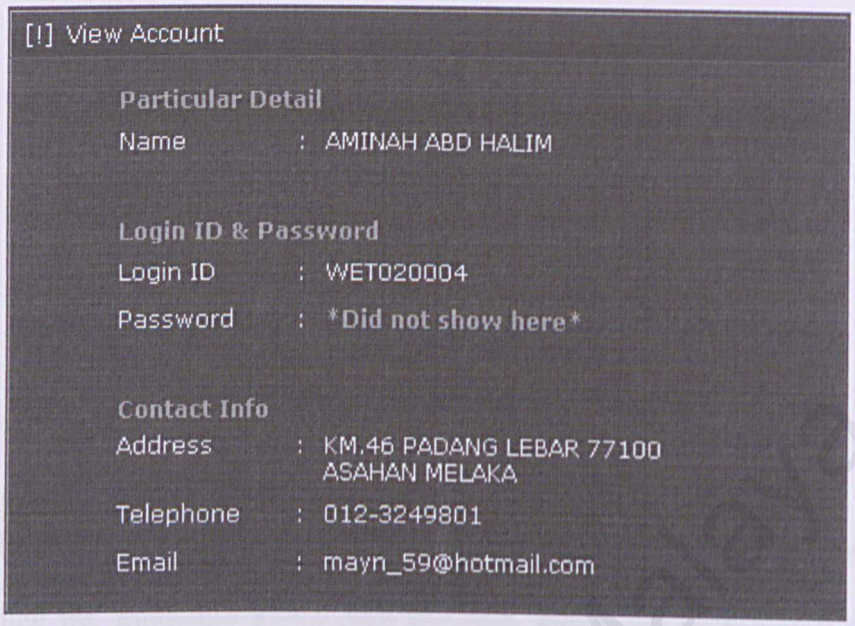


Figure 7 : User Profile details

2.4 Change Contact Info

1. Click the **Option** button and select **User Account** to change contact info.
2. In the User Account page, click **Change Contact Info** (Figure 8).

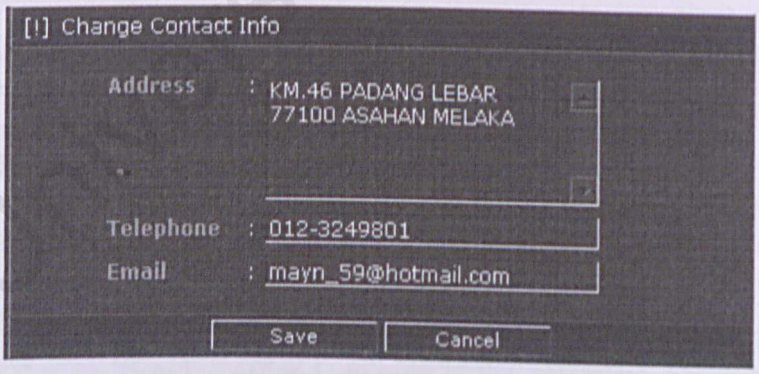


Figure 8: Change Contact Info form

2.5 Change Password

1. Click the **Option** button and select **User Account** to change password.
2. In the User Account page, click **Change Password** (Figure 9).

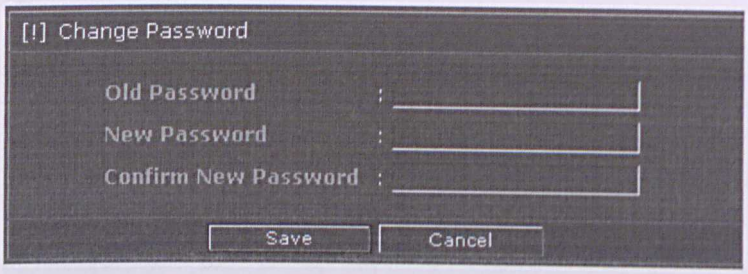


Figure 9: Change Password form

2.6 View FAQ

1. Click on **FAQ** button to view the Frequently Asked Questions.
2. The list of question is in the left box and the answer in the right box. Please select the question to view the answer.
3. Click the following buttons > , >> , << , < to go back and forth on the FAQ section.

Note:

- **Sorry! This section is currently not available** means that the FAQ section in database is empty.

2.5 Print FAQ

1. Click on the **Print FAQ** hyperlink on the top right of the screen. New window will appear.
2. Click on **File** and select **Print**.
3. Select **Print** to proceed and otherwise, click **Cancel** .

2.6 Log Off

1. Click  button to log off from the system.
2. Click **OK** to confirm log off and otherwise click **Cancel.**

2.7 Send Complaint

User can send complaint, suggestion and feedback by using User Complaint form.

1. Click on **Complaint** button.
2. In the User Complaint form (Figure 10), select the **Category** and write down the **Explanation**.

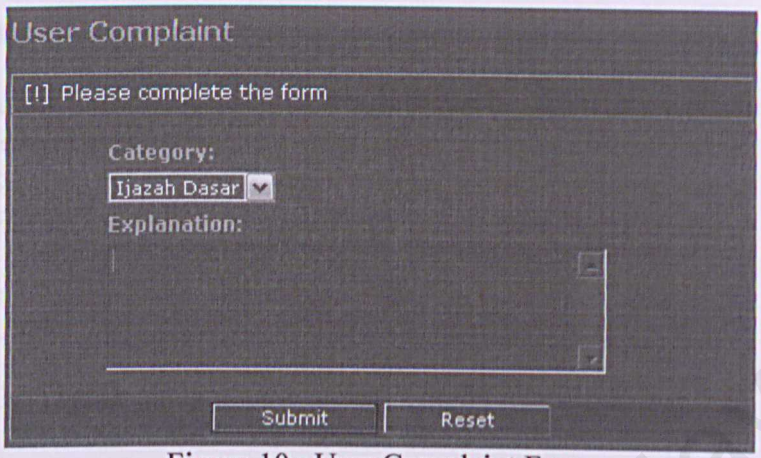


Figure 10 : User Complaint Form

3. Click **Submit** to send the complaint and **Reset** to clear the form.

Note:

- **Please write the Explanation** means that the user submits the blank Explanation.

2.8 User's Inbox

User's Inbox display the Lists of complaints that have been made and the status.

1. Click on **Main Page** button. The Inbox page will be displayed (Figure 11).

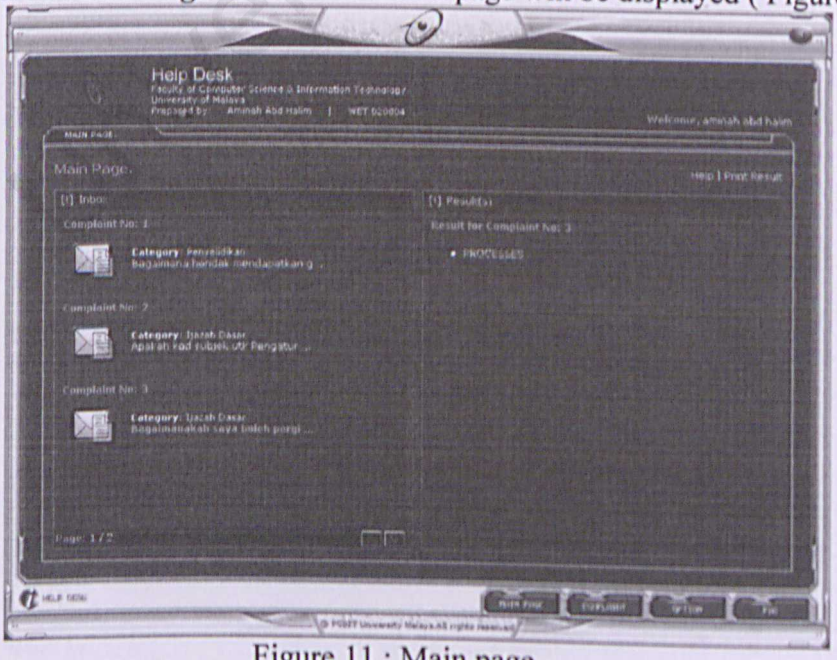


Figure 11 : Main page

2. The previous complaint is in the left and the status of the complaint is in the right.
3. User must click on the previous complaint to see the result.
4. Click the following buttons [>], [>>], [<<], [<] to go back and forth on the Inbox section to view the complaints and their results.

2.9 Print Result

1. Click on the **Print Result** hyperlink on the top right of the screen. New window will appear.
2. Click on **File** and select **Print**.
3. Select **Print** to proceed and otherwise, click **Cancel** .