Management of Teachers' Information System Maharagama

Neranga I Kalasinghe

129154N

Dissertation submitted to the Faculty of Information Technology, University of Moratuwa, Sri Lanka for the partial fulfillment of the requirements of the Degree of Master of Science in Information Technology.

2016

Declaration

I hereby declare that the thesis submitted by me is based on original works carried out by me. Any reference to work done by any other person or institution or any material obtained from other sources have been duly cited and referenced. I further certify that the dissertation has not been submitted in any form for any other degree or diploma at any university or other institution of tertiary education.

....

N. I. Kalasinghe

Date: 2016.12.15

I certify that the above particulars given are true and correct to the best of my knowledge.

Dr. H.T. Chaminda

Supervisor

Date:

Acknowledgements

First, I would like to thank Dr. H.T. Chaminda, Senior Lecturer, Faculty of Information Technology, University of Moratuwa as supervisor for the support, advices, comments given by him to succeed my research. And also like to thank all the lecturers and the staff of the Faculty of Information Technology, University of Moratuwa for their assistance throughout the course.

Secondly, I wish to thank academic staff, non academic staff and students of the National College of Education Maharagama for participating my informal interviews, giving their ideas and feedback, helping to build the system.

Then, I extend my heartiest gratitude to my husband and my little daughters for their unconditional love and encouragement.

Finally I would like to thank everyone who helping to finish my project report.

Abstract

National College Of Education On Entrepreneurship & Management - Maharagama is a one of a National College Of Education(NCOE) in Sri Lanka, which belongs to ministry of education. NCOE conduct special professional training programs for student teachers and in-service training to the teachers.
At present, Call for applications to this professional training program are gazette by government and recruiting students for NCOE by the Ministry of Education. Student those who are completed this special professional teaching training program are able to get "National Diploma in Teaching certificate".

After entering students to the NCOE, NCOE responsible for handling all activities until they pass out.

Collecting student information, course unit registration, recording student attendance, issuing examination admission, conducting internal exam, selecting students for final exam, publishing student results, maintaining course unit information, assigning lectures for courses, calculating student eligibility for final exam and maintaining pass out student information are done by the NCOE.

Above activities were done manually and using computer application package. Numbers of hard copies of files are stored in folders. Some time same set of papers are repeated again and again because there was not flexible relationship with folders. Though MS Excel and MS word are used, file managing was difficult. Also administrative staffs are unable to find specific student information within short time period. Some time there is no guarantee about accuracy of that information as all searching activities were done by manual.

An Information System should be introduced to overcome above problems. The system is accessed by different level users such as students, lectures, President, Vice President (admin), Vice President (academic), Vice President (continuous), office staff in different location of the College. The new information system achieved to meet online student and course unit registration, assigning lectures for courses, giving web portal to check student eligibility for examinations, storing student result, calculating grades, maintaining pass out student information. Also students and other staff members in the college are collaborated with the information system.

Students can access their web portal and do their academic relative activities online without delay. Further the system guarantees to provide accurate information for students.

The information system had been designed by using Object oriented design approach. It had been developed by using PHP, HTML, JavaScript, AJAX, cascading style sheet and MySQL database server technologies.

The system achieved user expectations. Student activities are managed by the system. The system gave management information to administrative staff also. Finally the system removed manual process errors and functioned smoothly.

Table of Contents

Page

List of Figures

List of Tale

List of abbreviations

Abbreviation	**Definition**
NCOE	National College Of Education
MOTIS	Management of Teachers' Information System
PHP	Personal Home Page
NIC	Network Interface Card
LAN	Local Area Network
IS	Information System
UML	Unified Modeling Language
ER	Entity Relationship
OOAD	Object Oriented Analysis and Designing

Chapter 1

Introduction

National College Of Education On Entrepreneurship & Management - Maharagama is a one of a National College Of Education (NCOE) in Sri Lanka which belongs to Ministry of Education. At present nineteenth (19) NCOE established within country. NCOE offers special professional teacher training programs for student teachers and in-service training to the teachers.

Call for applications to this professional teacher training program are gazette by government and recruiting qualified students for NCOE by the Ministry of Education. Then they are attach to the one of NCOE according to subject vice. National College Of Education –Maharagama, conduct professional teacher training courses On Entrepreneurship & Management.

At present students based activities are as follows. Every student should register first in the NCOE. Then they are categorizing according to register course units. NCOE giving opportunity to the register students, to follow the course. NCOE offers "National Diploma in Teaching certificate" after successfully completing within three years.

After first two years exert final exam as internal and external. Before the internal exam student admission is issued from NCOE office. Only eligible general subjects are included on admission. For selecting eligible subjects are according to student attendance, submitting assignments and its marks. Final external exam for professional subjects conduct and admission is issued by Department of Examination. Both examination admissions are issued as a hardcopy. Third year they are attach to the government school for practical teacher training.

National College of Education –Maharagama doing several activities which are related with student activities. One of that is lectures are assigned for course units. Also internal exam result is publishing, attach to the government school for practical training and sending internal exam marks to the national institute of education (NIE).

Final exam result issue by Department of Examination. Pass out student information is managed by the NCOE.

Aim of the Management of Teachers' Information System (MOTIS) is to be created one system which can handle above activities in smooth and user friendly way. Also the system functions accurately. System can handle different privilege level users. System is able to facilitate to access users different locations in NCOE premises such as, president office, vice presidents offices, registrar office, auditorium, computer lab lecture room, common room. Another major task is bulk of paper work related with student activities are slowly replaced by online system.

1 .1 Scope

From student registration to publish result of exam activities done more paper based. Those activities are not like smooth flow, and some activities are isolated. Exam admission issuing, result publishing activities and attach to school for practical training take long time because of manual process. Finding particular student information for administrative purpose is not easy. Different batches of student information are stored in different computers and manual files system.

Management of Teachers' Information System (MOTIS) is introduced to overcome above problems. Students can login to the system and they can do bellow activities. There are changing profiles, course unit registration, checking attendance, receiving examination admissions, downloading and uploading assignments. Also administrative staff can find student information, result, and pass out student information by using the system. Technical officers can login to the system and enter student attendance very simple and fast way rather than entering several Excel sheets.

1.2 Problem specification

In present situation, Student's name, and course are entered into computer. All information is not recorded into those machines. Remain part of information is in hard paper. If some administrative staff member is going to get information about a student, he/she will refer both computer and hard copies. So he/she will face another problem, students records are not available in one central computer. Changing of existing student record is not easy. If a student changes his/her address and new address information is given to the office then office staff will update address of the student, but there is not guarantee to update every file which includes his/her address information. Currently student attendance is recorded into attendant register by manually. Percentage attendance for each student will be calculated later. Student eligibility of a particular subject unit is decided one of methods' using the attendance percentage, which is calculated by manually. So there is a probability to happen mistakes. Top level administrative staff members do not like to spend too much time to find course unit information but they have to find one by one excel or hard copy of files while they are going to find course unit information.

1.3 Aim of the MOTIS

The aim of Management of Teachers' Information System is built a central location for student information and their academic activities. They can access the system in any location of the college. The system allows to student update their information. Also it facilitates to get useful information such as percentage attendance, assignment result, and exam result. Accuracy and robustness system was given to students and reduced paper work in the college also students were practiced to do their academic activities with the system.

1.4 Objectives

Different academic and non academic activities are put in one place. Introducing online system for students and other staff members. Users are categorized different privileges levels. Minimizing delays for exam admission issuing, result and finding student information. Also increasing accuracy about student attendance. Facilitating staff members to do their student related activities efficient way. Reducing paper based activities.

1.5 System requirements

The MOTIS is installed in a server computer. Users can access the MOTIS by using their client computers. Server and client computers should be properly interconnected by using a computer network. Here we will more focus hardware requirements about server and client computers.

Server computer

2 GHz processor (parallel processors)

4 GB RAM

500 GB free hard disk space

LAMP (Linux, Apache, Mysql, PHP) package

More than one Network Interface Cards (NIC)

Client computer

1 GHz processor

2 GB RAM

10 GB free hard disk space

Network Interface Card (NIC)

Windows or Linux Operating system with any web browser

Other requirement

Local Area Network (LAN)

Star network topology [16] is suitable for the MOTIS.

1.6 About next chapters

Chapter 2: Background

This chapter will explain functional and non functional requirements. Based on literature survey, the chapter will discuss existing systems which relate with the MOTIS, and how MOTIS is change from them.

Chapter 3: Analysis and Design

Methodology was used to design the system, techniques was used to gather requirements, different diagrams which explained system in different view ports will be discussed in this chapter.

Chapter 4: User Interfaces

Interfaces which were designed for the MOTIS will be discussed. Also quality factors of interface and some selected interface in the MOTIS will be explained in this chapter.

Chapter 5: Implementation of the MOTIS

Functions and model structure of the system will be discussed. Further the chapter will contain development tools & Technologies and some selected codes.

Chapter 6: Testing MOTIS

Testing process of the system will be discussed in this chapter. Testing process and some selected test cases will be presented.

Chapter 7: Evaluation

Achieved objectives will be discussed. Furthermore difficulties of the MOTIS will be discussed.

Chapter 8: Conclusion

Objectives and activities related with the project will be presented. Also future development and enhancement of the MOTIS will be discussed.

Chapter 2

Background

2.1 Introduction

This chapter will discuss functional and non functional requirements of the system. This was gathered according requirements of academic staff, non academic staff and student. Based on literature survey how existing systems related with the MOTIS and how MOTIS was differing from them is discus.

2.2 Requirements

2.2.1 Functional requirements

Recording student and course unit information.

System will be able to collect and store student personal information. Further those information should be arranged into easy to find, easy to update by administrative staff.

Issuing exam admission.

This function will issue examination admission for each student in the college based on attendance percentage level.

Assigning lectures for course units.

Lectures are assigned to available course units by the MOTIS. Those course units are offered by NCOE.

Course unit registration by student.

Collecting course unit registration information of each students. Registered course units by a particular student will not be confirmed until vice president (academic) will confirm.

Generating attendance sheet and calculating student attendance percentage.

This function generates attendance sheet when technical officer will enter course unit, and date. Then technical officer can mark that online attendance sheet. If there is a wrong marking then function will allow changing that mark.

Selecting students for final exam.

This function will provide form to select eligible student for final exam according to their attendance percentage, submitting assignment and 100% attendance of practical lessons of a particular student.

2.2.2 Non functional requirements

User can access the system any time within the college premises.

System should work 24 hours without break down. So server computer should run under regular and stable power supply. Also other network software and network hardware should work properly. System administrator trained well.

Number of users can access the system simultaneously.

System should be able to handle multiple user login at a time. Any user will not wait to login with the MOTIS during many users are accessing the system.

Respond time of the system should be minimum.

MOTIS should respond user request as far as possible. After requesting some result by user, the result will be set by the system soon. It does not take long time to give the result.

User friendly interfaces.

User interface of the MOTIS should not be complex. It should be simple. Any user can understand tasks are done by each link in his/her first watching.

Interfaces should be content based rather than graphical based.

The system should not contain too much graphic items and animations. It will lead to take long time for loading a web page of the system. Some time user cannot find required information though it is available in a page because of more graphics.

Backup of the system should be taken.

The MOTIS failures should be recovered soon. Backup of web root and database should be taken after some time period. To recover hardware or software failures a backup server should be maintained.

Size of interfaces should be normal (Not too small or too large).

User interfaces should be displayed in normal size. Users will face reading difficulties because of too small interfaces. User should scroll a window vertical and horizontal direction because of too large interfaces.

2.3 Literature review

MOTIS is an information system. The system can be categorized under Information System (IS). Below, going to look similar system to MOTIS.

SchoolTool

User collaboration is a one key part in our project. Also attendance taking parts should be included to MOTIS. SchoolTool [1] is an open source web based student information system which had been designed for developing countries schools. That tool is compatible with Ubuntu Linux operating system. Students, teachers and parents can collaborate by the tool. User groups are managed by the tool. Attendance can be tracked each student.

ClaSS

ClaSS [4] is another student information management system which had been designed for LAMP (Linux, Apache, Mysql and Php) environment. ClaSS has web based student administrative capabilities. Class room activities are covered by the system. Also the system had been implemented for speed access of student records. The system is independent from client computer operating system and web browser software. Web based administration part was used to design our project administrative tasks. The independent property from client and operating system of the ClaSS was applied to our project.

TechLearning

TechLearning [2] is an article which had been published in web. This article was written to describe objective, features and system requirements about student information systems. Student admission, information availability, centralized information service, monitoring student activities, scheduling of examinations and capabilities of student information were discussed by the article. The article was used to study more about centralized information services.

Student services project

Guide line for creating student services online [8] is another web based publication. It describes about a project which was designed for online student service. This project was helped to identify student services for online learners. Also motivation forces for online services were explained. The project scope was declining budgets, growing enrollments on campus and online, increased accountability, student expectation, etc. Motivation forces for online users section was helped improvement of our project.

Student self-services project

We found another student system project which had been developed for University of Melbourne by John Julian [5]. This is a web based interface student self-services. Student, academic and administrative are encouraged to interact by the project. This system had been designed for new generation students. Main goal of the project was enable significant services and facilities for student s and staff. Significant services of online student information system were absorbed to our project.

Web portal system

Web portal system for student information system [3] was found in large number of projects web site. Though it is a java project there were several concepts which could be kept to our MOTIS. The project had been designed for students, faculty and academic faculty. That is a module system it means each task had been done by a module. We studied modular structure of the web portal because our MOTIS is based on modules. Also this project is a solution for paper based traditional file system.

Immigration strategy from paper based system to online system was studied by using this system.

Business information systems

Business information systems are also information system and they have some relations with other information systems which can be helped to build student information systems. Management Information System by Kenneth C. Laudon [12] was referred to get picture about over all view of an information system. Contents of Information technology infra structure chapter of the book describes project planning which was helped to plan our MOTIS. Building and managing systems chapter were also helped to implement MOTIS.

Management Information Systems

Management Information Systems is another book which had been written for business information systems by Effy Oz [6]. The book explains different services which are provided by an information system. We decided which services should be provided by our MOTIS. Data management and data base models technologies were identified for information system. Services of our project were clearly identified by using this book. This book was helped to select suitable database model for our project.

W3Schools

W3Schools is a web developer information website, with tutorials and references relating to HTML, CSS, JavaScript, PHP, SQL, Bootstrap, and jQuery. Some codes are used form this website to developed my system.

Chapter 3

Analyzing and Designing

3.1 Introduction

In this chapter will discuss techniques which use to gather user requirements. Then analyze phase, those requirements were represented by diagrams such as use cases, class diagrams, sequence diagrams, etc. Conceptual model of the system was designed by using those diagrams. Also goals and relation within the conceptual model was discussed. Database was designed for the system. Then that suitable methodology will select [9].

3.2 Requirement gathering

Requirement gathering is a very important task, because entire project development is based on correct requirement gathering. This project several type of requirement gathering technique were used.

Observation in working environment was one of requirement gathering technique which was used for the project. Students, academic and non academic staff behaviors with academic and student activities were observed.

To get more functional understanding documents which were available in office were referenced. There was document available as hard copies. Under hard copies student registration book, student record book, student registration files, examination result books, attendance books, assignment marks book, practical marks book were referenced.

Requirements gathering from observation and document referencing but some requirements, activities and relations were not captured well. So used interviews also to collect requirements. Those interviews were conducted informal way rather than formal because users could communicate without any stress and they were encouraged to tell true and real activates rather than talking about pre defined stories. Administrative staff members, lectures, data entry operators and randomly selected students were interviewed.

3.3 How manual system is functioning

A student is registered by college office. Student information is recorded into papers. Hard copy of course registration form is stored in particular folder. Second year students should attach for school to completing their practical training as a teacher trainee. Every student dually filled application form (hard copy) for final exam is to be handover to the office. Then suitable student for final exam are filtered.

While students are going to lectures their attendance are recorded into registration book. Then those attendances for each course units are entered into books by management assistant. Those attendances are considered while issuing examination admission for student. Typically, student should reach at least 80% attendance for a particular course unit and submitting assignment and 100% participation practical lessons to be eligible to write the final exam. Then those records are entered into books and forms are sending to the department of examination.

Internal examination admission is designed by Word application software. After the internal exam student grades for each course units are entered into marks book and fill special mark sheet sent to the national institute of education. Pass out student's data is maintained by the college office, NIE also Education ministry. Other than student centric activities there is another several activity is happened. It is, lectures are assigned for course units. The task is responsible of vice president (academic) and course heads they develop those by manually.

3.4 Problems with the manual system

Problems of manual system are given below. All activities were based on hard papers and isolated computer application packages.

- Activities do not relate with each other.
- There is not a consistency of the manual system.
- Student information finding is difficult.
- Same data set is repeated again and again.
- Take long time to complete task (issuing exam admission, publishing result).
- There is a high probability to feed wrong records (entering attendance).
- Manual system cannot be controlled centralized.

3.5 Answers

Above problems was solved using one information system. Answers of above problems are given separately.

- Each activity is gathered into central system.
- Building an online system will establish consistency.
- All student information will be converted into electronic format.
- Data repeating can be controlled by critically analyzing data base relations.
- Online system with correct hardware devices will reduce time to complete a task.
- Functional requirements will be designed without errors to improve accuracy of the system.
- Designing administrative level users and groups the system will be controlled centralized.

3.6 System Users

Administrative users

President of NCOE Maharagama, Vice President (admin), Vice President (academic) Vice President (continues) are administrative users. They can view student information, view / edit course, approve / disapprove course unit registration, assign / remove lectures for course units, check student attendance, change requirements for exam eligibility, check / change student result, and find information about pass out students.

Lectures

Lectures, assistant lectures are in this group. They can find their course units, student attendance for their course units and find exam result for their course units.

Management assistant

NCOE has more than 10 numbers of management assistant. Data entering process is handled by them. They can generate attendance sheet, enter / edit student attendance.

Students

Students can do different activities with the system. Student can view / change profile, select course units, check attendance percentage, apply for final exam, view exam result and .

3.7 Design of the system

This section will discuss methodology which was used to develop the system, architecture design, component design and data design.

3.7.1 System development methodology

Natures of requirements for the project are stable and not changing frequently. Also requirements are clear there is not any unclear requirement. So best methodology is incremental prototype with nature. According to incremental proto typing, final product is built as separate prototypes. Each prototype is redesigned until user requirements are met. After that those separate prototypes are merged to complete the entire system.

3.7.2 Design Approach

The system was designed by using Object Oriented Analysis and Designing (OOAD) approach [10]. That approach was used to design conceptual model of the system. Encapsulation property of the approach was used to design modularized system. Abstraction and reusability properties were helped to do the task properly.

Collected user requirements were converted into business or system models by using OOAD tool. Used Argo UML [11] for our OOAD tool. Functional requirements as well as non functional requirements were modeled. Different views of the system were designed. There were Use Case diagrams, Class diagram and Activity diagram.

System Modularization

Entire system was divided into several main modules based on their domain. OOAD concepts were used to do above task. Each module was divided into sub modules for developing purposes. Other than that user groups were designed. Each user group had different privileges. User groups are Administrators, Assistant Registrars, Department Heads, Lecturers, Technical Officers, Office Users and student.

Information Management System main modules

01 Add / Edit / View teachers information

02 Add / Edit Course unit information

03 Course unit registration / cancel registration

04 Assign / View / Remove lectures for course units

05 Enter / Edit / View student attendance

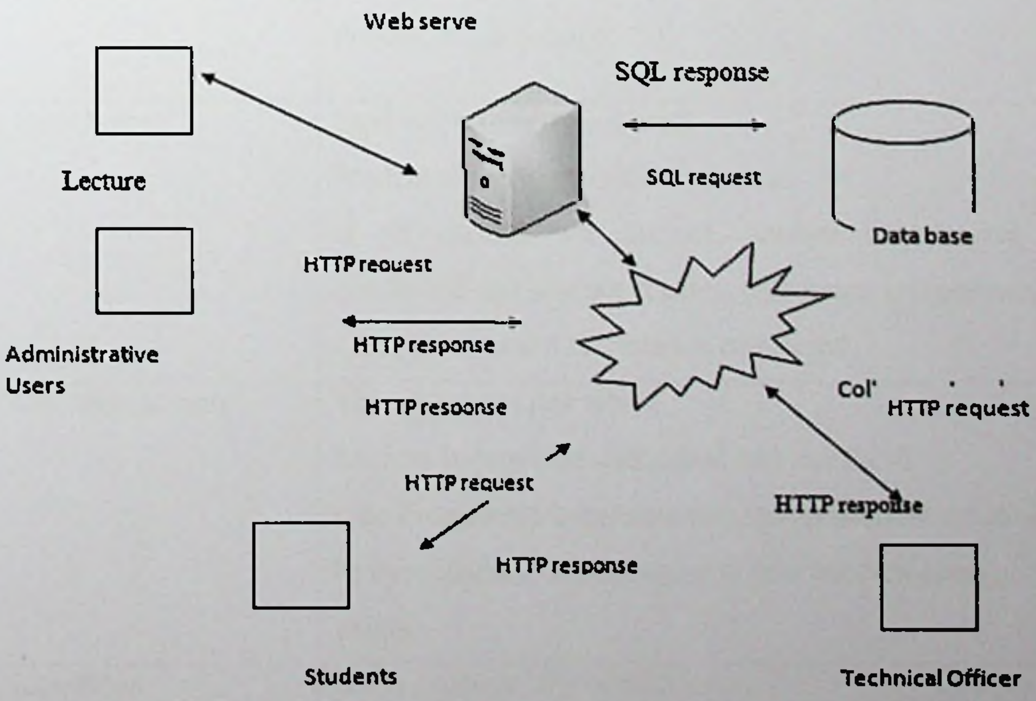
06 Show exam eligibility

07 Apply / Select for final exam

08 Enter / Edit / View exam result

3.7.3 Architecture design

The system is installed central location. User can access the system far away from the central location. Also the system can be accessed multiple users. Client server model is suitable architecture according to above requirements. Below diagram shows our system according to client server architecture.



System Architecture

Figure 1 System Architecture

3.7.4 Component design

Gathered requirements will be represented by using Unified Modeling Language (UML). Different diagrams are used to describe different perspectives in high level system definition. Use case diagram, class diagram and activity diagram will be discussed.

3.7.4.1 Use case diagram

System users and their activities are represented. It was helped to identify goals of each user. A user may be a physical user or a computer system. This section will describe more details about main use cases

student account Use case narrative	
Name	Create student account
Pre-condition	Student has not registered with the NCOE.
Use case description	Creating an account for student by storing student information.
Actors	Student, vice President(admin)
Use case initialization	This use case starts when fresher student will request to resister in the college.
Use case dialog	Student records are entered. Student records are double checked. If all records are correct, student information is confirmed and student is given user name and password, otherwise student records are re entered.
Use case termination	This use case ends when. Student is provided user name and password. vice President(admin) cancels student account creation. System displays a message that user account already exists.
Post-condition	Upon success full completion: User account is updated.

	Upon cancel registration, system returns to initial state. Upon user already exist message, system gives new interface to provide new user name and password.

Table 1- Use case narrative for student registration

Use case narrative Register for course unit	
Name	Register for course unit
Assumption	User has valid user name and password.
Pre-condition	Time period should be beginning of a first year.
Use case description	Student selects and requests to register valid course unit. VP (academic) confirms or not that course unit registration.
Actors	VP (academic) , student
Use case initialization	This use case is started by student.
Use case dialog	Student logins to the system. If login is correct proceed otherwise student is asked to re enter login details. Student selects appropriate course units. Course unit selection is confirmed by student. If student confirms without selecting any course unit, system will display error message and terminate process. VP (admin) logins to system. If login is correct proceed otherwise VP (admin) is asked to re enter login details. VP (admin) opens student course unit registration interface. Course unit registration is confirmed or not.
Use case termination	Course unit registration is confirmed. Student does not select course unit.

	Student cancels course unit registration. Registrar is unable to find any course unit registration record.
Post-condition	Upon successful completion of the course unit registration: 1.Course unit registration is updated Upon VP (admin) is unable to find any course unit registration record, course unit registration confirmation page is not loaded.

Table 2- Use case narrative for course unit registration by student

Create attendance sheet Use case narrative	
Name	Create attendance sheet
Assumption	Students have registered for a particular course unit. Lecture was conducted
Use case description	Attendance sheet for a particular course unit by using its conducting date. The course unit may be a theory or practical.
Actors	Technical officer
Use case initialization	This use case starts after conducting a lecture.
Use case dialog	Technical office logins to the system. If login success, proceed otherwise technical office is asked to re enter login information. Course unit, date and number of hours are selected by technical office. Technical officer is selected whether course unit is a theory or practical. Attendance sheet is generated.
Use case termination	Attendance sheet is successful generated. Attendance sheet for particular course unit and date has been already generated. Connection failure with the system while attendance sheet is being generated.
Post-condition	Up on generating attendance sheet, technical officer is allowed to enter attendance.

	Up on attendance sheet already exists, error message is displayed. Up on connection failure with the system while generating attendance sheet, system returns to its initial state.

Table 3 - Use case narrative for attendance sheet creation

Issue exam admission Use case narrative	
Name	Issue exam admission
Assumption	Student login to the system correct.
Pre-condition	Student has registered for course units. Student attendance percentage is equal or greater than critical level.
Use case description	Examination form is issued to student which includes only eligible course units out of registered course units.
Actors	Students
Use case initialization	This use case starts while a student is requesting for exam admission.
Use case dialog	Student logins to the system. If login is correct, proceed, otherwise student is asked to re entered login details. System checks time period, if the time period between study year and examination period, system takes eligible course units which are followed by student and it is printed with examination admission. Otherwise system shows message, request cannot be processed because invalid time period.
Use case termination	Admission is issued successful.

	Student does not register for any course unit. Printer failure while hard copy is being printed.
Post-condition	Up on printing examination admission, student is received printed admission card. Up on student does not register for course unit, error message is displayed. Up on printer failure, system returns its initial state.

Table 4-Use case narrative for issuing semester exam admission

3.7.4.2 Class diagram

The system is represented as set of classes. Class diagrams are type of a static structure diagrams which explain the system structure. The Class diagram illustrates the system's classes, their attributes, methods, and the relationships among the main classes.

3.7.4.3 Sequence diagram

A sequence diagrams are used to show how objects interact in a given situation. Also it shows how messages are passed between objects. We can get idea about order of message passing by using a sequence diagram. Sequence diagrams for Assign or remove lectures for courses, for Course unit registration, for Enter and edit attendance sheet etc.

3.7.4.4 Activity diagram

Activity diagram iş basically a flow chart to represent the flow form one activity to another activity. The activity can be described as an operation of the system. So the control flow is drawn from one operation to another. This flow can be sequential, branched or concurrent. Work flow of the system activities are shown step by step. The activity diagram pointes out choices and parallel activities in swim lanes well. Student registration, course unit creation, assigning lectures for course unit, student attendance and issuing examination admission activities are described by the activity diagram.

3.7.5 Database design

Data which is used in the system will be described in a structure. Data design part reflects data base management system of the system. Relational database concept was used to design data base in the system. The data base was modeled by using Entity Relationship (ER) diagram. Also ER diagram is used to represent structure of the database. According to the ER diagram student, lecture, course unit, attendance, result and admission are entities. Each entity is described by using their attributes. There may be multi valued attributes also. ER diagram is a conceptual model of a data base, to implement the physical data base ER diagram should be mapped into schema diagram.

Student

stu_id	f_name	l_name	gender	Date Of Birth	Address
email	Session	Group_Id			

Contact

contact_number	stu_id

Lecture

teacher_id	gender	Date Of Birth	Address
f_name	l_name	Place Of Birth	Degree
Married	Phone	email	Salary

Participate

stu_id	teacher_id

Courseunit

code_id	name	department	credits	core	semester	requirement
lect_name	availability	Level				

Register

stu_id	code

Subjects

sub_id	Course's Name	Teacher's Name	Year	Subjects's name	Note

Attendance

atten	coureseunit	eligibility	stu_id

CheckAttendance

stu_id	a_id

AdmissionSubjects

a_id	stu_id	Subject

Result

r_id	grade	reg_number	subject	stu_id

PassOutStudents

stu_id	pass_out_year

Figure 3.7.5 – Database schema diagram

Chapter 4

User Interfaces

4.1. Introduction

Hope to discussed Interfaces which were designed for the MOTIS and quality factors of interface in the MOTIS explained in this chapter.

The system was designed according to Client – Server architecture. All interfaces were Web User Interfaces. Client can graphically interact with the system

4.2. Quality of User Interfaces

User interfaces were designed to meet client's requirements. When a client is working with a user interface, he / she can understand content of the user interface without any ambiguities.

User interface qualities of the system and User familiarity

User interfaces contain user familiar words rather than technical words. Interfaces had been designed student oriented as well as other faculty members. Course Unit Registration, View Eligible Subjects, Find Result, etc are student's familiar words which were used for interfaces.

Consistency

Display settings of the system were formatted. Forms, tables, messages, etc were formatted by using their font color, style, size. There was not many more back ground and appearance change while a user is moving one page to another page.

User guidance

Data format for some fields were displayed on interface. That was used for forms of the system. It helps to user to prevent data entering errors. Ex: Date format, student registration number format, course unit format.

User diversity

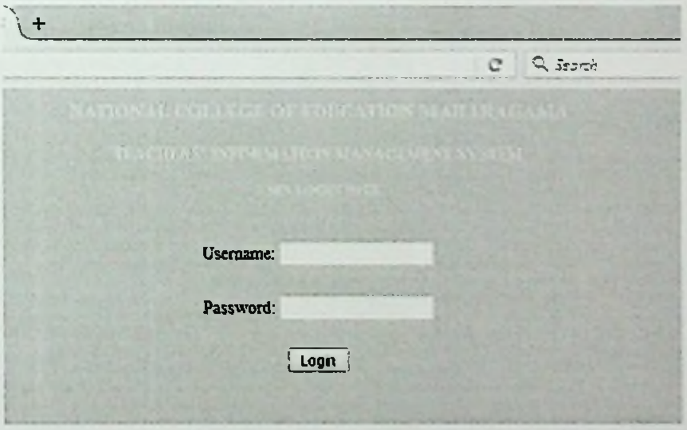
The system supports seven user groups. There are Administrators, principals, Department Heads, Lectures, Technical Officers, Office Users and Students. Interface for each group is same. But available functionalities are different.

4.3 Interfaces of MOTIS

Major user interfaces for each user group are briefly explained. All of interfaces are explained under User Manual (Appendix - B).

Login Form

Login form is placed in home page of the system. The form validates a user who is going to login to the system. It will give an error message if login is fail. Figure 4.1 shows the login form .



NATIONAL COLLEGE OF EDUCATION MAHARAGAMA

TEACHERS' INFORMATION MANAGEMENT SYSTEM

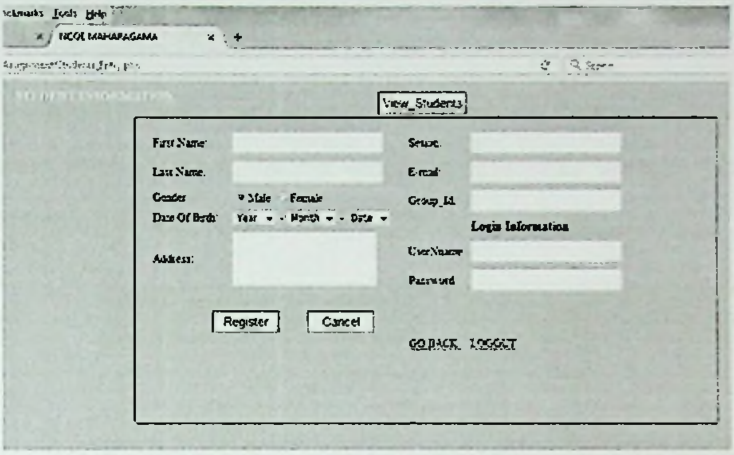
Username:

Password:

Figure 4.1 Login Form

Student Information Form

This form is used to enter student information into the system. Office user uses this form to enter newly attached student's information. Also the form is used to create new student user account for the system. The form is shown Figure 4.2



First Name:

Last Name:

Gender ● Male ○ Female

Date Of Birth: Year - Month - Date

Address:

Senior:

E-mail:

Group_Id:

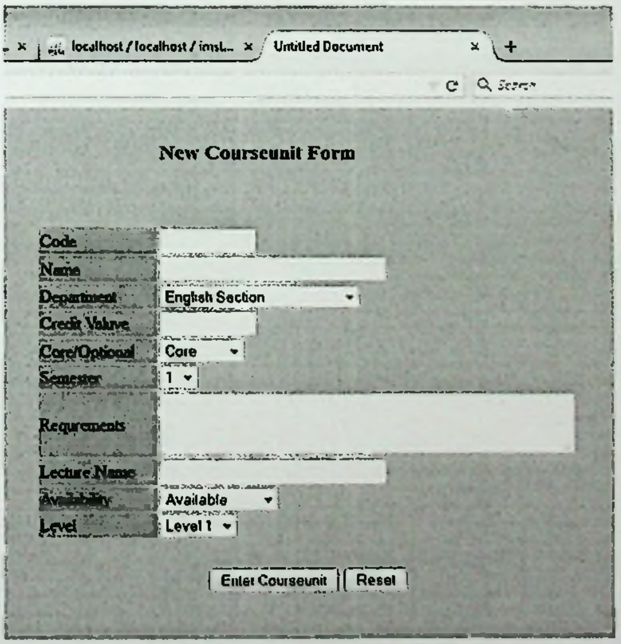
UserName

Password

Figure 4.2 Student Information Form

New Course unit Form

Course unit information which are offered by the college is entered to the system by using this from. The form is used by Office user in the college. New Course unit form is shown Figure 4.3.



Code

Name

Department: English Section

Credit Valuve

Core/Optional: Core

Semester: 1

Requrements

Lecture Name

Availability: Available

Level: Level 1

Figure 4.3 New Course unit Form

New Subjects Registration Form

Every student of the college must register for relevant course units beginning of academic year. After a student selecting his / her current year, the form is loaded with appropriate subjects. The form is shown Figure 4.4.

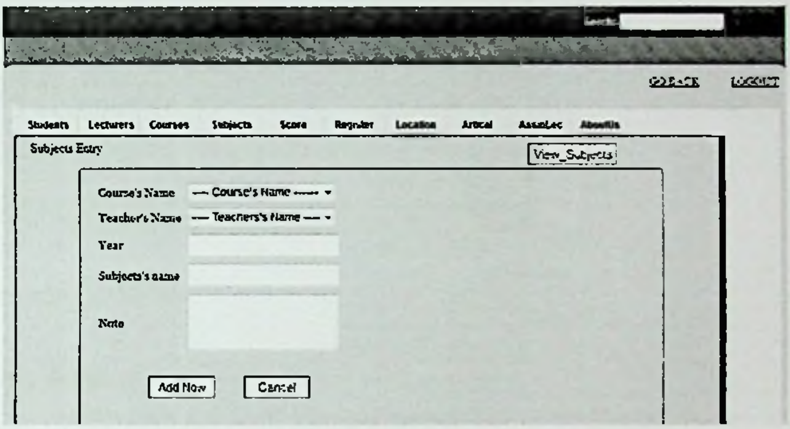


Figure 4.4 Course Unit Registration Form

Lecture Assigning Form

Lectures are assigned for course units by using this form. The vice president(academic) assign lectures for course units. The from is shown Figure 4.5

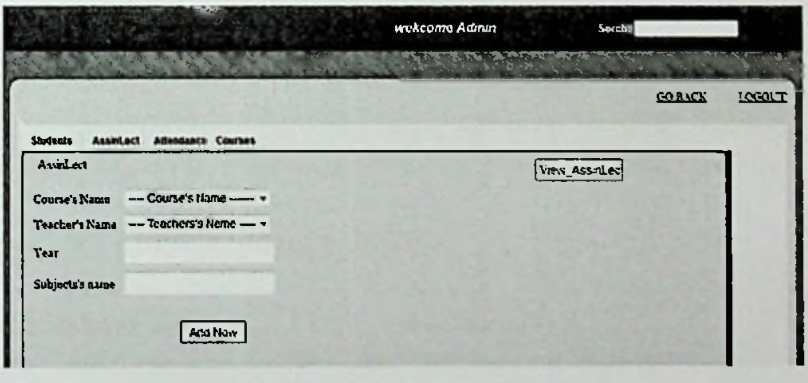
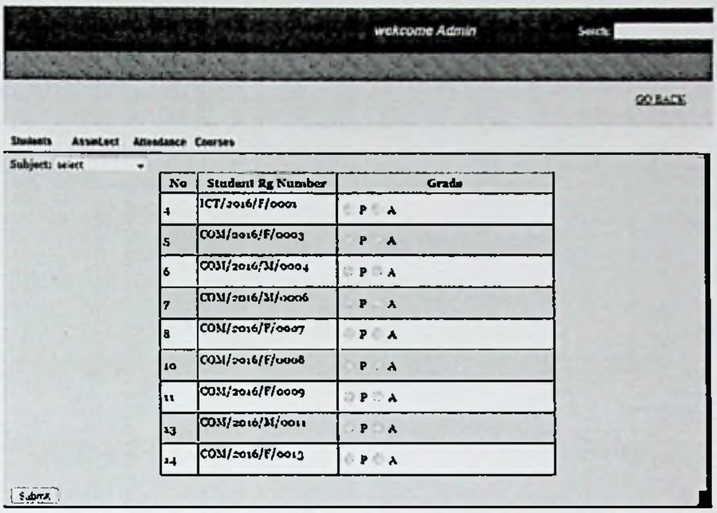


Figure 4.5 Lecture Assigning Form

Attendance Sheet

Student attendance is entered after each lecture. Technical Officers enter student attendance. To generate a attendance sheet technical officer should follow some steps which are discussed in User Manual. One of an attendance sheet is shown Figure 4.6.

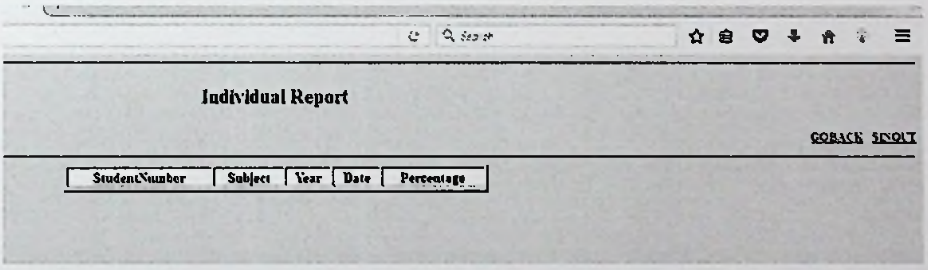


No	Student Rg Number	Grade
4	ICT/2016/F/0001	○ P ○ A
5	COM/2016/F/0003	○ P ○ A
6	COM/2016/M/0004	○ P ○ A
7	COM/2016/M/0006	○ P ○ A
8	COM/2016/F/0007	○ P ○ A
10	COM/2016/F/0008	○ P ○ A
11	COM/2016/F/0009	○ P ○ A
13	COM/2016/M/0011	○ P ○ A
14	COM/2016/F/0013	○ P ○ A

Figure 4.6 Attendance Sheet

Eligible Subjects Report

Students can check their subject eligibility which is based on attendance percentage before their final examination. Typically student should equal or exceed 80% attendance and 100% practical lesson cover to be eligible for a particular subject. Students are offered only eligible subjects for their final examination.



StudentNumber	Subject	Year	Date	Percentage

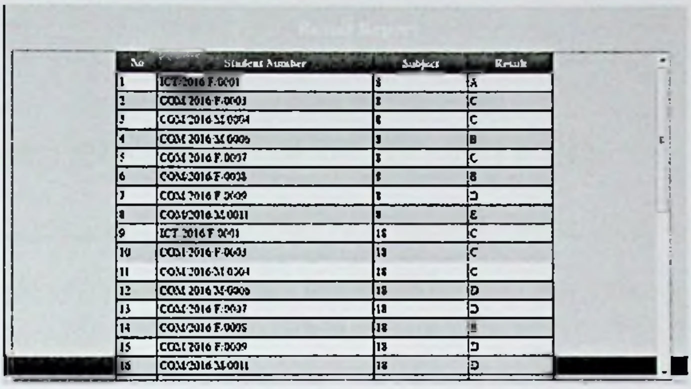
Figure 4.7 Eligible Subjects Report

Admission Card

An admission card is issued to each student before their examination. After preparing the admission card from office, student can take it from the system.

Result Sheet

Individual students can see their internal exam result. Result sheet for a student is shown Figure 4.8.



No	Student Number	Subject	Result
1	ICT/2016/F/0001	8	A
2	COM/2016/F/0003	8	C
3	COM/2016/M/0004	8	C
4	COM/2016/M/0006	8	B
5	COM/2016/F/0007	8	C
6	COM/2016/F/0008	8	B
7	COM/2016/F/0009	8	D
8	COM/2016/M/0011	8	E
9	ICT/2016/F/0001	18	C
10	COM/2016/F/0003	18	C
11	COM/2016/M/0004	18	C
12	COM/2016/M/0006	18	D
13	COM/2016/F/0007	18	D
14	COM/2016/F/0008	18	[illegible]
15	COM/2016/F/0009	18	D
16	COM/2016/M/0011	18	D

Figure 4.8 Result Sheet

Chapter 5

Implementation of the MOTIS

Interface developing, data base building and coding will done in this stage. Coding is done in modular based. In MOITS first data base is implementing before going to codlings.

5.1 Functions to be build

The MOTIS is build to perform bellow functions.

- 1 User account / student account management system
- 2 Course unit management system
- 3 Course unit registration system for students
- 4 Assigning lectures for course units
- 5 Attendance sheet generator
- 6 Eligibility for students
- 8 Student result management system
- 9 Pass out student information system

5.2 User Access Matrix

User access matrix is use to identify privileges of each user group. In the MOITS different functions can be accessed by different user groups. Table 5.1 shows user access matrix for the MOITS

User groups Functions	Administrators	Head of courses	Lecturer	Office staff	Technical officer	President	Student
Add student information	1	0	0	1	0	1	0
Edit student information	1	0	0	1	0	1	0
View student information	1	1	1	0	0	1	0
Add course unit	1	0	0	1	0	1	0
Edit course unit	1	0	0	1	0	1	0
Course unit registration	0	0	0	0	0	0	1
Cancel course unit registration	0	0	0	0	0	0	1
Assign / View / Remove lectures for course units	1	1	0	0	0	1	0
Enter / Edit student attendance	1	0	0	0	1	0	0
View student attendance	1	1	1	0	0	0	1
View exam eligibility	0	0	0	0	0	0	1
Receive exam admission	0	0	0	0	0	0	1
Enter / Edit exam result	0	0	0	1	0	0	0
View exam result	1	1	1	0	0	1	1
Set pass out student's information	0	0	0	0	0	1	0
View pass out student's information	1	1	1	0	0	1	0

1-Can access0- Cannot access

Table 5.1 - User access matrix

5.3 Module Structure

System module hierarchy is given by Figure 5.1. Different user groups can access different set of modules. It shows clear by the module structure diagram.

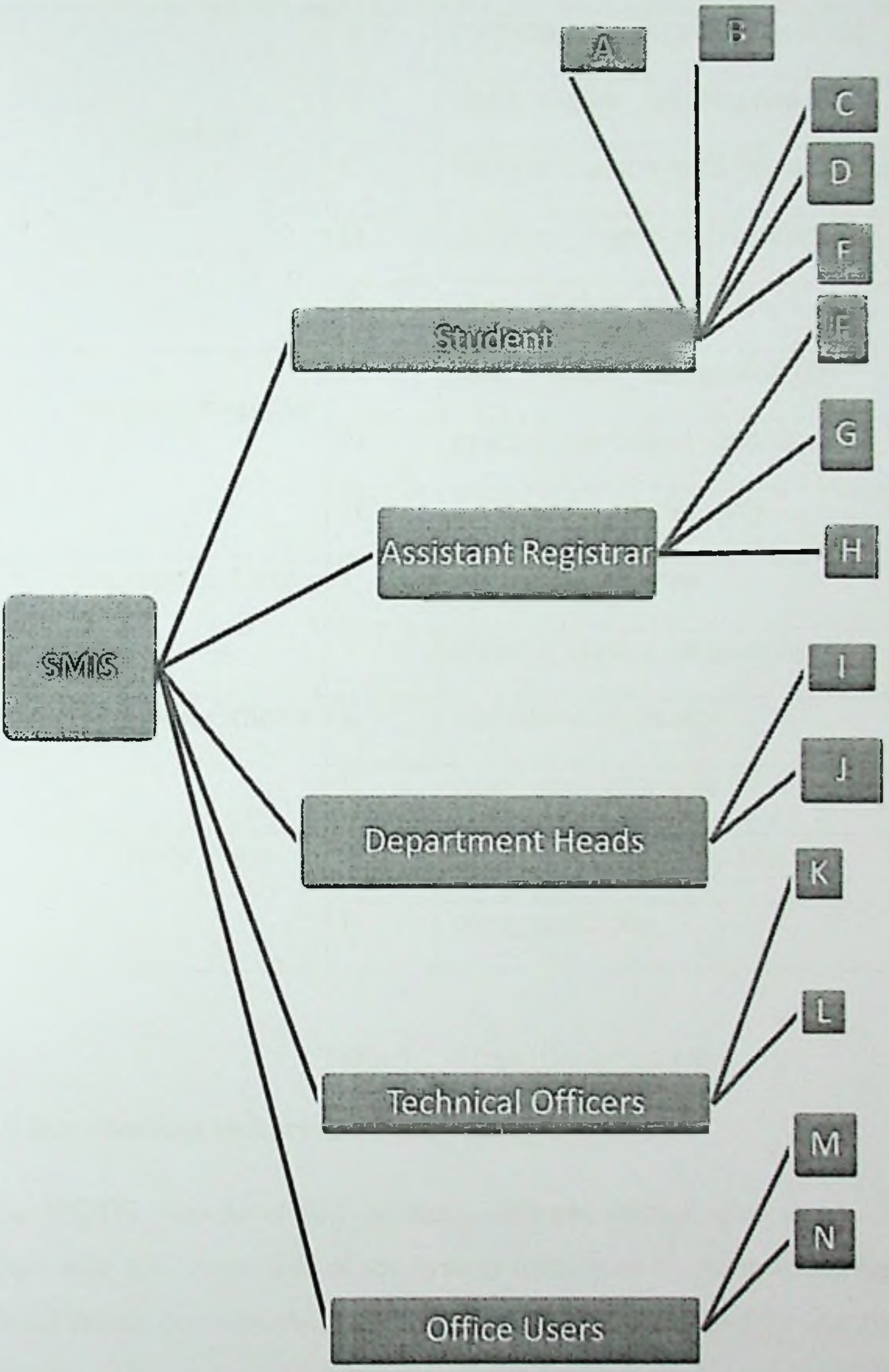


Figure 5.1 - Module Structure

Main file name for each module is given by Table 5.2. Those files are linked with other sub files also.

Main Module	Letter	Module Name
Student	A	courseunit_registration_form.php
	B	show_eligible_subjects.php
	C	issue_admission_card_to_student.php
	D	show_sp_degree_applications.php
	E	show_student_result.php
Assistant Registrar	F	enter_student_information.php
	G	process_admission_card.php
	H	set_passout_batches.php
Department Heads	I	assign_lectures.php
	J	approve_special_degree.php
Technical Officers	K	attendance_sheet.php
	L	enter_attendance.php
Office Users	M	enter_new_courseunit.php
	N	enter_result.php

Table 5.2 - Base file for modules

5.4 Development technologies and Tools

The MOTIS was developed by using different technologies and tools. Developing client side and server side of the system were done by those technologies and tools. Detail description of technologies and tools which were used for our system are given below.

5.4.1 Java Script

Java script was used to validate all forms which were contained in the system. The system is web based one so java script is ideal form validating technology. Because it can be embedded with web page easy. And it runs in client side which does not effect to server side traffic while large amount data is being processed. Empty fields finding, password matching, email address format checking, etc are done by java script. Then it was customized to meet our system requirements.

5.4.2 Ajax

This technology was used to load data, which was prepared from server side process into existing web page without refreshing the entire page. Another advantage is it reduces additional page usage for loading processed data. It is slowly minimize number of files for the system which leads to reduce system complexity.

5.4.3 LAMP and WAMP Server

LAMP means Linux, Apache, MySql and PHP. WAMP means Windows, Apache, MySql and PHP. Real system was installed into LAMP server but testing was done by using WAMP server. Linux or Windows is an operating system of a server computer. Apache is a web server. All modules of the system were installed into document root of the web server. MySql is data base server for the system. Data base of the system is managed by MySql server [14]. PHP is server side scripting language. All modules of the system were written in PHP. All commands which are given to the web server are done by PHP [12].

5.4.4 Dream Viewer Tool

This tool was used to design web interfaces. Not only that but also it was used for PHP coding, java script editing and style sheet designing. It was able to detect syntax errors in PHP coding within very short time period.

5.4.5 Implemented Environment

5.4.5.1 Server Computer

Hardware:

- Processor: 3 GHz or more
- RAM: 4 GB or more
- Hard Disk: 500 GB or more

Software:

- Operating system: WAMP server support

5.4.5.2 Client Computer

Hardware:

- Processor: 2 GHz
- RAM: 1 GB
- Hard Disk: 5 GB

Software

- Operating system: Windows 7
- Web browser: Mozilla Firefox
- Other: WAMP Server for testing computers

5.5 Major Code Samples

Main module codes will be explained in this section. Rest of codes will be discussed in Appendix C – Code Listing.

5.5.1 Database Connectivity

```
<?php  $host="localhost";  $db="testproject";  $user="root";  $pass="";  ?>
```

Parameter values for host, database name, user name and user password are set. It is saved as a separate file.

```
$con=mysql_connect($host,$user,$pass) or die ("Database server connection failure ");

mysql_select_db($db) or die ("Database failure ");
```

First a connection is created with data base server. Then particular data base is selected. Error will be given if any step is failure.

5.5.2 Check User Login

User name and password are passed into this module. First it checks whether user account is existed or not in the system. If user is existed, then system will check his / her group number. Based on group number the module loads student profile page or other user profile page.

```
<?php include  "logincheck.php"; ?>

<!DOCTYPE  html> <html>
```

```
<head>
<title></title>
</head> <body>
<header> <nav></nav>
</header> <div id="center">
<div id="center-set"> <div id="login"> <div id="login-st">
<form action ="" method ="POST" id="signin" id="reg">
 <?php
 $remarks = isset($_GET['remark_login']) ? $_GET['remark_login'] : '';
  if ($remarks==null and $remarks==""){
 echo '
<div id="reg-head" class="headrg"></div> ';    }
  if ($remarks=='failed'){
 echo '
<div id="reg-head-fail" class="headrg">Login Failed!, Invalid Credentials</div> ';
   }?>
<table border="0" align="center" cellpadding="2" cellspacing="0">
<tr id="lg-1"> <td class="tl-1">
<div align="left" id="tb-name">Username:</div></td>
<td><input type="text" id="tb-box" name="username" /></td>
 </tr> <tr>
 <td class="tl-1">&nbsp;</td>
 <td>&nbsp;</td> </tr>
<tr id="lg-1"> <td class="tl-1">
<div align="left" id="tb-name">Password:</div></td>
<td><input id="tb-box" type="password" name="password" /></td>
 </tr> <tr>
```

```
  <td  class="tl-1">&nbsp;</td>
  <td>&nbsp;</td> </tr>
</table>  <div  id="st">
  <div align="center">
    <input  name="submit"  type="submit"  value="Login"  id="st-btn"/>
  </div> </div> </form>
</body>
</html>
```

5.5.3 Student information entering module

This module enters student information into student's table.

```
<?php
        require("conection/connect.php");
$id=""; $opr="";
if(isset($_GET['opr']))
        $opr=$_GET['opr'];
if(isset($_GET['rs_id']))
        $id=$_GET['rs_id'];
//--------------add data-----------------
if(isset($_POST['btn_sub'])){
        $f_name=$_POST['fnametxt'];         $l_name=$_POST['lnametxt'];
        $gender=$_POST['gender'];
        $dob=$_POST['yy']."/".$_POST['mm']."/".$_POST['dd'];
        $addr=$_POST['addrtxt'];     $sesion=$_POST['sesiontxt'];
        $mail=$_POST['emailtxt'];    $utype=$_POST['utypetxt'];
        $uname=$_POST['unametxt'];          $pwd=md5(trim($_POST['pwdtxt']));
$sql_ins=mysql_query("INSERT INTO stu_tbl
        VALUES(     NULL,'$f_name','$l_name' ,  '$gender',
        '$dob',     '$addr',     '$sesion',     '$mail',
        '$utype',   '$uname',    '$pwd' )       ");
        $sql2_ins=mysql_query("INSERT INTO member
        VALUES(NULL,'$utype',     '$pwd', '$uname','$mail')");
if(($sql_ins==true)  && ($sql2_ins==true))
        $msg="1 Row Inserted";
else    $msg="Insert Error:".mysql_error();
         echo  $msg;}
if(isset($_POST['btn_upd'])){
        $f_name=$_POST['fnametxt'];          $l_name=$_POST['lnametxt'];
        $gender=$_POST['gender'];
        $dob=$_POST['yy']."/".$_POST['mm']."/".$_POST['dd'];
        $addr=$_POST['addrtxt'];      $sesion=$_POST['sesiontxt'];
        $mail=$_POST['emailtxt'];    $utype=$_POST['utypetxt'];
        $uname=$_POST['unametxt'];          $pwd=md5(trim($_POST['pwdtxt']));

        $sql_update=mysql_query("UPDATE stu_tbl SET   f_name='$f_name',
        l_name='$l_name' ,     gender='$gender',       dob='$dob',
        address='$addr', sesion='$sesion',       email='$mail',
        utype='$utype' uname='$uname',       pwd='$pwd'
        WHERE        stu_id=$id      ");
        if($sql_update==true)
                header("location:?tag=view_students");
        else
                $msg="Update Fail".mysql_error();   }
?>
<!DOCTYPE   html   PUBLIC   "-//W3C//DTD   XHTML   1.0   Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<link rel="stylesheet"  type="text/css" href="css/style_entry.css" />
<title>NCOE  MAHARAGAMA</title>
<style type="text/css">
```

```
<!--
.style3 {
        color: #0000FF;
        font-size: 18px;
        font-weight: bold;
}body {
        background-color: #99CCCC;
}.style4 {
        color: #FFFFFF;
        font-weight: bold;
}.style5 {color: #FFFFFF}
-->
</style> </head> <body>
<?php
if($opr=="upd"){
        $sql_upd=mysql_query("SELECT * FROM stu_tbl WHERE stu_id=$id");
        $rs_upd=mysql_fetch_array($sql_upd);
        list($y,$m,$d)=explode('-',$rs_upd['dob']);
?><!-- for form Upadte-->
<div id="top_style">
      <div id="top_style_text">
      Students Update </div>
      <!-- end of top_style_text-->
     <div id="top_style_button">
              <form method="post">
              <a href="?tag=view_students"><input type="button"
name="btn_view" title="View Students" value="Back" id="button_view"
style="width:70px;" /></a>
              </form>
     </div><!-- end of top_style_button-->
```

```
</div><!-- end of top_style-->

<div id="style_informations">

        <form method="post" >

        <div>

        <table border="0" cellpadding="4" cellspacing="0">

        <tr>

                <td>First Name:</td>

                <td>

                <input type="text" name="fnametxt" id="textbox" value="<?php echo
$rs_upd['f_name'];?>"/>

          </td>        </tr>

        <tr>

                <td>Last Name:</td>

                <td>

                <input type="text" name="lnametxt" id="textbox" value="<?php echo
$rs_upd['l_name'];?>"/>

          </td>        </tr>

        <tr>            <td>Gender:</td>

          <td>

                <input type="radio" name="gender" value="Male" <?php
if($rs_upd['gender']=="Male") echo "checked";?> />Male

             <input type="radio" name="gender" value="Female" <?php
if($rs_upd['gender']=="Female") echo "checked";?>/>Female

          </td>        </tr>

                <tr>

                <td>Date Of Birth:</td>

          <td>

                <select name="yy" >

                <option>years</option>

                <?php                                          $sel="";

                                               for($i=1985;$i<=2015;$i++){
                                                  if($i==$y){
                                            $sel="selected='selected'";}

                                                  else

                                                  $sel="";

                                            echo"<option   value='$i'   $sel>$i
</option>";                                       }
                 ?>


          </select>           -

        <select name="mm">

          <option>Month</option>

                                       <?php

                                            $sel="";


$mm=array("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","NOv","
Dec");                      $i=0;

             foreach($mm as $mon){

               $i++;
     if($i==$m){
     $sel=$sel="selected='selected'";}
                 else
          $sel="";

               echo"<option value='$i' $sel> $mon</option>";

             }

          ?>

        </select>          -

        <select name="dd">

          <option>Date</option>

                                       <?php

                                       $sel="";


          for($i=1;$i<=31;$i++){
     if($i==$d){
     $sel=$sel="selected='selected'";}
     else                                                  $sel="";
```

```
            ?>
            <option value="<?php echo $i ;?>"<?php echo $sel?> >
            <?php
            if($i<10)
               echo"0"."$i" ;
            else
               echo"$i";                                          ?>
                                                   </option>
                                                   <?php
                                                   }?>
                                          </select>
      </td>          </tr>          <tr>
            <td>Address:</td>               <td>
            <textarea   name="addrtxt"   cols="22"   rows="3">   <?php   echo
$rs_upd['address'];?></textarea>
                     </td>          </tr>
      <tr>
         <td colspan="2">
            <input type="reset" value="Cancel" id="button-in"/>
            <input type="submit" name="btn_upd" value="Update" id="button-in"
/>
         </td>          </tr>          </table>
  </div>       <div>
      <table border="0" cellpadding="4" cellspacing="0">
      <tr>            <td>Sesion:</td>               <td>
            <input type="text" name="sesiontxt" id="textbox" value="<?php echo
$rs_upd['sesion'];?>" />               </td>
      </tr>             <tr>
            <td>E-mail:</td>
         <td>
```

```
                <input type="text" name="emailtxt"  id="textbox" value="<?php echo
$rs_upd['email'];?> "/>             </td>

        </tr>                    <tr>

                <td>Group_Id: </td>

       <td><input type="text" name="utypetxt"  id="textbox" value="<?php echo
$rs_upd['utype'];?> "/></td>

       </tr>        <tr>

         <td>&nbsp;</td>

         <td><span class="style3">Login Information</span></td>

        </tr>         <tr>

         <td>UserNname:</td>

         <td><input type="text" name="unametxt"  id="textbox" value="<?php echo
$rs_upd['uname'];?> "/></td>

        </tr>         <tr>

         <td>Password: </td>

         <td><input type="password" name="pwdtxt"  id="textbox" value="<?php
echo $rs_upd['pwd'];?> "/></td>

        </tr>            </table> </div>            </form>

</div><!-- end of style_informatios -->

<?php  }else{?> <!-- for form Register-->

<div id="top_style">

      <div class="style4" id="top_style_text">

       <div align="center" class="style5">

        <div align="right">STUDENT INFORMATION</div>

       </div>      </div>

<!-- end of top_style_text-->

     <div id="top_style_button">

                <form method="post">

                <a          href="?tag=view_students"><input          type="button"
name="btn_view" title="View Students" value="View_Students" id="button_view"
style="width:120px;"  /></a>

                      </form>
```

```
    </div><!-- end of top_style_button-->
</div><!-- end of top_style-->
<div id="style_informations">
      <form method="post" >
  <div>
       <table border="0" cellpadding="4" cellspacing="0">
      <tr>
            <td width="101">First Name:</td>
       <td width="233">
      <input type="text" name="fnametxt" id="textbox"/>
        </td>
    </tr>                <tr>
            <td>Last Name:</td>
            <td>
            <input type="text" name="lnametxt" id="textbox"/>
        </td>       </tr>       <tr>
            <td>Gender:</td>
        <td>
<input type="radio" name="gender" value="Male" checked="checked" />Male
          <input type="radio" name="gender" value="Female"/>Female
        </td>       </tr>       <tr>
            <td>Date Of Birth:</td>
        <td>            <select name="yy" >
            <option>Year</option>
            <?php
for($i=1985;$i<=2015;$i++){
echo"<option value='$i'>$i</option>";
      }
      ?>            </select>
          <select name="mm">
```

```
                <option>Month</option>
          <?php
$mm=array("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","NOv","
Dec");                        $i=0;

                  foreach($mm as $mon){                        $i++;

                     echo"<option value='$i'> $mon</option>";

                  }                  ?>              </select>

              -                 <select name="dd">

                <option>Date</option>
          <?php                   for($i=1;$i<=31;$i++){                  ?>

                <option value="<?php echo $i; ?>">

                <?php

                if($i<10)                     echo"0".$i;                 else

                  echo"$i";      ?>     </option>
                <?php                                       }?>

                                  </select>

           </td>

       </tr>         <tr>

                <td>Address:</td>

                <td>

                <textarea name="addrtxt" cols="22" rows="3"></textarea>

                      </td>           </tr>

          <tr>            <td colspan="2">

                <input type="reset" value="Cancel" id="button-in"/>

                <input  type="submit"  name="btn_sub"  value="Register"  id="button-
in"  />            </td>          </tr>      </table>

     </div>       <div>

           <table border="0" cellpadding="4" cellspacing="0">

             <tr>      <td>Sesion:</td>

              <td><input type="text" name="sesiontxt" id="textbox" /></td>

       </tr>         <tr>      <td>E-mail:</td>

              <td><input type="text" name="emailtxt"  id="textbox" /></td>
```

```
    </tr>        <tr>
            <td>Group_Id: </td>
            <td><input type="text" name="utypetxt"  id="textbox" /></td>
    </tr>        <tr>      <td>&nbsp;</td>
            <td><span class="style3">Login Information</span></td>
    </tr>        <tr>
            <td>UserNname:</td>
            <td><input type="text" name="unametxt"  id="textbox" /></td>
    </tr>
        <tr>      <td>Password: </td>
            <td><input type="password" name="pwdtxt"  id="textbox" /></td>
    </tr>
        <tr>      <td>&nbsp;</td>
            <td>&nbsp;</td>       </tr>
        <tr>      <td>&nbsp;</td>
            <td>&nbsp;</td>       </tr>
        <tr>
            <td          colspan="2"><a          href="presidents.php">GO          BACK
</a>&nbsp;&nbsp; <a href="../index.php">LOGOUT</a></td>
    </tr>    </table>          </div>
  </form>
</div><!-- end of style_informatios -->
<?php ?>
</body> </html>
```

5.5.4 New Course Unit creating module

This module first captures parameters of a new course unit. Then that information is entered into course units table.

```
<?php $id=""; $opr=""; if(isset($_GET['opr']))   $opr=$_GET['opr']; if(isset($_GET['rs_id']))
        $id=$_GET['rs_id']; if(isset($_POST['btn_sub'])){
        $facuties_name=$_POST['fnametxt'];    $note=$_POST['notetxt'];
        $sql_ins=mysql_query("INSERT INTO facuties_tbl VALUES(
                              NULL,
          '$facuties_name',          '$note') "); if($sql_ins==true)    $msg="1 Row Inserted";else
          $msg="Insert Error:".mysql_error();

//------------------uodate data---------- if(isset($_POST['btn_upd'])){
          $fac_name=$_POST['fnametxt'];          $note=$_POST['notetxt'];
          $sql_update=mysql_query("UPDATE facuties_tbl SET
                              faculties_name='$fac_name',
                              note='$note'
          WHERE                    faculties_id=$id");      if($sql_update==true)
          header("location:?tag=view_faculties"); else              $msg="Update
Fail".mysql_error();?><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"><html
xmlns="http://www.w3.org/1999/xhtml"><head><meta http-equiv="Content-Type"
content="text/html; charset=utf-8" /><title>Maharagama NCOE</title><link rel="stylesheet"
type="text/css" href="css/style_entry.css" /><style type="text/css"> <!--body {
          background-color: #99CCFF;

}--></style></head><body>

<?php  if($opr=="upd") {          $sql_upd=mysql_query("SELECT * FROM facuties_tbl
WHERE faculties_id=$id");      $rs_upd=mysql_fetch_array($sql_upd);   <div
id="top_style">       <div id="top_style_text">              Courses Update       </div><!-- end
of top_style_text-->      <div id="top_style_button">                <form method="post">
          <a href="?tag=view_faculties"><input type="button" name="btn_view"
value="Back" title="View_faculties" id="button_view" style="width:70px;"  /></a>
                    </form>

     </div><!-- end of top_style_button--> </div><!-- end of top_style-->

<div id="style_informations">  <form method="post">           <div>

          <table border="0" cellpadding="4" cellspacing="0">           <tr>
          <td>Courses's Name</td>

          <td>                <input type="text" name="fnametxt" id="textbox" value="<?php
echo $rs_upd['faculties_name'];?>" />

            </td>          </tr>          <tr>          <td>Note</td>           <td>

   <textarea name="notetxt" cols="23" rows="4"><?php  echo $rs_upd['note'];

?></textarea>                </td>            </tr>           <tr>               <td colspan="2">
          <input type="reset" value="Cancel" id="button-in"/>

<input type="submit" name="btn_upd" value="Update" id="button-in"  />              </td>
</tr>         </table>  </div>   </form></div><!-- end of style_informatios -->

 <?php   } else {?>

<div id="top_style">    <div id="top_style_text">
```

```
        Courses Entry

      </div><!-- end of top_style_text-->

     <div id="top_style_button">

                  <form method="post">

                   <div align="right"><a href="?tag=view_faculties">

                <input type="button" name="button_view" title="View_faculties"
value="View_Courses" id="button_view2" style="width:120px;"  />

                        </a>                        </div>                 </form>

                  </div><!-- end of top_style_button-->

</div><!-- end of top_style-->


<div id="style_informations">

        <form method="post">           <div>

        <table border="0" cellpadding="4" cellspacing="0">

       <tr>               <td>Courses's Name</td>

        <td>              <input type="text" name="fnametxt" id="textbox" />            </td>

       </tr>         <tr>

        <td>Note</td>           <td>

                  <textarea name="notetxt" cols="23" rows="4"></textarea>           </td>

       </tr>

       <tr>

          <td colspan="2">

                  <input type="reset" value="Cancel" id="button-in"/>

                  <input type="submit" name="btn_sub" value="Add Now" id="button-in"  />
</td>         </tr>        <tr>          <td colspan="2">&nbsp;</td>

       </tr>         <tr>          <td colspan="2"><a href="../main_menu.html">GO BACK
</a></td>         </tr>        <tr>

         <td colspan="2"><a href="../index.php">SIGNOUT</a></td>

       </tr>         </table>   </div>   </form>

</div><!-- end of style_informatios -->
```

```
<?php } ?> </body>

</html>
```

5.5.5 Course Unit Registration Module

Student number and his / her registered course unit's numbers for a particular year are entered into courseunit_ register table by this module. Before that task the module takes all registered course units by a student.

```
<?php

$id=""; $opr=""; if(isset($_GET['opr'])) $opr=$_GET['opr'];        if(isset($_GET['rs_id']))
        $id=$_GET['rs_id'];     if(isset($_POST['btn_sub'])){    $fa_name=$_POST['factxt'];
        $teach_name=$_POST['techtxt'];         $semester=$_POST['semestertxt'];
        $sub_name=$_POST['subtxt'];  $note=$_POST['notetxt'];

$sql_ins=mysql_query("INSERT INTO sub_tbl  VALUES(      NULL, '$fa_name',
'$teach_name' '$semester',        '$sub_name' ,    '$note'  ) ");     if($sql_ins==true)
        $msg="1 Row Inserted"; else   $msg="Insert Error:".mysql_error();     }

if(isset($_POST['btn_upd'])){   $fac_id=$_POST['factxt'];       $tea_id=$_POST['techtxt'];
        $semester=$_POST['semestertxt'];        $sub_name=$_POST['subtxt'];

        $note=$_POST['notetxt'];                $sql_update=mysql_query("UPDATE sub_tbl
SET     faculties_id='$fac_id' ,  teacher_id='$tea_id' ,    semester='$semester' ,
        sub_name='$sub_name' ,         note='$note'     WHERE sub_id=$id     ");
        if($sql_update==true)   header("location:?tag=view_subjects");

 else    $msg="Update Fail!..."; }?>

<!DOCTYPE    html    PUBLIC    "-//W3C//DTD    XHTML    1.0    Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd"><html
xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta   http-equiv="Content-Type"   content="text/html;   charset=utf-8"   /><title>NCOE
Maharagama</title>
```

```
<link rel="stylesheet" type="text/css" href="css/style_entry.css" />

<style type="text/css"> <!--

body {  background-color: #99CCFF;

} -->

</style></head>

<body>

<?php

if($opr=="upd") {

          $sql_upd=mysql_query("SELECT * FROM sub_tbl WHERE sub_id=$id");

          $rs_upd=mysql_fetch_array($sql_upd);    ?>

<div id="top_style">

      <div id="top_style_text">

      Subjects Entry

      </div><!-- end of top_style_text-->

     <div id="top_style_button">

                    <form method="post">

          <a href="?tag=view_subjects" ><input type="button" name="btn_view" title="Back"
value="Back" id="button_view" style="width:70px;"  /></a>

                    </form>

     </div><!-- end of top_style_button-->

</div><!-- end of top_style-->


<div id="style_informations">

          <form method="post">

          <div>

          <table border="0" cellpadding="5" cellspacing="0">

          <tr>

          <td>Course's Name</td>           <td>

                    <select name="factxt" id="textbox">
                    <option>---- Course's Name   ------</option>

                     <?php

                       $fac_name=mysql_query("SELECT * FROM facuties_tbl"
while($row=mysql_fetch_array($fac_name)){
if($row['faculties_id']==$rs_upd['faculties_id'])
                    $iselect="selected";
                    else
          $iselect="";      ?>

<option value="<?php echo $row['faculties_id'];?>" <?php echo $iselect;?> > <?php echo
$row['faculties_name'];?> </option>

                         <?php   }                    ?>

                <select>            </td>          </tr>                   <tr>

          <td>Teacher's Name</td>

          <td>              <select name="techtxt" id="textbox">

                    <option>---- Teachers's Name   ----</option>

                     <?php                         $te_name=mysql_query("SELECT * FROM
teacher_tbl");
          while($row=mysql_fetch_array($te_name)){
                    if($row['teacher_id']==$rs_upd['teacher_id'])
                                                        $iselect="selected";
                                                        else
                                                $iselect="";
                              ?>

  <option value="<?php echo $row['teacher_id'];?>" <?php echo $iselect?> > <?php echo
$row['f_name'] ; echo " "; echo $row['l_name'];?> </option>
                                             <?php   }                  ?>            </select>

            </td>          </tr>        <tr>            <td>Year</td>           <td>

                    <input type="text" name="semestertxt" id="textbox" value="<?php echo
$rs_upd['semester'];?>"  />

            </td>          </tr>        <tr>

          <td>Subjects's name</td>              <td>

 <input type="text" name="subtxt"  id="textbox" value="<?php echo
$rs_upd['sub_name'];?>" />

            </td>          </tr>         <tr>            <td>Note</td>            <td>

    <textarea name="notetxt" cols="23" rows="3"><?php echo $rs_upd['note'];?></textarea>

            </td>          </tr>         <tr>            <td colspan="2">

            <input type="reset" value="Cancel" id="button-in"/>
            <input type="submit" name="btn_upd" value="Update" id="button-in"  />

            </td>          </tr>       </table>          </div>

   </form>  </div><!-- end of style_informatios -->

<?php  } else { ?>

<div id="top_style">       <div id="top_style_text">

      Subjects Entry       </div><!-- end of top_style_text-->

     <div id="top_style_button">                    <form method="post">

<a href="?tag=view_subjects" ><input type="button" name="btn_view" title="View
Subjects" value="View_Subjects" id="button_view" style="width:120px;"  /></a>

     </form>      </div><!-- end of top_style_button-->

</div><!-- end of top_style-->

<div id="style_informations">   <form method="post">

         <div>            <table border="0" cellpadding="5" cellspacing="0">              <tr>

         <td>Course's Name</td>          <td>            <select name="factxt"
id="textbox">            <option>---- Course's Name   ------</option>

 <?php

  $fac_name=mysql_query("SELECT * FROM facuties_tbl");
while($row=mysql_fetch_array($fac_name)){
              ?>

<option value="<?php echo $row['faculties_id'];?>"> <?php echo $row['faculties_name'];?>
</option>                    <?php   }           ?>

            </select>          </td>        </tr>

              <tr>              <td>Teacher's Name</td>               <td>

               <select name="techtxt" id="textbox">

               <option>---- Teachers's Name   ----</option>

                <?php                     $te_name=mysql_query("SELECT * FROM
teacher_tbl");
         while($row=mysql_fetch_array($te_name)){              ?>

  <option value="<?php echo $row['teacher_id'];?>"> <?php echo $row['f_name'] ; echo " ";
echo $row['l_name'];?> </option>
         <?php  }            ?>

            </select>         </td>        </tr>       <tr>

         <td>Year</td>          <td>
```

```
                <input type="text" name="semestertxt" id="textbox"  />
          </td>          </tr>          <tr>
        <td>Subjects's name</td>             <td>
                <input type="text" name="subtxt"  id="textbox" />
          </td>          </tr>          <tr>
        <td>Note</td>             <td>
 <textarea name="notetxt" cols="23" rows="3"></textarea>
          </td>          </tr>            <tr>
          <td colspan="2">
   <input type="reset" value="Cancel" id="button-in"/>
          <input type="submit" name="btn_sub" value="Add Now" id="button-in"  />
          </td>          </tr>               </table>          </div>
   </form>
</div><!-- end of style_informatios -->
<?php }       ?>
</body>
</html>
```

5.5.6 Lectures Assigning Module

Lectures are assigned for course unit by using this module. The module first stores lecture name and course unit number. Then that information is stored into assigned_ lectures table. Multiple lectures can be assigned for a particular course unit.

```
<?php
$id=""; $opr=""; if(isset($_GET['opr'])) $opr=$_GET['opr'];          if(isset($_GET['rs_id']))
        $id=$_GET['rs_id'];      if(isset($_POST['btn_sub'])){    $fa_name=$_POST['factxt'];
        $teach_name=$_POST['techtxt'];          $semester=$_POST['semestertxt'];
        $sub_name=$_POST['subtxt'];

$sql_ins=mysql_query("INSERT INTO assin_tbl
        VALUES(NULL,        '$fa_name',      '$teach_name',  '$semester',       '$sub_name'
        )");if($sql_ins==true)   $msg="1 Row Inserted";
```

```
else     $msg="Insert Error:".mysql_error();

f(isset($_POST['btn_upd'])){

         $fac_id=$_POST['factxt'];        $tea_id=$_POST['techtxt'];
         $semester=$_POST['semestertxt'];        $sub_name=$_POST['subtxt'];
         $sql_update=mysql_query("UPDATE assin_tbl SET
         faculties_id='$fac_id' ,
         teacher_id='$tea_id' ,

semester='$semester' ,sub_name='$sub_name' WHERE assin_id=$id     ");

if($sql_update==true)

         header("location:?tag=view assignLect");

else     $msg="Update Fail!...".mysql_error();}?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">                <html
xmlns="http://www.w3.org/1999/xhtml"> <head>

<meta          http-equiv="Content-Type"          content="text/html;          charset=utf-8"
/><title>NCOEM</title> <link rel="stylesheet" type="text/css" href="css/style_entry.css" />

</head>  <body> <?php  if($opr=="upd")  {

         $sql_upd=mysql_query("SELECT  *  FROM   assin_tbl  WHERE  assin_id=$id");
         $rs_upd=mysql_fetch_array($sql_upd); ?>

<div id="top_style">     <div id="top_style_text">       AssinLect      </div>

  <a  href="?tag=vew  assignLect"  ><input  type="button"  name="btn_view"  title="Back"
value="Back" id="button_view" style="width:70px;"  /></a>

<form method="post">          <div>          <p>&nbsp;</p>

<table border="0" align="center" cellpadding="5" cellspacing="0">        <tr>
         <td>Course's Name</td>          <td>

 <select name="factxt" id="textbox">          <option>---- Course's Name   ------</option>
<?php
```

```
                    $fac_name=mysql_query("SELECT * FROM facuties_tbl");

  while($row=mysql_fetch_array($fac_name)){
                   if($row['faculties_id']==$rs_upd['faculties_id'])
        $iselect="selected";
        else                                    $iselect="";               ?>

 <option value="<?php echo $row['faculties_id'];?>" <?php echo $iselect;?> > <?php echo
$row['faculties_name'];?> </option>

      <?php        }          ?>     </select>         </td>         </tr>         <tr>

        <td>Teacher's Name</td>   <td>          <select name="techtxt" id="textbox">

        <option>---- Teachers's Name   ----</option>       <?php

  $te_name=mysql_query("SELECT * FROM teacher_tbl");
        while($row=mysql_fetch_array($te_name)){
        if($row['teacher_id']==$rs_upd['teacher_id'])

$iselect="selected";  else          $iselect="";      ?>

 <option value="<?php echo $row['teacher_id'];?>" <?php echo $iselect?> > <?php echo
$row['f_name'] ; echo " "; echo $row['l_name'];?> </option>

<?php               ?>

  </select>    </td>  </tr>     <tr>        <td>Year</td>  <td>

<input type="text" name="semestertxt" id="textbox" value="<?php echo
$rs_upd['semester'];?>"  />

   </td>  </tr>  tr> <td>Subjects's name</td>            <td>

  <input type="text" name="subtxt"  id="textbox" value="<?php echo
$rs_upd['sub_name'];?>" />   </td>     </tr>    <tr>

   <div id="style_informations">         <tr>            <td colspan="2"

<input type="submit" name="btn_upd" value="Update" id="button-in"  />

            </td> </tr> </table>  </div>     </div>  </form>

<?php } else { ?>

<div id="top_style">      <div id="top_style_text">        AssinLect       </div>

<!-- end of top_style_text-->      <div id="top_style_button">     <form method="post">

<a href="?tag=view assignLect" ><input type="button" name="btn_view" title="view
AssinLect" value="View_AssinLect" id="button_view" style="width:120px;"  />

</a>     </form>      </div><!-- end of top_style_button--></div>

<form method="post">            <div>   <table border="0" cellpadding="5" cellspacing="0">
```

```
          <tr>  td>Course's Name</td>       <td>  <select name="factxt" id="textbox">

<option>---- Course's Name   ------</option>                        <?php

   $fac_name=mysql_query("SELECT * FROM facuties_tbl");
while($row=mysql_fetch_array($fac_name)){

?>

<option value="<?php echo $row['faculties_id'];?>"> <?php echo $row['faculties_name'];?>
</option>

  <?php   }     ?>

 </select>    </td>    </tr>      <tr>

<td>Teacher's Name</td>          <td>

 <select name="techtxt" id="textbox">

 <option>---- Teachers's Name   ----</option>

  <?php

  $te_name=mysql_query("SELECT * FROM teacher_tbl");

while($row=mysql_fetch_array($te_name)){?>

   <option value="<?php echo $row['teacher_id'];?>"> <?php echo $row['f_name'] ; echo " ";
echo $row['l_name'];?> </option>

                              <?php                        }                ?>

  </select>     </td>   </tr>    <tr> <td>Year</td>          <td>

<input type="text" name="semestertxt" id="textbox"  />

 </td>          </tr>                   <tr>

          <td>Subjects's name</td>              <td>

   <input type="text" name="subtxt"  id="textbox" />

            </td>          </tr>          <tr>              <td colspan="2">

<input type="submit" name="btn_sub" value="Add Now" id="button-in"  />

  </td>          </tr>                 </table>   </div>   </form>

<!-- end of style_informatios --><?php }?>

</body>

</html>
```

5.5.7 Attendance Sheet Module

Attendance sheet is handled by this module. Before data is transferred to the module, attendance sheet was generated by another module. The module receives student attendance status for a particular lecture. Also conducted lecture information is received by previous module. Then that student attendance status information is stored into participation table.

```
<?php

$msg="";        $opr="";        if(isset($_GET['opr']))  $opr=$_GET['opr'];
if(isset($_GET['rs_id'])) $id=$_GET['rs_id'];      if(isset($_POST['submit'])){
//var_dump($_POST);

die;      $count = 0;      $sql3_ins = ";

    $subject = $_POST['stu_tbl']; $date = date("Y-m-d");

    foreach($_POST['result']        as        $reg_number        =>        $attendance){
$sql3_ins=mysql_query("INSERT   INTO   attend   VALUES(   NULL,   '$reg_number',
'$attendance', '$subject','$date')");

    $count ++;      }      if($sql3_ins==true)        $msg= $count ." Row(s) Inserted";

    Else

      $msg="Insert Error:".mysql_error();        }

echo $msg; ?>

<!DOCTYPE    html    PUBLIC    "-//W3C//DTD    XHTML    1.0    Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml"> <head> <meta http-equiv="Content-Type"
content="text/html;    charset=utf-8"    />    <title>NCOEM</title><link    rel="stylesheet"
type="text/css"

ref="../../../../Users/neranga/Users/neranga/Downloads/css/style_view.css" />

<style type="text/css"><!—

.style2 {font-size: 24} > </style> </head>

body><!--- end of style_div --> <div id="content-input"> <form method="post">
```

```
 Subject:  <select name= "stu_tbl">   <option> select</option>

<?php

 $sb_name=mysql_query("SELECT * FROM sub_tbl");

while($row=mysql_fetch_array($sb_name)){
        if($row['sub_id']==$rs_upd['sub_id'])
        $iselect="selected";              else    $iselect="";     ?>

<option  value="<?php  echo  $row['sub_id'];?>"  <?php  echo  $iselect?>  >  <?php  echo
$row['sub_name'] ; ?> </option>

<?php           }                   ?>   </select>

 <table   border="1"   width="541"   align="center"   cellpadding="3"   class="mytable"
cellspacing="0">       <tr>

 <th width="40">No</th>         <th width="173"="2">Student Rg Number</th>

  <th width="272"="2">Grade</th>      </tr>

    <?php

        $key="";        if(isset($_POST['searchtxt']))            $key=$_POST['searchtxt'];
                if($key !="")

$sql_sel=mysql_query("SEIECT * FROM register WHERE classname  like '%$key%' ");

else      $sql_sel=mysql_query("SELECT * FROM register");

  $i=0;    while($row=mysql_fetch_array($sql_sel)){

  $i++;

  $color=($i%2==0)?"lightblue":"white";   ?>       <tr bgcolor="<?php echo $color?>">

        <td height="20"><?php echo $row['reg_id'];?></td>

        <td><?php echo $row['reg_number'];?>

 <label><!--<input     type="hidden"     name="present_st[]"     value="<?php     /*echo
$row['pob'];*/?>"/>-->

                <p>       </p></td> <td>
<input type="radio" name="result[<?php echo $row['reg_number'];?>]" id="box2" value="P"
/> P

   <input type="radio" name="result[<?php echo $row['reg_number'];?>]" id="box2"
value="A" /> A   </td>       </tr>

       <?php   }

   ?>

     </table><p>        <label>

       <input type="submit" name="submit" id="submit" value="Submit" />

       </label>   </p>  </form>

</div>

<form id="form1" name="form1" method="post" action="">

</form> <div align="center"></div>

</body>

</html>
```

5.5.8 Eligible Subjects Report generating Module

While a student is requesting to check his / her eligible subjects for current year this module shows result based on student's attendance up to date. Also the report contains percentage attendance for each subject.

```
<?php

session_start();

$st_number=$_SESSION["user_name"];                    $level=$_POST["level"];

$semester=$_POST["year"];


//Get student s registerd sublects by filtering level and semester

//Calculate attendance percentage for subjects

//Give elgibility according to attendance percentage

echo '<link rel="stylesheet" type="text/css"
href="http://localhost/imscsit/style_sheet/profile.css" />';

echo '<link rel="stylesheet" type="text/css"
href="http://localhost/mscsit/style_sheet/error_msg.css" />';
```

```
echo '<table>';

echo '<tr> <td><p class="title">NATIONAL COLLEDE OF EDUCATION
MAHARAGAM; Technology <p class="title">mscsit</td> </tr>';

echo '<tr><td><p class="normal">Cuttoff Mark is 80%</p></td></tr>';

echo '<tr><td><p class="normal">Attendence percentage and eligibility are calculated up to
today, Not for entire lecture series</p></td></tr>';

echo '<tr><td><p class="normal">Level '.$level.' year '.$year.'</p></td></tr>';

echo '<tr><td><table>';

echo '<tr><th>Subject</th><th>Percentage</th><th>Status</th></tr>';

include 'config.php';

$con=mysql_connect($host,$user,$pass) or die ("Database server connection failure ");

mysql_select_db($db) or die ("Database failure ");



$query1="SELECT A.code FROM courseunits A,courseunit_register B WHERE
A.semester=$semester AND A.level=$level AND A.code=B.code AND
B.st_number='$st_number'";

$result1=mysql_query($query1);

        while($row1=mysql_fetch_array($result1))

        {

        //echo $row1['code']."<br/>";

        //count how many lectures are held

        $st_attendence=0;

        $code=$row1['code'];

        $query2="SELECT lecture_id FROM lecture WHERE course='$code'";

        $result2=mysql_query($query2);

        $total_num_of_lectures=mysql_num_rows($result2);

                echo '<tr>';

        echo '<td> <p class="normal">'.$code.'</p></td>';

        //count how many lectures were participated by student

                while($row2=mysql_fetch_array($result2))
```

```
                    {
                    $lect_id=$row2['lecture_id'];
                    $query3="SELECT status FROM participation WHERE lect_id=$lect_id
AND st_number='$st_number'";
                    $result3=mysql_query($query3);
                    //$row3=mysql_fetch_array($result3);
                    if(mysql_num_rows($result3) > 0)
                    $st_attendence++ ;
                            }
                    //find percentage
            $percentage=($st_attendence/$total_num_of_lectures) * 100;
            echo '<td> <p class="normal">'.$percentage.' %</p></td>';
                                    if($percentage >= 80)
                                    echo '<td><p class="normal">Eligible</p></td>';
                                    else
                                    echo '<td><p class=error>Not Eligible</p></td>';
                                    echo '</tr>';}
//echo $st_number;
echo '</table></td></tr>';
echo '<tr><td><a href="../pages/eligible_subjects_form.php">Back</a></td></tr>';
echo '</table>';
?>
```

5.5.9 Examination Admission Module

This module issues admission card of final examination for each student. First the module stores student registration number and year. Then it will display registered course units for that year . Also not eligible subjects are highlighted.

```
<?php   $year=$_POST["year"];
include 'config.php';
```

```
$con=mysql_connect($host,$user,$pass) or die ("Database server connection failure
");

mysql_select_db($db) or die ("Database failure ");

        if($year == 2)

$query="SELECT A.code,B.name,A.status FROM exam_admission A,courseunits B
WHERE A.st_number='$user_name' AND A.year=$year AND B.year=1 AND
A.code=B.code AND B.level=$level";

        else

                $result=mysql_query($query);

?>

<body>

<form name="form1" method="post" action="">

 <table width="932" border="0">   <tr>

   <td width="926"><p class="title">Teachers' Collge; Technology

   <p class="title">motis</td>

  </tr>   <tr>     <th>Admission Card </th>    </tr>    <tr>

   <th>Year <?php echo $year; ?> Semester <?php echo $year; ?></th>

  </tr>   <tr>

   <td><table width="925" border="2">

    <tr>

     <th width="69">Code</th>

     <th width="286">Subject</th>

     <th width="112">Eligibility</th>

     <th width="125">Date</th>

     <th width="141">Student Signature </th>

     <th width="150"> Invigilator Signature </th>

    </tr>

                <?php
```

```
            while($row=mysql_fetch_array($result))
            {
            $status=$row['status'];
                if($status=='Eligible')
                echo '<tr>';
                else
                echo '<tr bgcolor=red>';
            echo '<td>'.$row['code'].'</td>';
            echo '<td>'.$row['name'].'</td>';
            echo '<td>'.$status.'</td>';
            echo '<td>&nbsp;</td>';
            echo '<td>&nbsp;</td>';
            echo '<td>&nbsp;</td>';
            echo '<tr>';
            }
            mysql_close($con);
        ?>
  </table></td>
</tr>
<tr>
  <td>&nbsp;</td>
</tr>
<tr>
  <td><?php
        if($group_id == 7)
        echo '<a href="../profile/student_profile.php">Back</a>';
        else
```

```
            echo '<a href="../profile/other_profile.php">Back</a>';
            ?>  </td>   </tr>  </table>
</form>
</body>
</html>
```

Chapter 6

Testing MOTIS

6.1 Introduction

The system was tested against client requirements and potential. This system is web based system. Different parameters were tested. There were content of pages, functionalities, navigation between pages, usability, performance and security. Content testing, interface testing, navigation testing, component testing, configuration testing, performance testing and security testing were done against the system to detect errors. Errors were fixed after they had been detected.

6.2 Testing Process

6.2.1 Content Testing

This test was performed to check whether the system meets user level goals. The content which was displayed on the monitor screen was focused. Spellings and grammar checking, image alignments and sizes, appearance of menus were checked. Also accuracy of the system was checked under content testing.

6.2.2 Interface Testing

User interface were checked. Unit test was used to check interface individually. Appearance of text and images were checked. Also font colors and styles were checked whether user eye friendly or not. Forms which contained with interface were checked against its validation script. Empty field detection, password matching, e mail format checking, etc were done under that validation script checking. Also dynamic content which was loaded to a interface such as AJAX was tested whether output alignment, format, etc correct or not.

6.2.3 Navigation Testing

Hyper links were tested under navigation testing. Dead links were detected. Improper links and error links were detected. Also that test was used to detect not existence destination URLs after redirecting.

6.2.4 Component Testing

Functioning errors of individual modules were tested. Some module had more than one functions, those type of module was tested against each functions. First black box testing was performed against outside of the module. Then white box test was performed inside of the module to check functions. Also database was tested under component testing.

6.2.5 Configuration Testing

Configuration settings of server computer and client computers were tested. Additionally that testing servers were also tested.

Server Computer:

Service status of MySql server and Apache web server were tested. Then document root and MOTIS system files compatibility was tested. Before running actual PHP files of the system, test PHP files were run and tested.

Client Computers:

Existence and loading of Mozilla Firefox web browser was tested. Then Java Script, Popup window and Firewall enabling were tested.

Testing Servers:

WAPM server setting was checked. MySql and Apache web server status were tested. Running of server side scripting was tested. MOTIS and document root compatibility was tested.

6.2.6 Usability Testing

Under usability testing the system was tested for how easy user to access the system. User interfaces were mainly focused for that test. Interface interactivity with users, easy navigation, readability of text, appearance of graphics, Display layout (Resolution and Size), time delay to load some result were tested by usability testing.

6.2.7 Performance Testing

Web server computer performance was mainly focused. Other than that LAN and client computer performance were tested. Under that test, we tested how to server handle large amount of data, multiple user login with the server, time delay to fulfill a particular service by server computer.

6.2.8 Security Testing

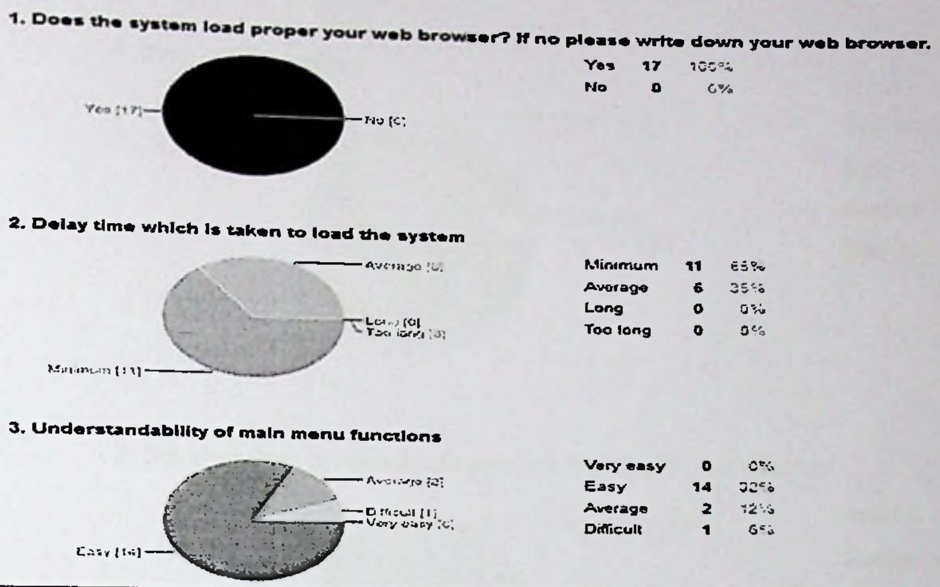
Under security testing unauthorized user login, password of user account changing by unauthorized person, denial of service attack to individual modules, fake script running in the system were checked. Secure data transfer (especially sensitive data) between client and server computer was also tested. And bugs of web browser software were tested.

6.3 Test Cases

Test cases were used to test the MOTIS. We can satisfy a function or module work properly if expected result of a test case is same as actual result. To perform a test case first we select a field or fields of a module, then different values should be entered as input. After that we should compare expected and actual result.

6.4 User Evaluation

Qualities of non functional requirements of the system were tested. That were web browser stability, site loading delay time, understandability of main menu functions, user friendliness of back ground color, font readability, accuracy, navigating pages and stable of the system. The system was accessed by users in seven user groups. There were administrators, registrar, and head of department, lecturer, technical officer, office staff and students groups. Pre designed multiple choice questioner was given each user. The questioner is available Figure D.1 and Figure D.2 (refer Appendix D). Then user feedback was recorded. Recorded user feedback is available Figure D.3 (refer Appendix D). Summary of user feedback is presented by Figure 6.1, Figure 6.2 and Figure 6.3.



1. Does the system load proper your web browser? If no please write down your web browser.

Yes	17	100%
No	0	0%

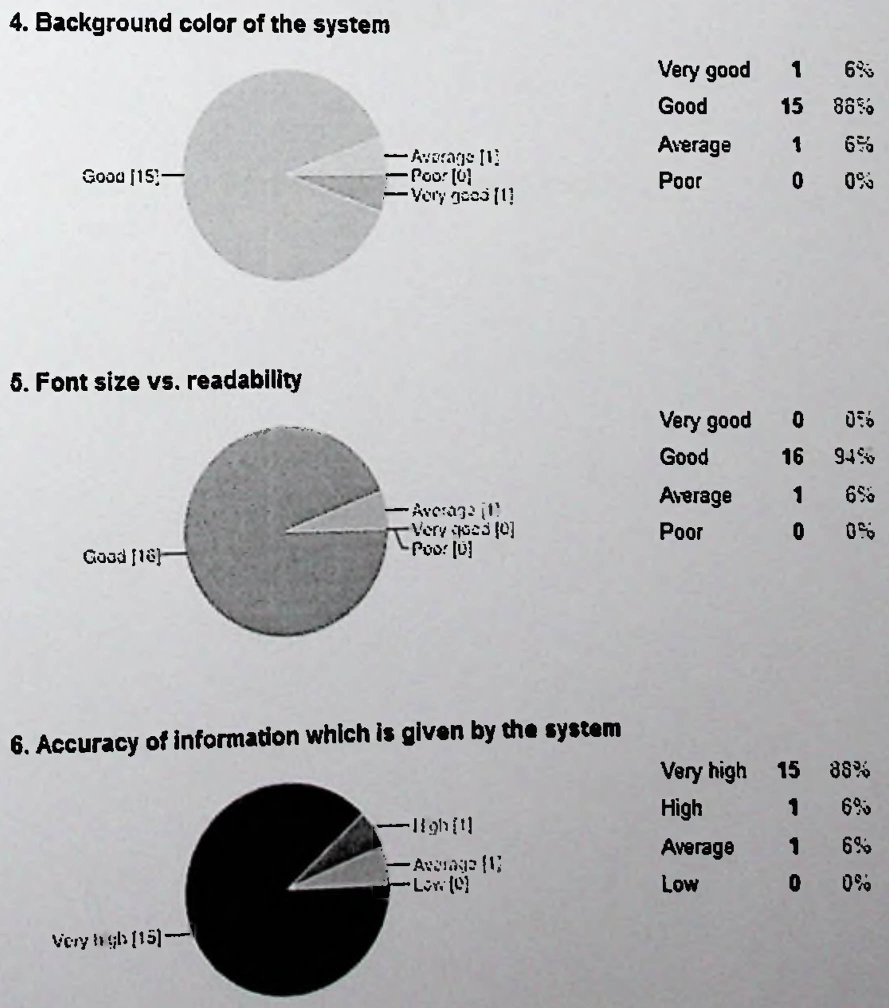
2. Delay time which is taken to load the system

Minimum	11	65%
Average	6	35%
Long	0	0%
Too long	0	0%

3. Understandability of main menu functions

Very easy	0	0%
Easy	14	82%
Average	2	12%
Difficult	1	6%

Figure 6.1 – User Evaluation Summary Part 1



4. Background color of the system

Very good	1	6%
Good	15	88%
Average	1	6%
Poor	0	0%

5. Font size vs. readability

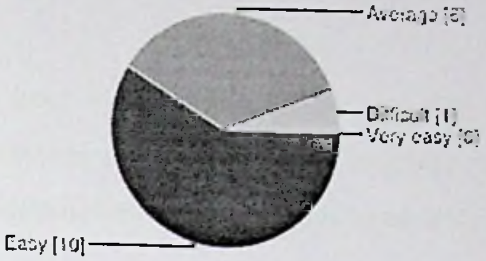
Very good	0	0%
Good	16	94%
Average	1	6%
Poor	0	0%

6. Accuracy of information which is given by the system

Very high	15	88%
High	1	6%
Average	1	6%
Low	0	0%

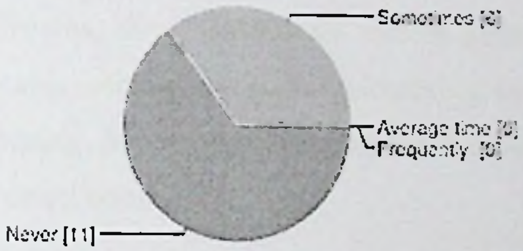
Figure 6.2 – User Evaluation Summary Part 2

7. Page navigation



Very easy	**0**	0%
Easy	**10**	59%
Average	**6**	35%
Difficult	**1**	6%

8. Service unavailable while you are login with the system



Never	**11**	65%
Sometimes	**6**	35%
Average time	**0**	0%
Frequently	**0**	0%

Figure 6.3 – User Evaluation Summary Part 3

Chapter 7

Discussion

This system is web base system plan to test against client requirements and expectations. To detect errors, content tests, interface testing, navigation tests, component testing, configuration testing, performance tests and security test will do against the system.

7.1 Content Testing

Content Testing is executing check whether the system meets user level goals. The content follow CRAP also spellings and grammar checking, accuracy of the system is check under content testing. Structure of the college and its activities will display to end user are check to detect errors.

7.2 Interface Testing

Unit test is use to check interface individually. Appearance of text, images, font colors and styles, output hope to check. Empty field detection, password matching, e mail format checking,

7.3 Navigation Testing

Hyper links are testing under navigation testing.

7.4 Component Testing

Functioning errors of individual modules and database are testing.

7.5 Configuration Testing

Configuration settings of server computer and client computers are testing.

Server Computer: Service status of MySql server and Apache web server are testing.

Client Computers: Existence and loading of Mozilla Firefox web browser is testing.

Testing Servers: WAPM server setting and MySql , Apache web server status are checking.

7.6 Usability Testing

Usability testing is the system testing for how easy user to access the system (easy navigation, readability of text, appearance of graphics, Display layout, etc.)

7.7 Performance Testing

Web server computer performance LAN and client computer performance are mainly focus.

7.8 Security Testing

Under security testing unauthorized user login, password of user account changing by unauthorized person, bugs of web browser software.

7.9 Test Cases

Test cases were used to test the MOTIS.

Non functional requirements of the system are test by using user evaluation. The system access users are administrator, President, vice Presidents, lecturer, technical officer, office staff and students. Using multiple choice questioner to user, gathering evaluation about web browser stability, site loading delay time, understandability of main menu functions, user friendliness of back ground color, font readability, accuracy, navigating pages and stable of the system.

Main objective of developing the MOTIS is giving online solution for students' management of the NCOE and minimizing paper based activities.

Hope to meet bellow user goals successfully by using the MOTIS

- Maintain student information.
- Maintain course unit information.
- Course unit registration by students.
- Assigned lectures for course unit.
- Generate attendance sheet and enter student attendance for each lecture.
- Students can check their eligibility for final exam.
- Issue end examination admission to each student.
- Enter student result for each subject.
- Maintain pass out student information.
- Maintain user groups with different security levels.

Compare with other education system MOTIS has not offered any service for lectures. If lecture services will be add they can check student attendance for their course units, student result, etc.

The system can be Accesses College' premises network only. Hope to link the system with the internet.

We are going to design message sending mechanism by using e mails like other system.

Chapter 8

Evaluation

8.1 Introduction

MOTIS will be evaluated in this chapter. Outcome results of the entire project will be evaluated with user goals and objectives. Then difficulties of the MOTIS will be discussed. Finally future developments or enhancements will be discussed.

8.2 Achieved Objectives

8.2.1 Maintain Student Information

Student information entering and student account creating can be done same time. After that any authorized person at the college can access that student information. Traditional paper base system can be replaced. And fast searching ability is increased.

8.2.2 Maintain Course Unit Information

New course unit information can be entered to the MOTIS. Students can select and register course units online. Also department heads can assign lectures for course units. Student or department head can do above task just login with the system. But before that student should submit course registration form to the office and department heads should assign lectures manually via excel sheet, and it should be passed to office.

8.2.3 Online Attendance Sheet

Before implementing the MOTIS, student attendance for lectures were entered manually. Technical officer was received hard copy of attendance sheet or student's signatures. Then he entered it into excel sheet. After every lecture series percentage attendance of each student was calculated. But so many errors were happened that process. The MOTIS can give solution. Using the MOTIS, technical officer can enter student attendance online for any lecture series. While attendance is being entered, the system is able to give percentage attendance and subject eligibility of students. The MOTIS Percentage attendance and eligibility status are high accurate than traditional calculation mechanism.

8.2.4 Online Exam Admission

Students can obtain their final examination admission card via the System. Now they do not wait some time during their study leave period to obtain it as traditional paper based one. Not eligible subjects are highlighted by the system. Before that long time period is taken to find eligible subject of a particular student. And also that process calculation mistakes might be happened. While principal (academic) is giving approval to generate admission cards, student can obtain it.

8.2.5 Student Result

The system allows entering student results online. Typically office staff members entered student grades into excel sheet. Typing mistakes might be happened. And it took long time to publish that result to students. The MOTIS had been designed minimizing typing errors. While an office user is going to enter result via the MOTIS, he will select correct grade for a particular student rather than type it. Students can view their exam result online without waiting long time.

8.2.6 Pass out Student Information

The System maintains pass out student's information. Authorized user can find pass out student information in batch wise also. Contact information and other information are available. Typically pass out student information were stored in paper based. It was difficult to find a particular student.

8.2.7 Security

Only authorized users can access the system. Before login to the system, your user account should be created. Student user accounts are created from office of the College. Other user accounts are created by System administrator of the college. The system maintains different user groups. Each user can access the system under some restrictions. That group concept helps to maintain security levels between different user groups. Different user groups can access different set of functions.

8.3 Difficulties in the MOTIS

Students should do activities with the system in correct time period. Because student related acidities are processed as a chain. For an example, without registering for course units, student cannot get any further service from the system such as exam admission, result, etc. . Though reducing paper based work some users (Not students) do not like to move new online environment. They like to stay with older environment. The system had been designed mainly by focusing student activities of the college not other users too much. Some users are expecting the system can full fill any requirement they think such as inventory controlling, laboratory management, etc.

Chapter 9

Conclusions

9.1 Introduction

This chapter we will discuss how much success the MOTIS objectives with user goals and future developments.

9.2 Objectives of the MOTIS

Main objective of developing the **MOTIS** is giving online solution for student management of the NCOE. Isolated activities such as course unit registration, issuing admission, etc are linked together and established relationships among them. Another objective is minimizing paper based activities.

9.3 Activities of the MOTIS

The MOTIS can meet bellow user goals successfully.

- Maintain student information.
- Maintain course unit information.
- Course unit registration by students.
- Assigned lectures for course unit.
- Generate attendance sheet and enter student attendance for each lecture.
- Students can check their eligibility for final exam for each subject based on their attendance.
- Issue final year examination admission to each student.
- Enter student result for each subject.
- Maintain pass out student information.
- Maintain user groups with different security levels.

9.4 Future Developments

9.4.1 Added Survives for Lectures

Current MOTIS has not offered more service for lectures. Future lecture services will be added. After that they can check student attendance for their course units, student result upload, etc.

9.4.2 Login to MOTIS any where

Now the system can be accessed inside the College network. We are going to link the system with the internet. After that student can access anywhere if they have a internet connection.

9.4.3 Configuring E Mail System

We are going to design message sending mechanism by using e mails. Course unit registration starting and ending date, admission card issuing date and other useful information can be sent to students quickly and easy.

9.4.4 Connecting with Other Systems

The MOTIS will be connected with other student service systems. For example the system can be connected with Student Hostels Management System. After that students can access both systems in one login interface. Also functional level relationships can be built between two systems. Then both systems will work consistency.

Reference

[1] SchoolTool. [Online]. http://www.schooltool.org/

[2] TechLearning. [Online]. http://www.techlearning.com/article/48288

[3] Web portal system for student information system a Java project. [Online]. http://1000projects.org/student-information-system-a-java-project.html

[4] Cook. (2005, Dec.) ClaSS. [Online]. http://lwn.net/Articles/165066/

[5] J. Julian. Student System Project. [Online]. http://ssp.unimelb.edu.au/about

[6] E. Oz, *Management Information Systems.*, 2008

[7] P. Shea and S. Armitage. Guide line for creating student services online. [Online]. http://wcet.wiche.edu/wcet/docs/beyond/overview.pdf

[8] R.Stair and G. Reynolds, *Principles of Information Systems.*, 2011

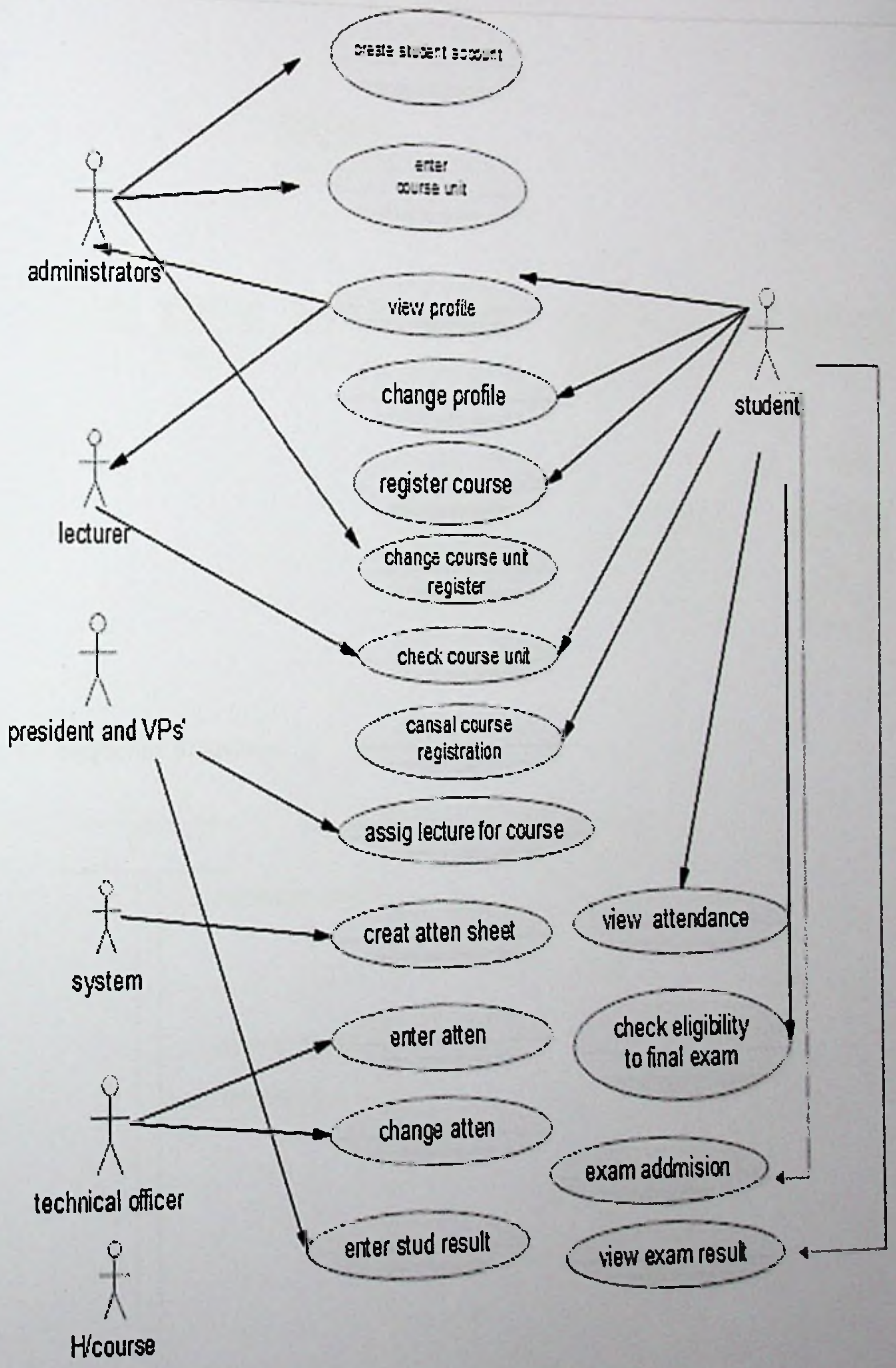
[9] C. Jones, *Software Engineering Best Practices.*, 2010.

[10] G. Booch, et al., *Object-Oriented Analysis and Design with Applications.*

[11] D.Pilone and N. Pitman, *UML 2.0 in a Nutshell.* O'Reilly Media, 2005.

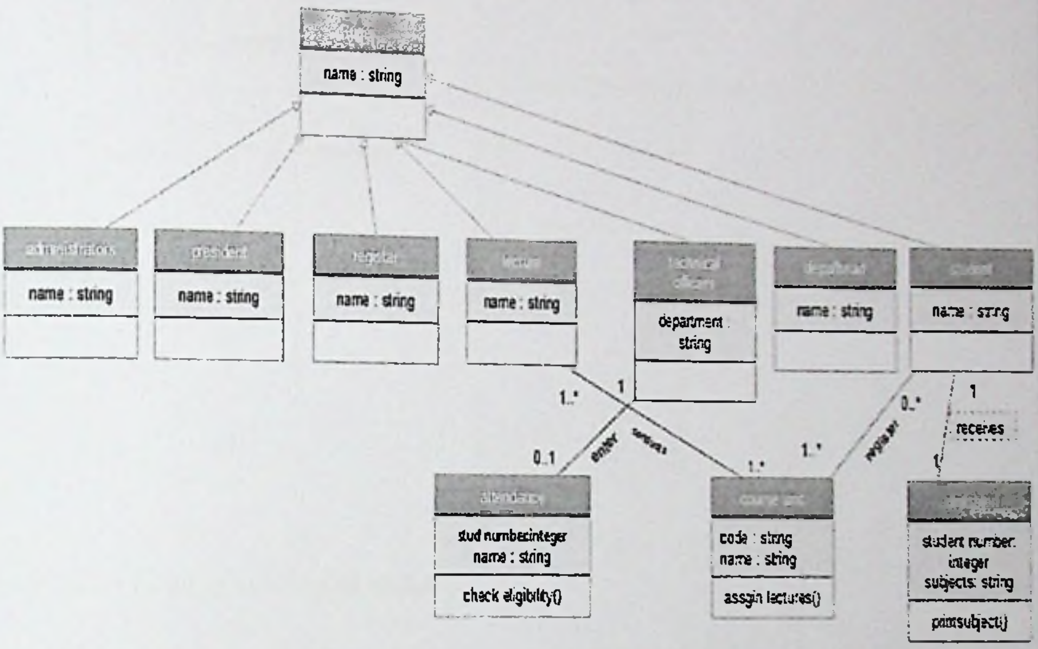
[12] K. C. Laudon, *Management Information Systems.*, 2011

Appendix A – Design Documentation



Use case diagram

A.2 Class diagram



Class diagram

A.3 Sequence diagrams

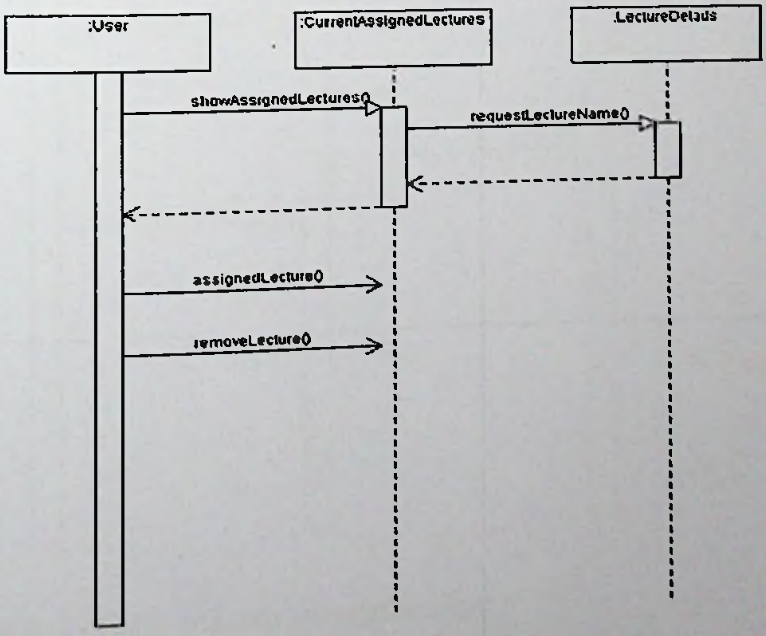


Figure A.1 - Assign or remove lecture

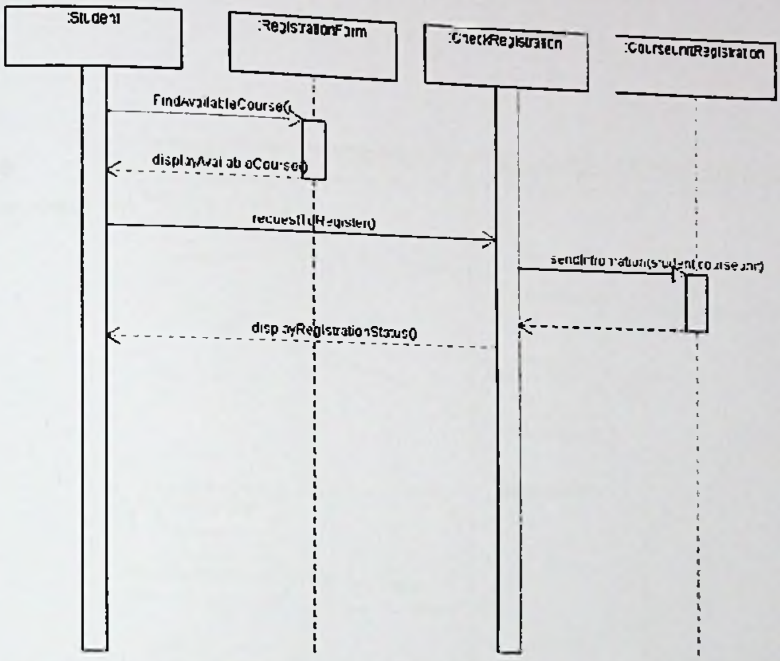


Figure A.2 - Course unit registration

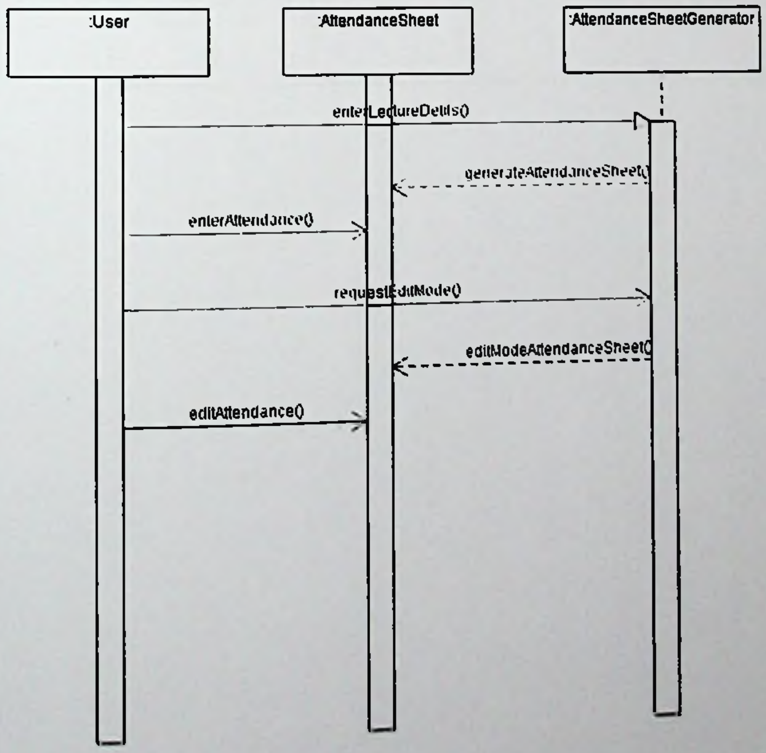
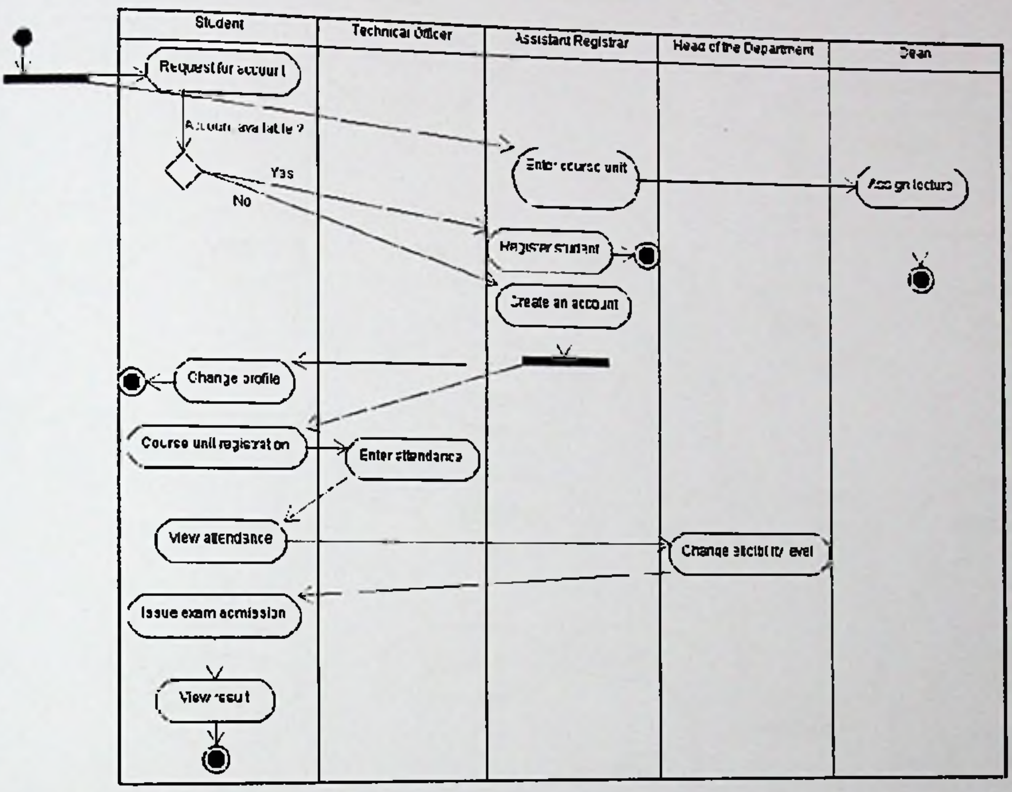
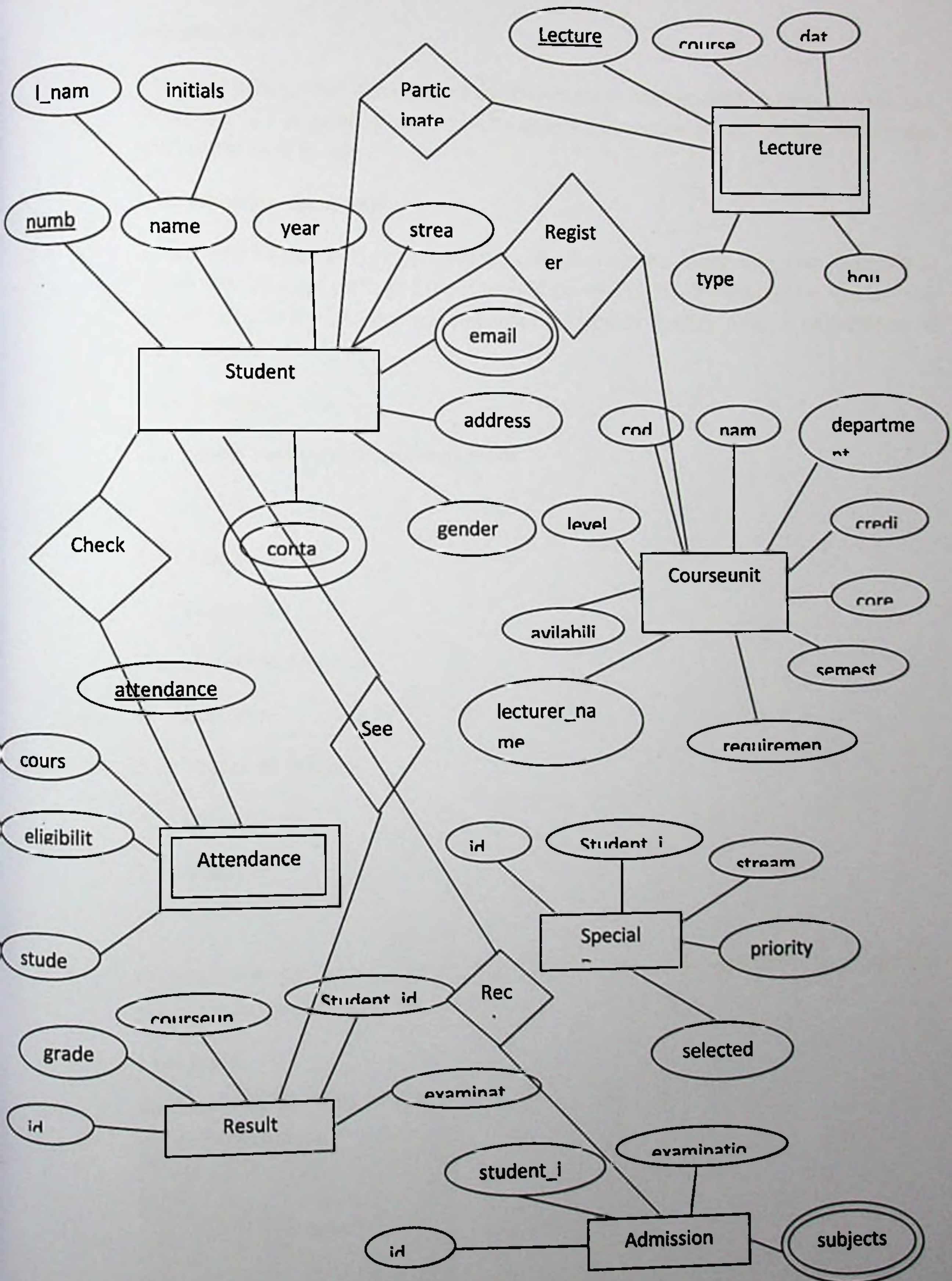


Figure A.3 - Enter and edit attendance sheet

A.4 Activity diagram



A.5 ER diagram



Appendix B – User Manual

Introduction

This user manual had been written for Information Management System of Teachers (MOTIS). It will guide users to how to access the system. It will explain application level of the system only.

How to access the system

System can be accessed only inside the college complex. Make sure your computer is connected with the College Local Area Network (LAN). To access the system you should be enter a user name and password provide by System Analyst or president of the college.

User groups

The system maintains below user groups.

1 Administrators

2 President

3 Department Heads

4 Lectures

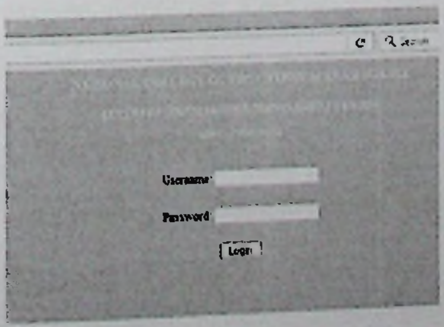
5 Technical Officers

6 Office Users

7 Students

Different user groups have different privileges. That means different user groups can get different services from the system.

User login



You should enter your user name and password and then click on Login button. If your information is correct then your profile page will be loaded. If there is any error about your user name and password, the system will display it.

Profile page

There are two types of profiles. There are student profile and other user profiles. Student profile can be edited but other user profile cannot be edited. Number of services and type of services which are available in a menu of the profile is different from different group of user profiles.

Administrative Profile

User can not edit his/her profile. User can get bellow services,

Enter Student Information

Enter New Courseunit

Assigned Lectures

Generate Attendance Sheet

Enter Student Attendance

View Eligible Subjects (Student service)

Process Admission Card

Enter Result

Set Pass out Students

Create New User Account

Find Result (Student Service)

View Pass out Students

president Profile

User can not edit his/her profile. User can get bellow services,

Enter Student Information

Process Admission Card

Set Pass out Students

Office User Profile

User can not edit his/her profile. User can get bellow services,

Enter New Courseunit

Enter Result

Department Head Profile

User can not edit his/her profile. But user can get bellow services,

Assigned Lectures

Technical Officer Profile

User cannot edit his/her profile. But user can get bellow services,

Generate Attendance Sheet

Enter Student Attendance

Student Profile

Student can change his/her profile.

Also student can get bellow services,

Course Registration

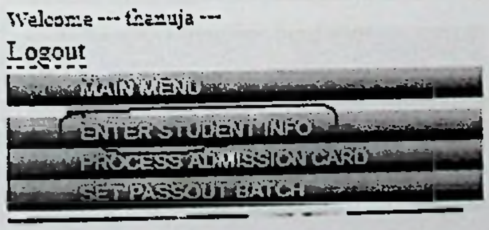
View Eligible Subjects

Obtain Examination Admission Card

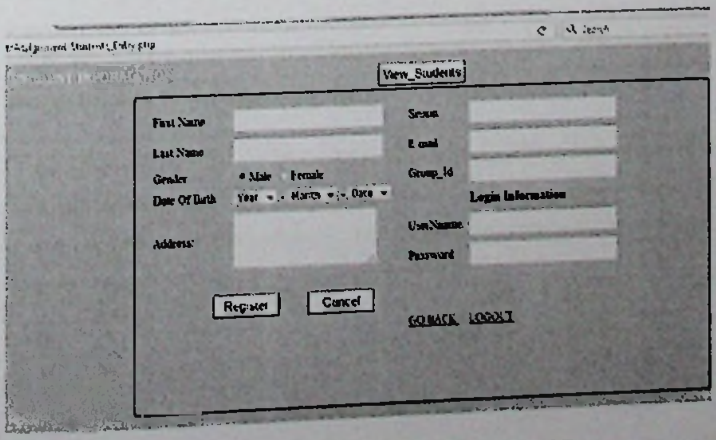
Find Result

B.1 How to enter student information

Login to the system as Administrative or president or Office user.



Then click on Enter Student Info service which is available in the menu.



Fill all fields of above form. Then you should click on Enter Record button to complete the task.

B.2 How to change student profile

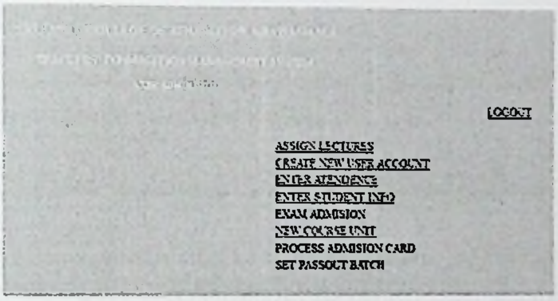
First login to the system as a Student user.

Then click on Edit Profile which is available bottom of the student profile page. Then you can edit your information. Only your postal address, e mail address can be edited. Also your password can be changed.

Your profile changes will be applied after clicking on Save Changes button.

B.3 How to enter new course unit

Login to the system as Administrative or Office user.



Then click on New Course unit service from the menu.

Fill form and click on Enter Course unit to enter new course unit.

B.4 How to Assigned Lectures

You should login as a Administrative or Department head user.Then you should select Assign Lectures service from the menu.

First you should select year. Then course unit is selected from available course unit list. A lecturer who will conduct the course unit is selected from lecture list. Finally you should click on Assigned button.

B.5 How to Enter Student Attendance

You should login as an Administrative or Technical Officer user to do that task. You should click on Enter Attendance service in the menu.

You should select Year and Department. Then you should select course unit from Course Unit menu. Then you have to set lecture date, number of lecture hours, starting time and lecture type. After that you should click on Generate Attendance Sheet button to obtain attendance sheet.

You should click on check box in front of student number who participated for the lecture. After that this form should be submitted by using Submit button. You cannot edit the sheet again once you submitted it.

B.6 How to Check Subject Eligibility

This service can be access only student users. First student should log into the system. Then student will select Eligible Subjects service form the menu, Note: Subject eligibility and attendance percentage will be calculated up to current date.

Then you should select year you are going to check subject eligibility from menu , After that you can see your subjects eligible status.

B.7 How to Enter Student Result

You should login as an Office user or an Administrative user. Select Enter Result service from the menu.

Then select Course unit you are going to enter student result and year the exam was held. After that click on Go button.

B.8 How to maintain Pass out Student's information

First you should login as an Assistant Registrar or an Administrative user. You should click on Set Pass out Batch in the menu.

Then select registration year of the batch. To proceed click on Set as Pass out button.

B.9 How to Create New User Account

First you should login as an Administrative user.Then you should select Create New User Account service from the menu.

User login name should be typed. Then you will select user group. Password and confirm the password should be done. Finally click on Create button.

B.10 How to View Pass out Students

You should login as an Administrative user. Then you will select View Passout Student service from the menu.

Then select registered year of the student(s) or batch. Then you can see pass out students for that registered year.

B.11 Edit Student Information

You should login as an administrative or registrar or office user. You select edit student information from main menu. Then you can direct enter student number or search student from registered year. After that you can edit selected student information. You should click on edit record button to save changes.

B.12 Edit Course unit

You should login as an administrative or office user. You select edit course unit from main menu. After editing you should click on edit button to save changes.

B.13 Edit Result

You should login as an administrative or office user. You will select edit result from main menu. Then you select course unit and exam year. Result sheet will be displayed in edit mode which has satisfied given information. Now you can edit result.

B. 14 Search Students

This facility can be accessed by administrator, lecturer, department head and assistant registrar users. You should select view student information from main menu. Then you enter student number or you can search student from registration year.

B.15 View Student Result

Lectures, department heads can view student exam results. You should select view student result from main menu. Then you enter course unit and exam year. Then you can view result.

Appendix C – Code Listings

C.1 Student Result Entering

This script enters student result into result table.

```
<?php

$id="";

$opr="";

if(isset($_GET['opr']))

        $opr=$_GET['opr'];

if(isset($_GET['rs_id']))

        $id=$_GET['rs_id'];

if(isset($_POST['btn_sub'])){

        $stu_name=$_POST['sudenttxt'];      $fa_name=$_POST['factxt'];

        $sub_name=$_POST['subjecttxt'];    $miderm=$_POST['midermtxt'];

        $final=$_POST['finaltxt'];

        $note=$_POST['notetxt'];

$sql_ins=mysql_query("INSERT INTO stu_score_tbl

        VALUES(NULL,      '$stu_name',   '$fa_name' ,   '$sub_name',   '$miderm',

        '$final',         '$note' )");

if($sql_ins==true)

        $msg="1 Row Inserted";

else    $msg="Insert Error:".mysql_error();

        }

if(isset($_POST['btn_upd'])){

        $stu_id=$_POST['sudenttxt'];         $faculties_id =$_POST['factxt'];

        $sub_id=$_POST['subjecttxt'];        $miderm=$_POST['midermtxt'];

        $final=$_POST['finaltxt'];    $note=$_POST['notetxt'];

        $sql_update=mysql_query("UPDATE stu_score_tbl SET

        stu_id='$stu_id' ,       faculties_id='$faculties_id' ,

        sub_id='$sub_id' ,       miderm='$miderm' ,   final='$final' ,
        note='$note'    WHERE ss_id=$id    ");
if($sql_update==true)
        header("location:?tag=view_scores");
else
        $msg="Update Fail!...";}>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title></title>
<link rel="stylesheet" type="text/css" href="css/style_entry.css" />
</head> <body><?php
if($opr=="upd"){
$sql_upd=mysql_query("SELECT * FROM stu_score_tbl WHERE ss_id=$id");
$rs_upd=mysql_fetch_array($sql_upd);?>
        <div id="top_style">    <div id="top_style_text">
    Scores Update       </div><!-- end of top_style_text-->
     <div id="top_style_button">
                <form method="post">
                <a href="?tag=view_scores"><input type="button" name="btn_view"
value="Back" id="button_view" style="width:70px;"  /></a>
     </form>    </div><!-- end of top_style_button-->
</div><!-- end of top_style-->
<div id="style_informations">
<form method="post">        <div>
  <table border="0" cellpadding="5" cellspacing="0">
<tr>            <td>Students's Name</td>            <td>
<select name="sudenttxt" id="textbox"> <option>---- Students's Name -----</option>
```

```
  <?php   $student_name=mysql_query("SELECT * FROM stu_tbl");

        while($row=mysql_fetch_array($student_name)){
         if($row['stu_id']==$rs_upd['stu_id'])
        $iselect="selected";                         else
        $iselect="";    ?>

 <option value="<?php echo $row['stu_id'];?>" <?php echo $iselect ;?> > <?php echo
$row['f_name']; echo" "; echo $row['l_name'];?> </option>

<?php  }                    ?>              </select>

    </td>       </tr>      <tr><td>Facuties's Name</td>    <td>

<select name="factxt" id="textbox"> <option>---- Facuries's Name   ------</option>

  <?php

                    $fac_name=mysql_query("SELECT * FROM facuties_tbl");
while($row=mysql_fetch_array($fac_name)){
if($row['faculties_id']==$rs_upd['faculties_id'])
        $iselect="selected";
        else
        $iselect="";                  ?>

<option value="<?php echo $row['faculties_id'];?>" <?php echo $iselect ;?> > <?php
echo $row['faculties_name'];?> </option>

                    <?php   }  ?>  </select>   </td>  </tr>    <tr>

                <td>Subjects's Name</td>           <td>

                <select name="subjecttxt" id="textbox">

                <option>------------ Sujects -----------</option>

                  <?php

                    $subject=mysql_query("SELECT * FROM sub_tbl");
          while($row=mysql_fetch_array($subject)){
          if($row['sub_id']==$rs_upd['sub_id'])
        $iselect="selected";
        else
        $iselect="";           ?>

 <option value="<?php echo $row['sub_id'];?>" <?php echo $iselect ;?> > <?php
echo $row['sub_name'];?> </option>

   <?php        }          ?>

 </select>    </td>      </tr>     <tr>     <td>Assignment</td>  <td>

<input type="text" name="midermtxt" id="textbox" value="<?php echo
$rs_upd['miderm'];?> "/>    </td>  </tr>      <tr>  <td>Exam</td>      <td>
```

```
<input type="text" name="finaltxt"  id="textbox" value="<?php echo
$rs_upd['final'];?>" />     </td>          </tr>                    <tr>

d>Note</td>              <td>
                                           <textarea name="notetxt" cols="23"
rows="3"><?php echo $rs_upd['note'];?></textarea>

          </td>          </tr>       <tr>            <td colspan="2">

               <input type="reset" value="Cancel" id="button-in"/>

<input type="submit" name="btn_upd" value="Update" id="button-in" title="Update"
/>          </td>          </tr>                    </table>  </div>   </form>

</div><!-- end of style_informatios --><?php         }else{?>

   <div id="top_style">   <div id="top_style_text">         Scores Entry

        </div><!-- end of top_style_text-->    <div id="top_style_button">

               <form method="post">

<a href="?tag=view_scores"><input type="button" name="btn_view"
value="View_Scores" id="button_view" style="width:120px;"  /></a>

          </form>

 </div><!-- end of top_style_button-->  </div><!-- end of top_style-->

<div id="style_informations"> <form method="post">       <div>

        <table border="0" cellpadding="5" cellspacing="0">  <tr>

               <td>Students's Name</td>              <td>

               <select name="sudenttxt" id="textbox">

               <option>---- Students's Name -----</option>

                 <?php

$student_name=mysql_query("SELECT * FROM stu_tbl");
        while($row=mysql_fetch_array($student_name)){
        ?>

  <option value="<?php echo $row['stu_id'];?>"> <?php echo $row['f_name']; echo"
"; echo $row['l_name'];?> </option>

<?php  }                   ?>           </select>          </td>         </tr>

        <tr>    <td>Facuties's Name</td>             <td>

               <select name="factxt" id="textbox">

               <option>---- Facuries's Name  ------</option>

                 <?php
  $fac_name=mysql_query("SELECT * FROM facuties_tbl");
while($row=mysql_fetch_array($fac_name)){        ?>

<option value="<?php echo $row['faculties_id'];?>"> <?php echo
$row['faculties_name'];?> </option>

 <?php   }            ?>

    </select>   </td></tr> <tr>        <td>Subjects's Name</td>            <td>

<select name="subjecttxt" id="textbox">

         <option>------------ Sujects -----------</option>

        <?php

   $subject=mysql_query("SELECT * FROM sub_tbl");
while($row=mysql_fetch_array($subject)){          ?>

 <option value="<?php echo $row['sub_id'];?>"> <?php echo $row['sub_name'];?>
</option>                   <?php        }              ?>

             </select>           </td>        </tr>        <tr>

                <td> Assignment</td>                 <td>

                <input type="text" name="midermtxt" id="textbox" />

            </td>        </tr>        <tr>              <td>Exam</td>

            <td>

                <input type="text" name="finaltxt"  id="textbox" />

            </td>        </tr>        <tr>     <td>Note</td>

            <td>

                <textarea name="notetxt" cols="23" rows="3"></textarea>

            </td>        </tr>             <tr>

            <td colspan="2">

                <input type="reset" value="Cancel" id="button-in"/>

<input type="submit" name="btn_sub" value="Add Now" id="button-in"  />

            </td>        </tr>        </table>

  </div>   </form>

</div><!-- end of style_informatios -->

<?php}?>

</body></html>
```

C.2 Pass out Student's Information

```
<?php

session_start();

$year=$_POST["year"];

 include '../connection/config.php';

$con=mysql_connect($host,$user,$pass) or die ("Database server connection failure
");

mysql_select_db($db) or die ("Database failure ");

$query="SELECT st_number from student WHERE year=$year";

$result=mysql_query($query);

while($row=mysql_fetch_array($result))      {

        $st_number=$row['st_number'];

        $query2="INSERT INTO passout_student VALUES ('$st_number',$year)";

        mysql_query($query2);        }

        mysql_close($con);

echo '<link rel="stylesheet" type="text/css"
href="http://localhost/imst/style_sheet/profile.css" />';

echo '<p class="normal">Passout students have been set successfully</p>';

include 'index.php';

?>
```

C.3 Creating New User Form

```
<?php


        $msg="";        $opr="";        $id="";

        if(isset($_GET['opr']))         $opr=$_GET['opr'];

        if(isset($_GET['rs_id']))       $id=$_GET['rs_id'];

        if(isset($_POST['btn_sub'])){ $f_name=$_POST['fnametxt'];

        $l_name=$_POST['lnametxt'];         $gender=$_POST['genderrdo'];
        $dob=$_POST['yy']."/".$_POST['mm']."/".$_POST['dd'];

        $pob=$_POST['pobtxt'];      $addr=$_POST['addrtxt'];

        $degree=$_POST['degree'];  $salary=$_POST['slarytxt'];

        $married=$_POST['marriedrdo'];     $phone=$_POST['phonetxt'];

        $mail=$_POST['emailtxt'];   $note=$_POST['notetxt'];


$sql_ins=mysql_query("INSERT INTO teacher_tbl

VALUES(NULL,      '$f_name', $l_name' ,  '$gender',       '$dob',

        $pob',  '$addr','$degree','$salary' ,'$married', '$phone','$mail',
        '$note'  )");if($sql_ins==true)

        $msg="1 Row Inserted"; else

        $msg="Insert Error:".mysql_error();}

//------------------uodate data----------

if(isset($_POST['btn_upd'])){

        $f_name=$_POST['fnametxt'];        $l_name=$_POST['lnametxt'];

        $gender=$_POST['genderrdo'];
        $dob=$_POST['yy']."/".$_POST['mm']."/".$_POST['dd'];

        $pob=$_POST['pobtxt'];      $addr=$_POST['addrtxt'];

        $degree=$_POST['degree'];  $salary=$_POST['slarytxt'];

        $married=$_POST['marriedrdo'];     $phone=$_POST['phonetxt'];

        $mail=$_POST['emailtxt'];   $note=$_POST['notetxt'];

        $sql_update=mysql_query("UPDATE teacher_tbl SET
        f_name='$f_name' ,   l_name='$l_name' ,

gender='$gender' ,      dob='$dob' ,   pob='$pob' ,   address='$addr' ,degree='$degree'
,salary='$salary' ,       married='$married' ,  phone='$phone' ,       email='$mail' ,

note='$note'    WHERE teacher_id=$id       ");

if($sql_update==true)

        header("location:?tag=view_teachers");
```

```
else     $msg="Update Fail!...";

}?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">

<head>

<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />

<link rel="stylesheet" type="text/css" href="css/style_entry.css" />

<title>NCOE MAHARAGAMA</title></head>

<body> <?php

if($opr=="upd"){

$sql_upd=mysql_query("SELECT * FROM teacher_tbl WHERE teacher_id=$id");

$rs_upd=mysql_fetch_array($sql_upd);

list($y,$m,$d)=explode('-',$rs_upd['dob']);?>

<div id="top_style">

        <div id="top_style_text">

        Lecturer

        Update</div>

        <!-- end of top_style_text-->

       <div id="top_style_button">

 <form method="post">

<a href="?tag=view_teachers"><input type="button" name="btn_back" value="Back"
id="button_view" style="width:70px;"  /></a>

</form>

       </div><!-- end of top_style_button-->

</div><!-- end of top_style-->

<!-- for form Upadte-->

<div id="style_informations">        <form method="post">
```

```
      <div>            <table border="0" cellpadding="5" cellspacing="0">

            <tr>            <td>First Name</td>            <td>

<input type="text" name="fnametxt" id="textbox" value="<?php echo
$rs_upd['f_name'];?>" />       </td>       </tr>       <tr>

            <td>Last Name</td>            <td>

<input type="text" name="lnametxt" id="textbox"  value="<?php echo
$rs_upd['f_name'];?>"/>      </td>   </tr>   <tr>         <td>Gender</td>

          <td>

            <input type="radio" name="genderrdo" value="Male"<?php
if($rs_upd['gender']=="Male") echo "checked";?>  />Male

            <input type="radio" name="genderrdo" value="Female"<?php
if($rs_upd['gender']=="Female") echo "checked";?> />Female

          </td>         </tr>          <tr><td>Date Of Birth:</td>

        <td>               <select name="yy" >

            <option>years</option>

            <?php  $sel="";
      for($i=1957;$i<=2017;$i++){
                  if($i==$y){
            $sel="selected='selected'";}
            else                  $sel="";
            echo"<option value='$i' $sel>$i </option>";
                  }?>

            </select>

          <select name="mm">          <option>Month</option>

      <?php                $sel="";

$mm=array("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","NOv","
Dec");                $i=0;

               foreach($mm as $mon){

                  $i++;
      if($i==$m){
      $sel=$sel="selected='selected'";}
                  else
            $sel="";
```

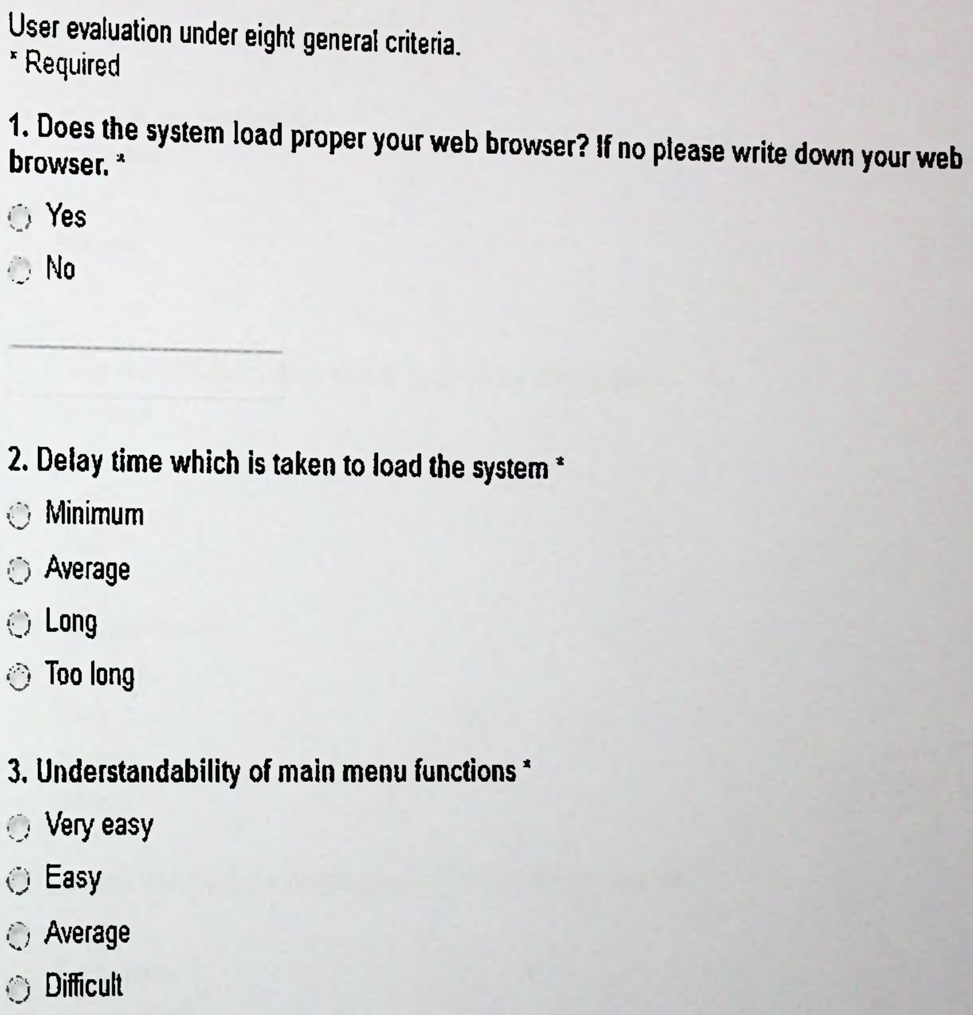
```
                echo"<option value='$i' $sel> $mon</option>";                        }
            ?>              </select>      -          <select name="dd">
            <option>Date</option>
    <?php                                           $sel="";
            for($i=1;$i<=31;$i++){
    if($i==$d)
    $sel="selected='selected'";                      else
                    $sel="";                         ?>
            <option value="<?php echo $i ;?>"<?php echo $sel ;?> >
            <?php                if($i<10)
              echo"0"."$i" ;
            else
              echo"$i";                   ?>
    </option>                       <?php                                          }?>
                                    </select>
      </td>       </tr>           <tr>
            <td>Place Of Birth</td>              <td>
            <input type="text"  name="pobtxt" id="textbox" value=" <?php echo
$rs_upd['pob']; ?>"/>
        </td>           </tr>           <tr>
            <td>Address</td>
            <td>
<textarea name="addrtxt" cols="23" rows="3"><?php echo
$rs_upd['address'];?></textarea>
 </td>          </tr>         <tr>              <td colspan="2">
            <input type="reset" value="Cancel" id="button-in"/>
    <input type="submit" name="btn_upd" value="Update" id="button-in"  />
      </td>        </tr>          </table >  </div>       <div>
    <table border="0" cellpadding="5" cellspacing="0">
        <tr>              <td>Degree</td>              <td>
```

```
            <select name="degree" id="textbox" >

              <?php

                $mm=array("Bachelor","Master","P.HD");

                $i=0;

                foreach($mm as $mon){

                  $i++;
            if($mon==$rs_upd['degree'])
                                    $iselect="selected";
                                          else
                                          $iselect="";

      echo"<option value='$mon' $iselect> $mon</option>";

                }                 ?>              </select>

          </td>          </tr>                    <tr>

            <td>Salary</td>              <td>

 input type="text" name="slarytxt" id="textbox" value="<?php echo
$rs_upd['salary'];?>" />           </td>          </tr>

            <tr>          <td>Married</td>           <td>

            <input type="radio" name="marriedrdo" value="Yes"<?php
if($rs_upd['married']=="Yes") echo "checked";?>/>Yes

            <input type="radio" name="marriedrdo" value="No"<?php
if($rs_upd['married']=="No") echo "checked";?> />No

          </td>          </tr>

        <tr>

                  <td>Phone</td>          <td>

            <input type="text"  name="phonetxt" id="textbox" value="<?php echo
$rs_upd['phone'];?>" />           </td>          </tr>

        <tr>                        <td>E-mail</td>

          <td>          <input type="text" name="emailtxt" id="textbox"
value="<?php echo $rs_upd['email'];?>" />

          </td>          </tr>          <tr>          <td>Note</td>
```

```
                <td>

                    <textarea name="notetxt" cols="23" rows="3"><?php echo
$rs_upd['note'];?></textarea>

                </td>           </tr>       </table>  </div>    </form>

</div><!-- end of style_informatios -->

<?php  }else{?>

<div id="top_style">

    <div id="top_style_text">

    Lecturers Entry

    </div>

    <!-- end of top_style_text-->

    <div id="top_style_button">

                <form method="post">

                <a href="?tag=view_teachers"><input type="button"
name="btn_view" title="View Teachers" value="View Teachers" id="button_view"
style="width:120px;"  /></a>

                </form>

    </div><!-- end of top_style_button-->

</div><!-- end of top_style-->

<!-- for form Upadte-->

<div id="style_informations">

      <form method="post">

      <div>

      <table border="0" cellpadding="5" cellspacing="0">

            <tr>          <td>First Name</td>          <td>

            <input type="text" name="fnametxt" id="textbox" />

          </td>         </tr>          <tr>

            <td>Last Name</td>
          <td>

            <input type="text" name="lnametxt" id="textbox" />

          </td>            </tr>            <tr>

            <td>Gender</td>

          <td>

<input type="radio" name="genderrdo" value="Male" checked="checked"/>Male

            <input type="radio" name="genderrdo" value="Female" />Female

          </td>          </tr>          <td>Date Of Birth</td>          <td>

            <select name="yy" style="height:25px;">

            <option>Year</option>

            <?php
      for($i=1957;$i<=2017;$i++){

      echo"<option value='$i'>$i</option>";
            }                    ?>

            </select>                -

          <select name="mm" style="height:25px;">

            <option>Month</option>
      <?php
$mm=array("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","NOv","
Dec");                    $i=0;

              foreach($mm as $mon){

                $i++;

                echo"<option value='$i'> $mon</option>";

              }            ?>

          </select>                -

          <select name="dd" style="height:25px;">

            <option>Date</option>
      <?php
            for($i=1;$i<=31;$i++){
                ?>

                <option value="<?php echo $i; ?>">

                <?php

                if($i<10)

                   echo"0".$i;

                else

                   echo"$i";                                                     ?>
                                          </option>
                <?php                                                  }?>
                </select>

            </td>          </tr>          <tr>                <td>Place Of Birth</td>

              <td>

                <input type="text"  name="pobtxt" id="textbox"/>

              </td>            </tr>        <tr>          <td>Address</td>

                <td>

                <textarea name="addrtxt" cols="23" rows="3"></textarea></td>

          </tr>          <tr>                <td colspan="2">

                <input type="reset" value="Cancel" id="button-in"/>

                <input type="submit" name="btn_sub" value="Register" id="button-
in"  />               </td>          </tr>

          </table >

    </div>

          <div>

          <table border="0" cellpadding="5" cellspacing="0">

              <tr>

                <td>Degree</td>

              <td>                    <select name="degree" id="textbox">

                  <option>------------ Select ------------</option>
                <?php

                  $mm=array("Bachelor","Master","P.HD");

                  $i=0;

                  foreach($mm as $mon){

                    $i++;
              echo"<option value='$mon'> $mon</option>";

                    /echo"<option value='$i'> $mon</option>";

                  }                    ?>

              </select>              </td>          </tr>

              <tr>            <td>Salary</td>           <td>

              <input type="text" name="slarytxt" id="textbox" />

            </td>          </tr>

              <tr>

              <td>Married</td>

            <td>

              <input type="radio" name="marriedrdo" value="Yes"
checked="checked"/>Yes

              <input type="radio" name="marriedrdo" value="No" />No

            </td>          </tr>

          <tr>                     <td>Phone</td>

            <td>            <input type="text"  name="phonetxt" id="textbox"/>

            </td>         </tr>               <tr> <td>E-mail</td>

            <td>

              <input type="text" name="emailtxt" id="textbox" />

            </td>         </tr>        <tr> <td>Note</td>

            <td>            <textarea name="notetxt" cols="23"
rows="3"></textarea>           </td>        </tr>

        </table>  </div>  </form>
```

```
</div><!-- end of style_informatios -->

<?php

}

?>

</body>

</html>
```

Appendix D – User Evaluation



User evaluation under eight general criteria.
* Required

1. Does the system load proper your web browser? If no please write down your web browser. *

○ Yes

○ No

2. Delay time which is taken to load the system *

○ Minimum

○ Average

○ Long

○ Too long

3. Understandability of main menu functions *

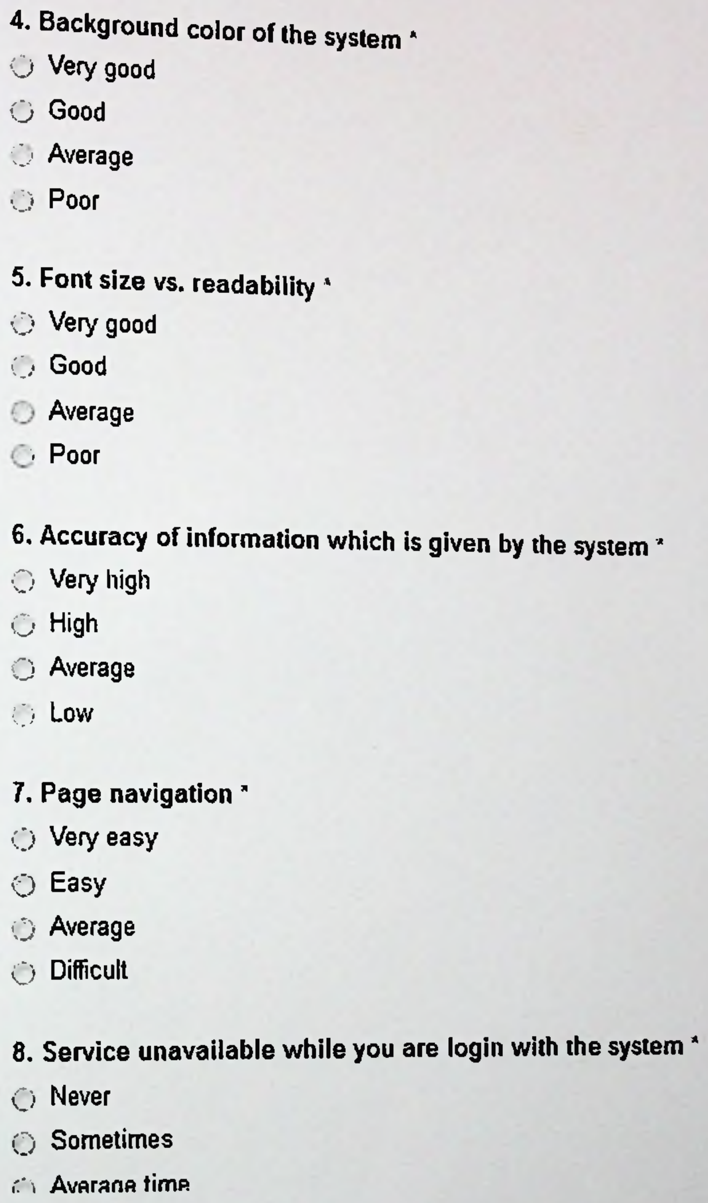
○ Very easy

○ Easy

○ Average

○ Difficult

Figure D.1 – Questioner



4. Background color of the system *

○ Very good

○ Good

○ Average

○ Poor

5. Font size vs. readability *

○ Very good

○ Good

○ Average

○ Poor

6. Accuracy of information which is given by the system *

○ Very high

○ High

○ Average

○ Low

7. Page navigation *

○ Very easy

○ Easy

○ Average

○ Difficult

8. Service unavailable while you are login with the system *

○ Never

○ Sometimes

○ Average time

Figure D.2 – Questioner