ROUTINE MANAGEMENT SYSTEM

(RMS)

BY

MANSURA BINTA MOZID
ID: 152-15-6100

This Report Presented in Partial Fulfillment of the Requirements for the Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

FAHAD FAISAL
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised By

MD. SADEKUR RAHMAN
Assistant Professor
Department of CSE
Daffodil International University

DAFFODIL INTERNATIONAL UNIVERSITY

DHAKA, BANGLADESH

DECEMBER 2019

APPROVAL

This Project **"Routine Management System"**, submitted by Mansura Binta Mozid, ID No: 152-15-6100 to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on date 5 December, 2019.

BOARD OF EXAMINERS

Dr. Syed Akhter Hossain — **Chairman**
Professor and Head
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Nazmun Nessa Moon — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

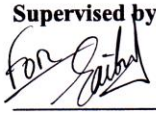
Gazi Zahirul Islam — **Internal Examiner**
Assistant Professor
Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Dr. Mohammad Shorif Uddin — **External Examiner**
Professor
Department of Computer Science and Engineering
Jahangirnagar University

DECLARATION

I hereby declare that, this project has been done by me under the supervision of **Fahad Faisal, Assistant Professor, and Department of CSE** Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Fahad Faisal
Assistant Professor
Department of CSE
Daffodil International University

Co-Supervised by:

Md. Sadekur Rahman
Assistant Professor
Department of CSE
Daffodil International University

Submitted by:

Mansura Binta Mozid

Mansura Binta Mozid
ID: 152-15-6100
Department of CSE
Daffodil International University

©Daffodil International University iii

ACKNOWLEDGEMENT

First, I express my heartiest thanks and gratefulness to almighty Allah for His divine blessing makes me possible to complete the final year project successfully.

I really grateful and wish my profound my indebtedness to **Fahad Faisal**, **Assistant Professor**, Department of CSE, Daffodil International University, Dhaka. Deep Knowledge & keen interest of my supervisor in the field of "*Routine Management System*" to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

I would like to express my heartiest gratitude to **Dr. Syed Akhter Hossain, Head, Department of CSE**, for his kind help to finish my project and also to other faculty member and the staff of CSE department of Daffodil International University.

I would like to thank my entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, I must acknowledge with due respect the constant support and patience of my parents.

ABSTRACT

"Routine Management System (RMS)" is a fast, simple yet modern looking desktop-based application to create routines, can easily search classes by course name, course-code, teacher's initial or room number. The main target of this project to reduce the complexity to book any classroom and reduce the waste of time. I am trying to implement all the features which will come in handy and create a user- friendly interface. The All the data and resources are stored in a database for better access and security. Here users can add, update and delete content data of teachers, Rooms and routine. Routine can be viewed from different perspectives. The Admin has all access to verified teacher's identity. The most important issue in my project is that teacher can directly book room when they need extra classes. Here, Teachers can be identified by their email address and phone number. User can access here as a teacher or a student. Then after selecting their campus, department and program they go the next step and then their expected routine will be showed. User also can search by their teacher's initial to see Teacher's routine day wise. They also get the facility to search by subject name or code. The aim of this project to make more-easier to find routine and teacher accession for booking room when extra class needed. This application is also helpful to recycle the wasted time with proper way.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Introduction

RMS is a software system that provides a user interface through a web browser. This is a fast, simple yet modern looking routine website designed for the admin, students and teachers of Daffodil International University. It currently fully supports the latest class routine of all Department. It is an open-source web-based system. It gives the best experiences for both teachers and students to find and book classroom.

1.2 Motivation

I examine and apply an information system for run class routine for our University which is automated. The system which I suggest, makes easy to implement the handbook system. It will be able to cope up with class schedule of the department by present-day computer. It will help to supply all the information of the class plan and position of departments in a quicker process successfully. My project may be used as a model for all department. So, we see there are many problems to book and to find classroom in our University. The necessity of Technology of the modern lifestyle is unbelievable. Technology has improved human lives significantly by providing efficiency. It has made easily able to done for us to access education, communication, medicine, transportation, sports etc. Our aim is student & teacher can find out their class routine easily. Student, Teacher & Admin can save their time.

1.3 Objectives

Object's world we live in. when we use these products, we see this existence in nature in human made attribute's, in business. We are categorized it, and combine it, manipulate it, combine it, and also can create it. The technology which we use is a new way of sense about problems using part by part module put in order to around client real-world idea.

I used here web-based applied science as PHP and MySQL for database to plan and create my project proposal, routine management system in client server environment. I make this project for

developing software manufacture. The details system with at least handbook interposition. I designed the Software to do informational tasks like Teachers, Student and the Admin.

I select this software to analyze and design and finally implement it.

- MySQL for Designing the database
- PHP
- Java Script, HTML, CSS, Java Script and jQuery for UI Design

1.4 Expected Outcome

The expectation of making this system teacher and student can find out their class room easily and able to booking classroom easily.

- Admin can check all user list
- Identify them
- Manage all users
- Teacher can watch class routine
- Teacher can book classroom
- Student can watch class routine

CHAPTER 2

BACKGROUND

2.1 Introduction

Technology is necessity of the modern lifestyle. Technology has improved human lives significantly by providing convenience and efficiency. It has made easily possible for us to access education, communication, medicine, transportation, sports etc. Our aim is student & teacher can find out their class routine easily. Student & teacher can save their time.

The following figure 2.1 shows Business Process Modeling.

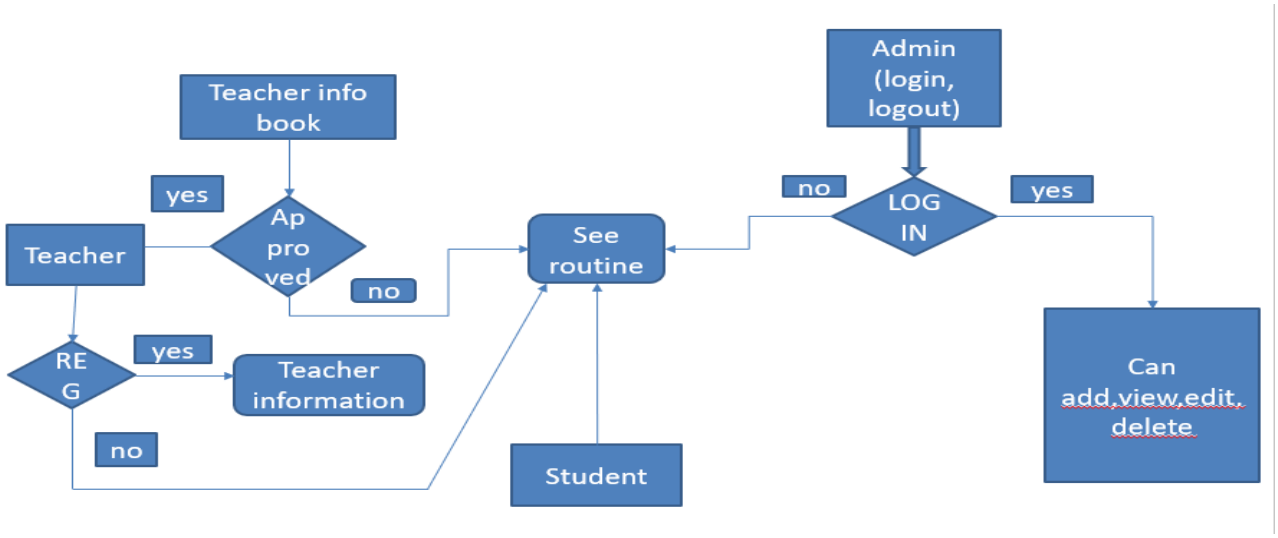


Figure 2.1: BPM diagram for total System

2.2 Related Works

Requirement gathering techniques are like brainstorming, questionnaires, interviews, user observation and document research are used for problem area to identify requirement.

By meeting with our supervisor first we list the requirements from our perspective. To collect real life requirements, we talk with student and teacher what type of problem they face. Collecting them we have find out this solution.” Class organizer apps” is similar and it the motivation of my

project. But in this apps Teacher can't directly book a room for when he/she needed extra class. Here in my project I add this facility extra here. In future I want to make it as android view.

2.3 Comparative Studies

Our University schedule is way of distributing assets such as teachers and classrooms over a fixed period of time. This task can be difficult and very long-delayed. If the operation of generating timetables is automated with the help of algorithms then this can help save both time and money for the educational institute. In this project a general schedule is presented along with a set of constraints commonly used in varsity scheduling. Two meta fact-finding algorithms with previous fulfilling results, pretended Annealing and Tabu Search, are implemented and benchmarked against each other in order to evaluate the performance of these. The results show that although both algorithms are good candidates for creating timetables, Simulated Annealing has the edge both in run time and the quality of the schedule.

2.4 Scope of the Problem

The dissimilarity area where we can use this implementation as:

- Any education institution makes use of it providing class schedule
- It can be utilized in offices and moderations can be easily done as stated by requirements.

2.5 Challenges

The system has been developed for research purpose to see how much impact it has. A product has been built without compromising its main goal. If we can get support from larger team and this project then lots of advanced and rich feature can be implemented. If the all goals of this project can be implemented with more new goals it will become one of the most popular and grateful projects in our country. We have tried our best to build this system better and fulfill all the requirements but some of them were not possible built for time shortage.

Major weakness or missing feature of this system are-

- Log of every action on this system.

CHAPTER 3

REQUIREMENT SPECIFICATION

3.1 Requirements Gathering

Requirement gathering techniques are like brainstorming, questionnaires, interviews, user observation and document research are used for problem area to identify requirement. By meeting with our supervisor first we list the requirements from our perspective. To collect real life requirements, we talk with student and teacher what type of problem they face. Collecting them we have find out this solution.

3.2 Requirement Collection and Analysis

User exceptions for new or modified product need analysis for determine their actual requirement, this process is known as Requirement Analysis. Requirements must be quantifiable, relevant and detailed. Functional and Non-functional requirements are available. By requirements gathering and specifying them requirements analysis done.

3.2.1 Functional Requirement: Functional requirement is given below

- The semester, course teacher information, routine is stored
- Course allotment is managed
- Class schedule is managed
- Users is also managed
- Class schedule status is viewed by routine
- The system is logged in
- Change the password

3.2.2 Non-functional Requirement:

- Font size and face acceptable for all device and user.
- Admin, Teacher and Student are responsible.
- Time and Frame through the development phase

- Include in all feature.
- The reason to choose for better usability and readability.
- Background color and font color acceptable for all devices and user.
- Security is also a major requirement of this system.

3.3 Use-Case Modeling and Description:

The following figure 3.1 shows the total system Use Case Diagram.

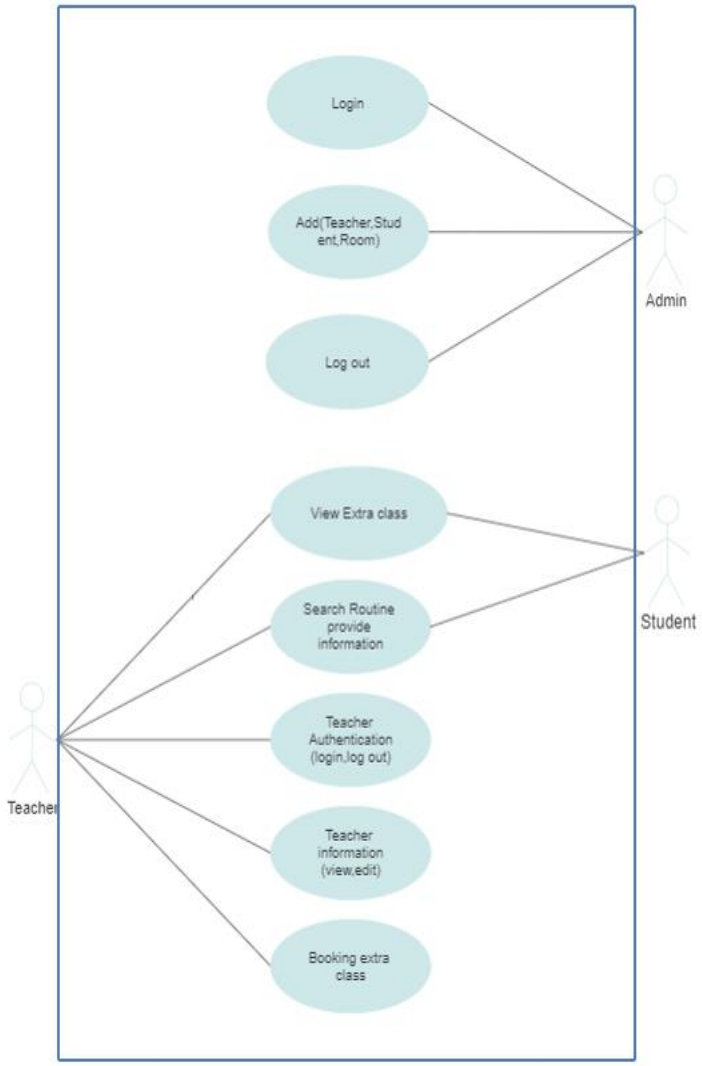


Figure 3.1: Use Case Diagram

For details Use Case Documentation Check Appendix Section.

3.4 Logical Data Model:

The complete schema design for proposed system.

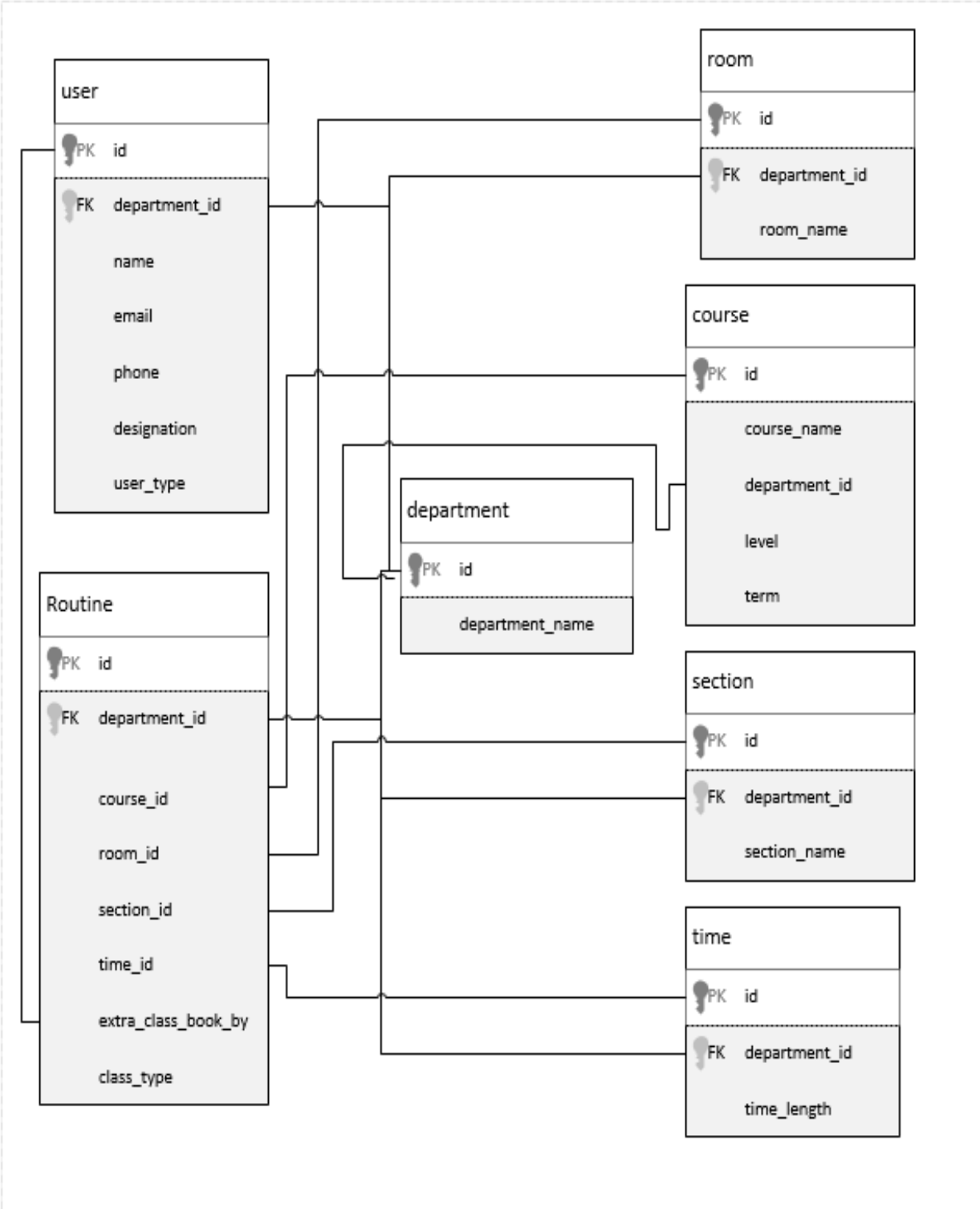


Figure 3.2: Database Design.

3.5 Design Requirements

Sequence diagram illustrate how the different parts of a system interact with each other to carry out a function, and the order in which the interactions occur when a particular use case is executed.

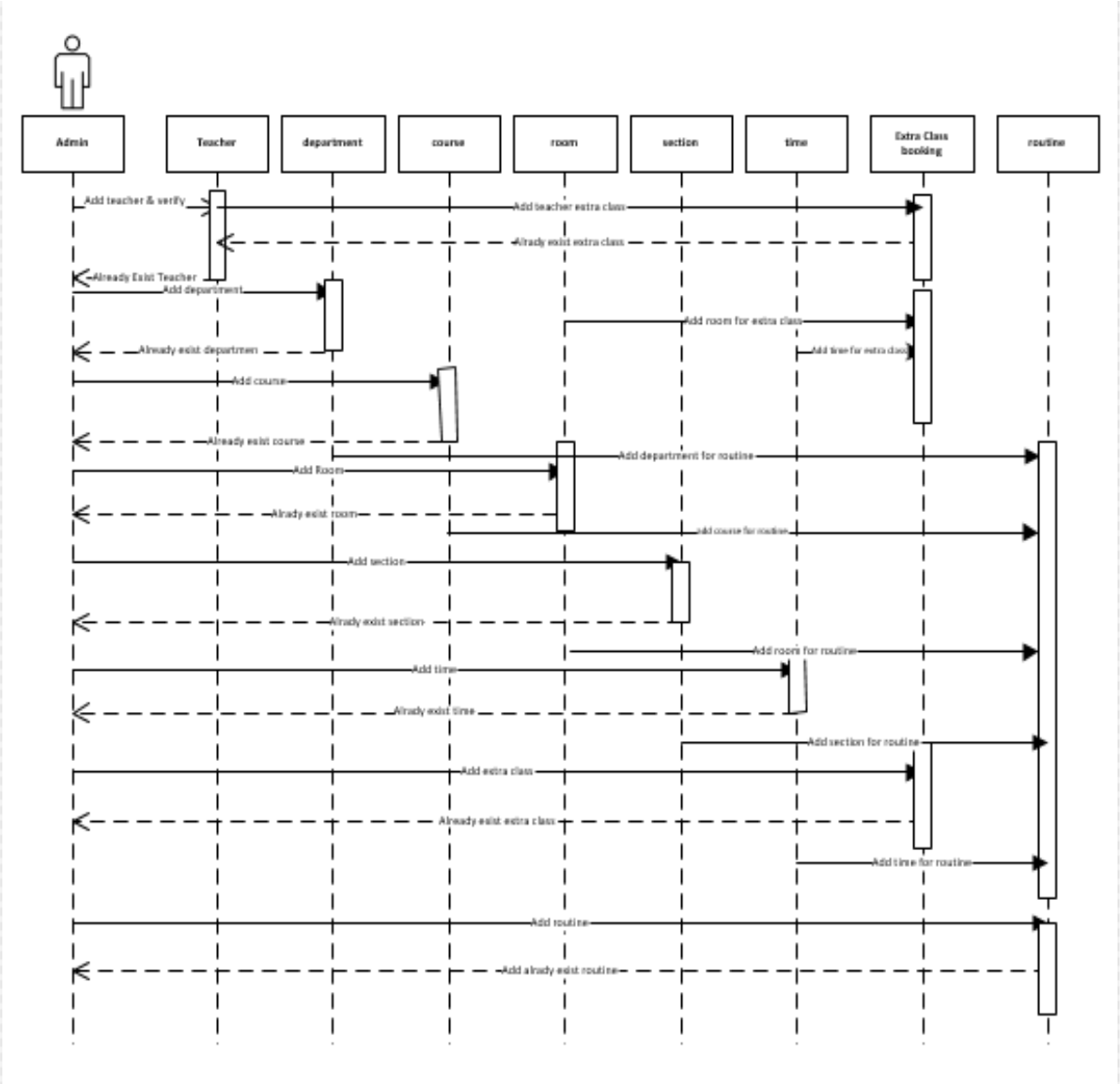


Figure 3.5: Sequence Diagram for admin

The following figure 3.5 shows the teacher's diagram system which is Sequentially arranged.

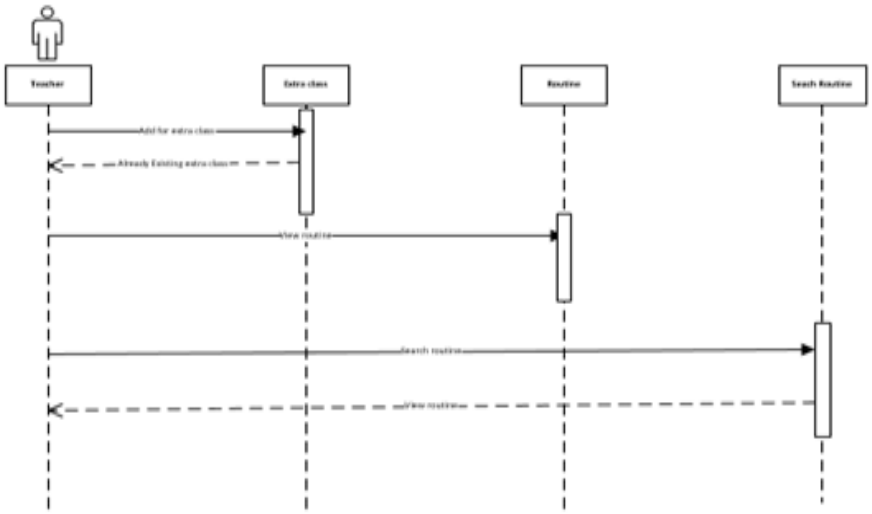


Figure 3.5: Sequence Diagram for teacher

The following figure 3.5 shows the student's diagram system which is Sequentially arranged.

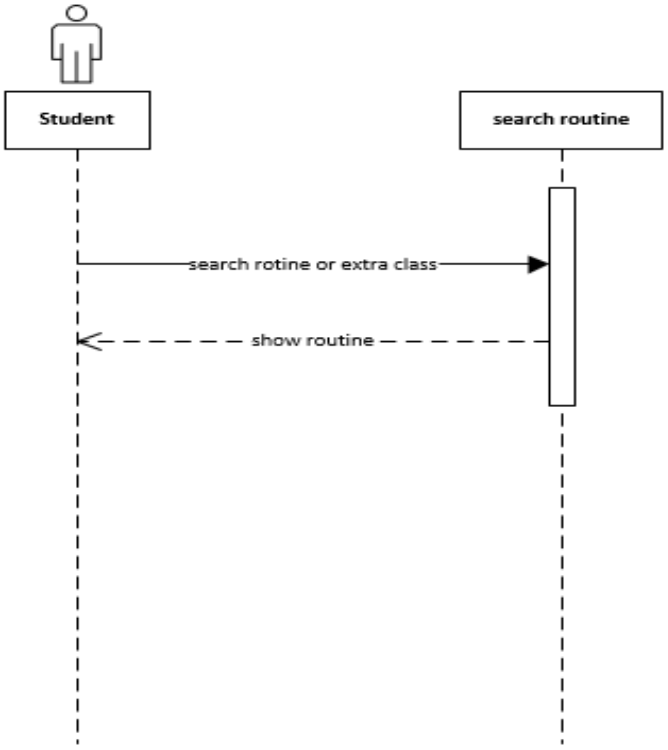


Figure 3.5: Sequence Diagram for student

CHAPTER 4

DESIGN SPECIFICATION

4.1 Front-end Design

- Java Script
- HTML
- CSS
- jQuery

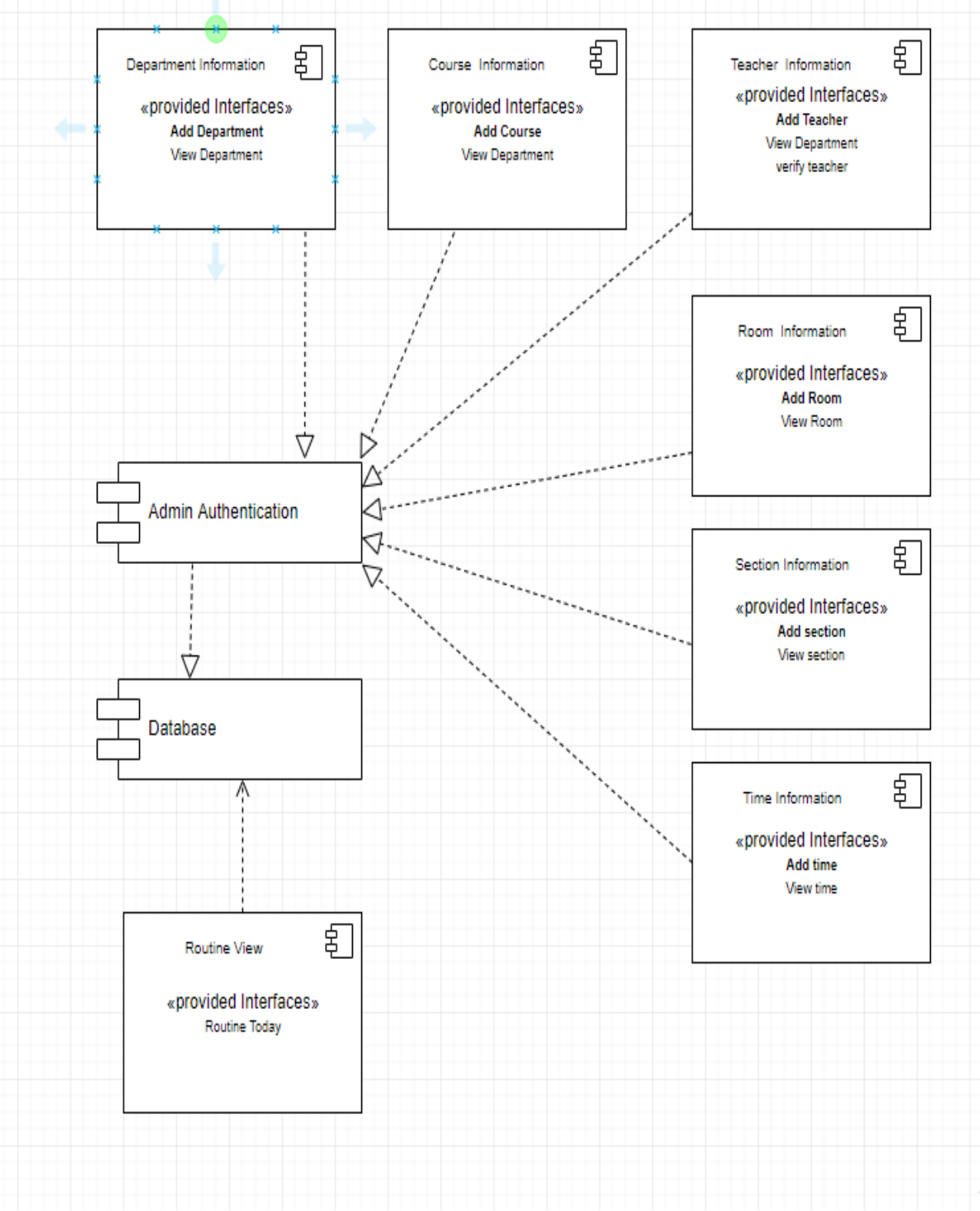
4.2 Back-end Design

- **MySQL database**
- **PHP as server-side scripting language**

4.3 Interaction Design

To achieve a better system to make development work easier, time consume and less error prone system design is must needed. System design is way to design structure of a system like architecture, component and database design etc.

4.3.1 Component Diagram



The relationship between different components in this system.

4.3.2 Class Diagram

The purpose of the class diagram describes the attributes and operations of a class and also the constraints imposed of this system.

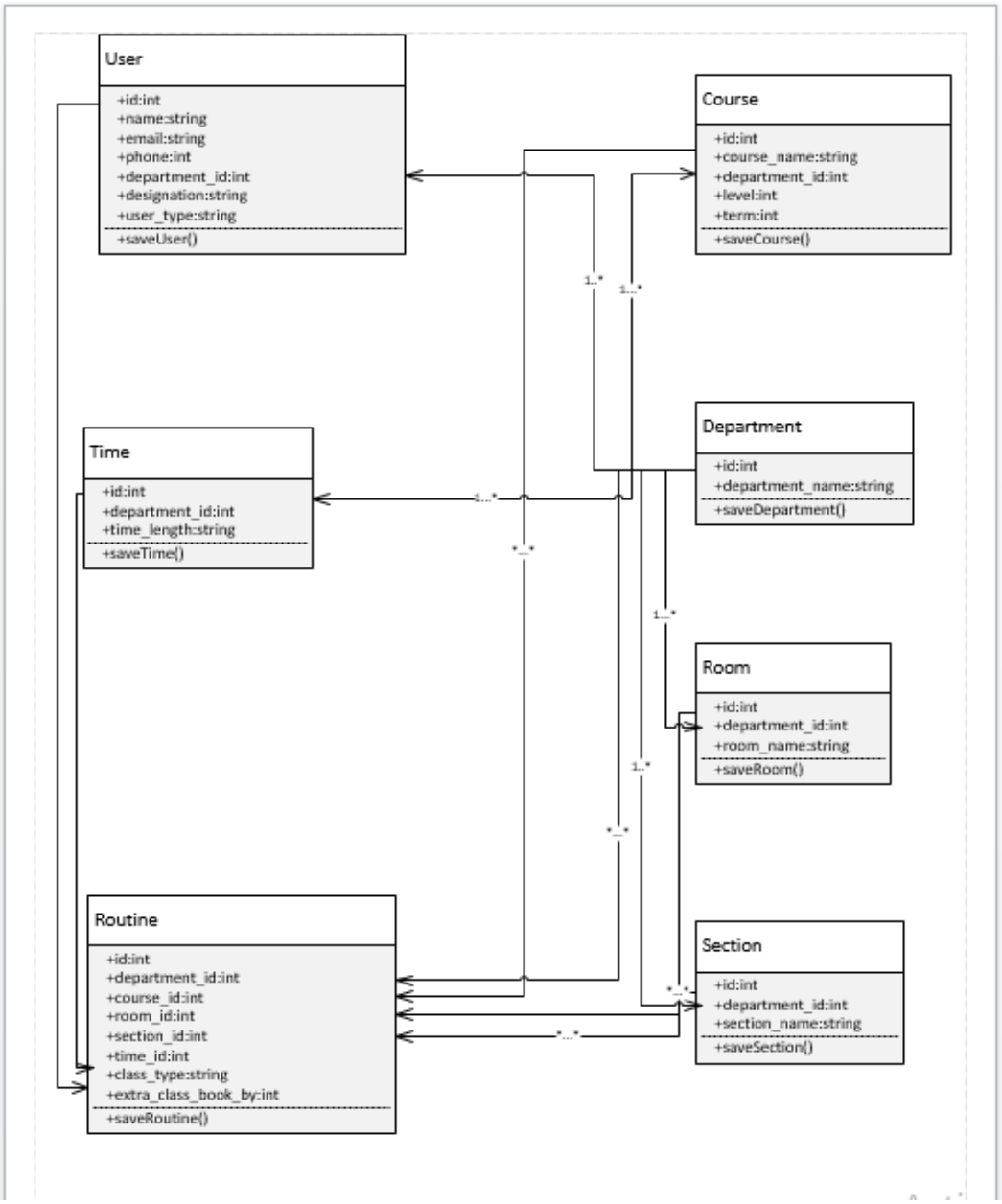


Figure 4.2: Class Diagram

4.3.3 Crow's Foot Entity Relationship Diagram

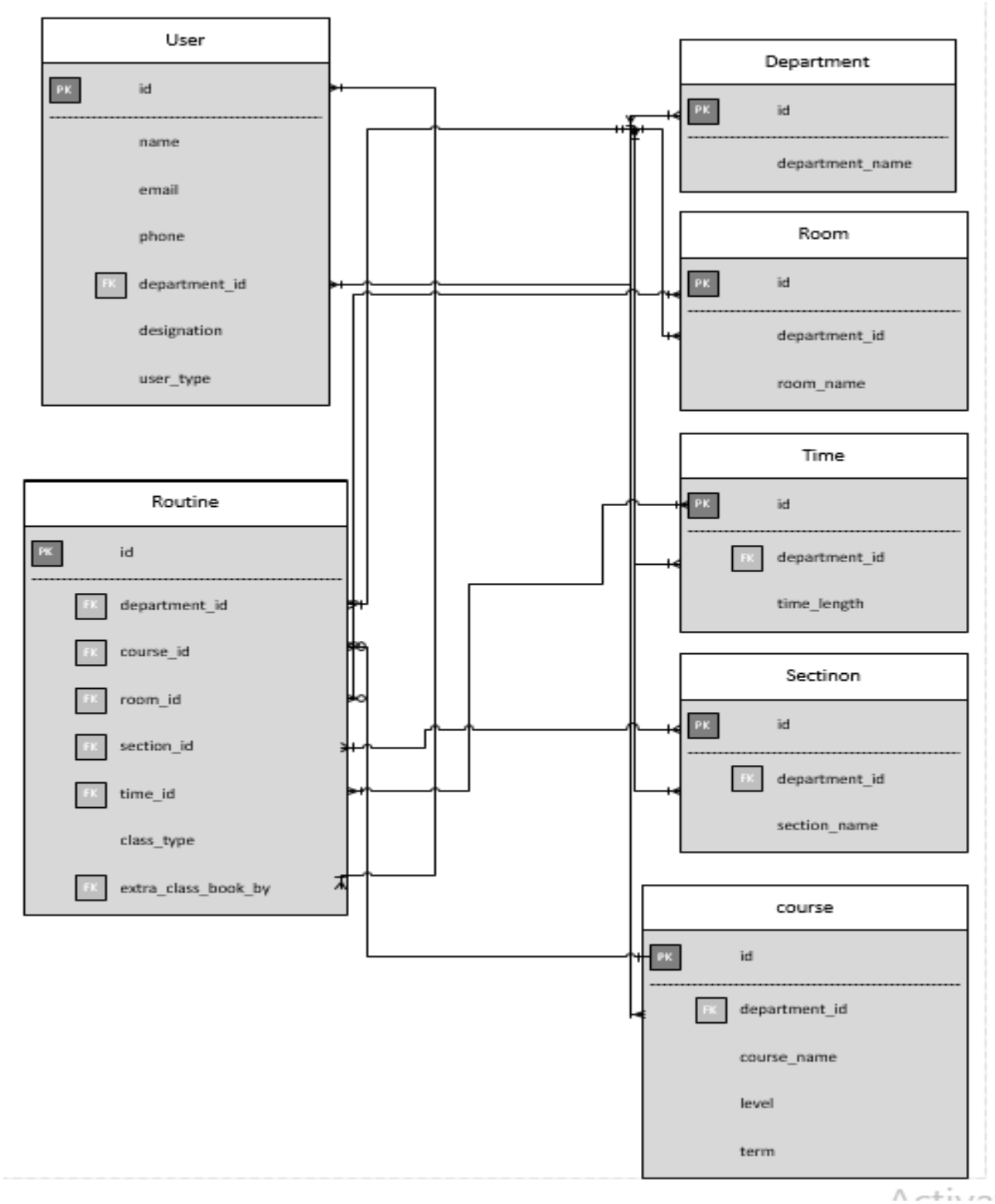


Figure 4.3: Crow's foot entity relationship diagram

4.4 Implementation Requirements

I have executed this system rely on my above investigation and database designing. To make this system, I have isolated the full system in some particular division such as user registration and login, add/edit semester, course, teacher information, handle course allotment, handle schedule and view routine. These division have been briefly discussed below.

- **Login**

Teacher can login

- **Dashboard**
- **Add/Edit syllabus information**

Admin like chairman can add/edit syllabus information

- **Add/Edit Semester Information**

Admin can add/edit semester information by the form.

- **Course List**

Admin can add/edit course information by this form.

- **Weekly Routine**

We can view this module as weekly routine.

- **Specific Routine**
- **Class Schedule**
- **Update Routine**
- **Course Allotment**

CHAPTER 5

IMPLEMENTATION AND TESTING

5.1 Introduction

Without testing it is not possible to trust the system functionality if it works properly or not. We will test the system that has been already built by following testing rules. Every parts of the system will be tested.

5.2 Test Case

For testing this system for any kind types this test case standard will follow

No	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail

5.3 Black Box Testing

In unit testing, here all the system units will be tested according to the test case and if it does not match the expected result, further action will be taken to fix the problem.

- For Admin

Table 5.1: Black box testing for admin

No	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Pass/Fail
01	Login (valid data)	• Go to/login URL • Enter Email id & password • Click submit	admin@email.com & 123456	Logged In	As expected,	Pass
02	Admin (invalid data)	• Go to/login URL • Enter email id & password	admin@email.com & 989482	Cannot browse any URL and redirect to login page	As expected,	Pass

		• Click submit				
03	Admin related URL check without login	Try to browse any admin related URL		Cannot browse any URL and redirect to login page	As expected,	Pass
04	Add new department	Click on "add department" button and fill up form	Provide all information	Save on DB and show in a page	As expected,	Pass
05	Add new course	Click on "add course" button and fill up form	Provide all information	Save on DB and show in a page	As expected,	Pass
06	Add teacher	Click on "add teacher" button and fill up form	Provide all information	Save on DB and show in a page	As expected,	Pass
07	Add room	Click on "add room " button and fill up form	Provide all information	Save on DB and show in a page	As expected,	Pass
08	Add section	Click on "add section" button and fill up form	Provide all information	Save on DB and show in a page	As expected,	Pass
09	Add time	Click on "add time " button and fill up form	Provide all information	Save on DB and show in a page	Ax expected	Pass
10	Add days	Click on "add days " button and fill up form	Provide all information	Save on DB and show in a page	As expected,	pass
11	Verify teacher	Click on "verify " button and fill up form	Provide all information	Save on DB and show in a page	As expected,	Pass

- For Teacher (public) section

Table: 5.2 Black box testing for Teacher

No	Test scenario	Test steps	Test Data	Expected Results	Actual Result	Pass/Fail
11	Add new teacher	Click on "add teacher " button and fill up form	Mr. x & mrx@gamil.com & 123456 etc.	Save in DB and show in a page	As expected	Pass
12	Add extra class	Click on "add extra class "	Provide all information	Save in DB and show in a page	As expected	Pass

		button and fill up form				
13	Search routine	Click on "search " button and fill up form	Provide all information	Save in DB and show in a page	As expected	Pass

- For Student (public) section

Table: 5.3 Black box testing for Student

No	Test Scenario	Test steps	Test data	Expected results	Actual results	Pass/Fail
14	Search routine	Click on "search" button and fill up form	Provide all information	Save in DB and show in a page	As expected	Pass

5.4 Module Testing

Modules are the combination of units. Here all the modules that are tested to prove that they are working as expected.

- For admin section

Table 5.4: Module testing for admin

No	Test scenario	Test steps	Test data	Expected results	Actual results	Pass/fail
01	Login (valid data)	1. Go to/login URL 2. Enter email id & password	admin@email.com & 123456	Logged in	As expected	Pass
02	Admin (invalid data)	1. Go to/login URL	admin@email.com & 9832445	Logged in	As expected,	Pass
03	Admin related URL check without login	Try to browse any admin related URL		Cannot browser any URL and redirect	As expected,	Pass

				to login page		
04	Teacher information	click on "new teacher" button to create new teacher button to create new teacher by fill up form and from auto generated id for teacher verification possible	Mr. Kamal & Mr. Hasan & Mr. Jaman	Save in DB and show in a page	As expected,	Pass
05	Department information	Click on "new department" button and give all information and view the department list	CSE & SWE	Save in DB and show in a page	As expected,	Pass
06	Course information	Click on "new course "button and give all information and view the course list	Computer fundamental & Learning c programming	Save in DB and show in a page	As expected,	Pass
07	Room Information	click on "new room" button and give all information and view the room list	AB-505 & CSE505	Save in DB and show in a page	As expected	Pass
08	Section information	click on "section" button and give all information and view section list	A & B & c	Save in DB and show in a page	As expected	Pass
09	Time information	click on "time" button and give all information and view time list	8:30 – 10:00 & 10:00 – 11:30	Save in DB and show in a page	As expected	Pass
10	Days information	Click on "days" button and give all information and view days list	Saturdays & Sunday & Monday	Save in DB and show in a page	As expected	Pass

- For teacher section

Table 5.5: Module testing for teacher

No	Test scenario	Test step	Test data	Expected results	Actual results	Pass/Fail
11	Teacher view	1. Go to /teacher URL 2. view information	Mr. Hasan	Show teacher basic info	As expected,	Pass

5.5 All Other Testing

- **Acceptance testing –** By unit and module testing its working as its expected which are already fulfill the acceptance testing so no need separately do it.
- **Performance testing –** We input lots of data and tested in so many ways to read and write data but system was stable and reliable.
- **Security testing –** By unit and module testing its working as its expected which are already fulfill the security testing so no need separately does it.

CHAPTER 6

CONCLUSION AND DEVELOPMENT

6.1 Discussion and Conclusion

This system is a very sensitive because of the information it collects and for its functionality so hand on experience will best option for it. One day training mandatory for all admin level user. The main goal is to make a central database and give web application to access them.

6.2 Scope for Further Developments

This is a developing software and it is good standard software. I am just trying to please the actual need of the department system need. There is always a scope for further development. I have made the software and try my level best to deliver an excellent system though it has some bug i will try to recuperate them.

REFERENCES

[1]"Business Process Modeling: Definition, Benefits and Techniques", Tallyfy, 2019. [Online]. Available at https://tallyfy.com/business-process-modeling,last accessed on: 31- 10- 2019 at 11:15am.

[2] "Customer Experience Management | CX Platform | Zoho CRM Plus", Zoho, 2019. [Online]. Available at https://www.zoho.com/crm/crmplus/?src=top-header&ireft=CRM,last accessed on 31- 10- 2019 at 8:15am.

[3] Lucidchart.com, 2019. [Online]. Available at https://www.lucidchart.com/pages/uml-use-case-diagram.,last accessed on 31- 10- 2019 at 10:06am.

[4] Lucidchart.com,2019. [Online]. Available at https://www.lucidchart.com/pages/er-diagrams,last accessed on 31-10-2019.

[5] "Ux design| CX Platform | Zoho CRM Plus", Zoho, 2019. [Online]. Available at https://www.zoho.com/crm/crmplus/?src=top-header&ireft=CRM,last accessed on 31- 10- 2019 at 07:08pm.

APPENDIX
Project Reflection

USE CASE DOCUMENTATION

Use case Name:	Routine Information Management
Actors:	Admin
Preconditions:	Admin logged in into system
Post conditions:	Admin has access permission to perform action
Primary scenario:	➢ Fill up form to add new department to database. ➢ Search department by name if anyone forget to enter name.

Use Case Name:	Admin Login
Actors:	Admin
Preconditions:	Entered into login page
Post Conditions:	➢ Enter into Dashboard ➢ New Session Created
Primary Scenario:	➢ Enter email and password ➢ System verify email and password ➢ If verified system allowed to enter dashboard otherwise give error message.

Use Case Name:	Subject Entry
Actor:	Admin
Preconditions:	Enter department info
Post Conditions:	Admin has access permission to perform action
Primary Scenario:	➢ Add department ➢ Check duplicate department

Use Case Name	Course Entry

Actor:	Admin
Preconditions:	Enter course info
Post Conditions:	Admin has access to perform action
Primary Scenario:	➢ Add course ➢ Add level ➢ Add term ➢ Check duplicate course

Use Case Name:	Verify teacher
Actor:	Admin
Preconditions:	Verify teacher
Post Conditions	Admin has access to perform action
Primary Scenario:	➢ Verify teacher ➢ View teacher info ➢ Edit or delete teacher

Use Case Name:	Room Entry
Actor:	Admin
Preconditions:	Enter room info
Post Conditions:	Admin has access to perform action
Primary Scenario:	➢ Add room ➢ View room ➢ Edit room ➢ Delete room

Use Case Name:	Sections Entry
Actor:	Admin
Preconditions:	Enter section info
Post Conditions:	Admin has access to perform action
Primary Scenario:	➢ Add section ➢ View section ➢ Edit section ➢ Delete section

Use Case Name:	Time Entry
Actor:	Admin
Preconditions:	Enter time info
Post Conditions:	Admin has access to perform action
Primary Scenario:	➢ Add time ➢ View time ➢ Edit time ➢ Delete time

Use Case Name:	Routine Entry
Actor:	Admin
Preconditions:	Enter routine info
Post conditions:	Admin has access to perform action
Primary Scenario:	➢ Add routine ➢ View routine ➢ Edit routine ➢ Delete routine

Use Case Name	Booking Extra Class
Actor:	Admin
Preconditions:	Enter extra class info
Post Conditions:	Admin has access to perform action
Primary Scenario:	➢ Add extra class ➢ View extra class ➢ Edit extra class ➢ Delete extra class

Use Case Name:	Search Routine
Actor:	Admin (public user)
Preconditions:	Search routine info
Post Conditions:	Admin has to perform action
Primary Scenario:	➢ Admin search class routine provide keyword

Use Case Name:	Register account
Actor:	Teacher
Preconditions:	Register
Post Conditions:	Teacher has access to perform action
Primary Scenario:	➢ Register own ➢ View profile ➢ Edit profile

Use Case Name:	Booking extra class
Actor:	Teacher
Preconditions:	Booking extra class
Post Conditions:	Teacher has access to perform action
Primary Scenario:	➢ Add extra class ➢ Edit extra class ➢ Delete extra class

Use Case Name:	Search routine
Actor:	Student (public user)
Preconditions:	Search routine
Post Conditions:	Student has access to perform action
Primary Scenario:	➢ Search routine provide information.

Turnitin Originality Report

Processed on: 09-Dec-2019 17:45 +06
ID: 1230530961
Word Count: 2867
Submitted: 1

Routine Management System (RMS) By Mansura Binta Mozid

Similarity Index	Similarity by Source	
15%	Internet Sources:	N/A
	Publications:	N/A
	Student Papers:	15%

2% match (student papers from 03-Jun-2014)
Submitted to University of Greenwich on 2014-06-03

2% match (student papers from 03-Apr-2018)
Submitted to University of Strathclyde on 2018-04-03

2% match (student papers from 04-Jul-2019)
Submitted to Florida Virtual School on 2019-07-04

2% match (student papers from 16-Aug-2019)
Submitted to University of Moratuwa on 2019-08-16

1% match (student papers from 26-Nov-2019)
Submitted to St. Petersburg High School on 2019-11-26

1% match (student papers from 03-Jul-2018)
Submitted to Indian Institute of Technology, Bombay on 2018-07-03

1% match (student papers from 20-Mar-2017)
Submitted to Queen Mary and Westfield College on 2017-03-20

1% match (student papers from 30-Aug-2017)
Submitted to Open University of Mauritius on 2017-08-30

1% match (student papers from 05-Jan-2018)
Submitted to Gulf College Oman on 2018-01-05

< 1% match (student papers from 26-Nov-2019)
Submitted to Daffodil International University on 2019-11-26

< 1% match (student papers from 02-Apr-2019)
Submitted to Daffodil International University on 2019-04-02

< 1% match (student papers from 19-Apr-2018)
Submitted to Colorado Technical University Online on 2018-04-19

< 1% match (student papers from 04-Nov-2019)
Submitted to Daffodil International University on 2019-11-04

< 1% match (student papers from 17-Jan-2010)
Submitted to University of Reading on 2010-01-17

< 1% match (student papers from 18-May-2018)
Submitted to Universiti Malaysia Perlis on 2018-05-18

< 1% match (student papers from 28-Apr-2016)
Submitted to University of Greenwich on 2016-04-28

< 1% match (student papers from 19-Mar-2016)
Submitted to University of Sunderland on 2016-03-19

< 1% match (student papers from 15-Jan-2019)
Submitted to Asia Pacific University College of Technology and Innovation (UCTI) on 2019-01-15

< 1% match (student papers from 03-Oct-2006)
Submitted to University of Sunderland on 2006-10-03

< 1% match (student papers from 11-Nov-2018)
Submitted to Midlands State University on 2018-11-11

1 ABSTRACT "Routine Management System (RMS)" is a fast, simple yet modern looking desktop-based application to create routines, can easily search classes by course name, course-code, teacher's initial or room number. The main target of this project to reduce the complexity to book any classroom and reduce the waste of time. I am trying to implement all the features which will come in handy and create a user- friendly interface. The All the data and resources are stored in a database for better access and security. Here users can add, update and delete content data of teachers, Rooms and routine. Routine can be viewed from different perspectives. The Admin has all access to verified teacher's identity. The most important issue in my project is that teacher can directly book room when they need extra classes. Here, Teachers can be identified by their email address and phone number. User can access here as a teacher or a student. Then after selecting their campus, department and program they go the next step and then their expected routine will be showed. User also can search by their teacher's initial to see Teacher's routine day wise. They also get the facility to search by subject name or code. The aim of this project to make more-easier to find routine and teacher accession for booking room when extra class needed. This application is also helpful to recycle the wasted time with proper way. 2 CHAPTER 1 INTRODUCTION 1.1 Introduction RMS is a software system that provides a user interface through a web browser. This is a fast, simple yet modern looking routine website designed for the admin, students and teachers of Daffodil International University. It currently fully supports the latest class routine of all Department. It is an open-source web-based system. It gives the best experiences for both teachers and students to find and book classroom. 1.2 Motivation I examine and apply an information system for run class routine for our University which is automated. The system which I suggest, makes easy to implement the handbook system. It will be able to cope up with class schedule of the department by present-day computer. It will help to supply all the information of the class plan and position of departments in a quicker process successfully. My project may be used as a model for all department. So, we see there are many problems to book and to find classroom in our University. The necessity of Technology of the modern lifestyle is unbelievable. Technology has improved human lives significantly by providing efficiency. It has made easily able to done for us to access education, communication, medicine, transportation, sports etc. Our

aim is student & teacher can find out their class routine easily. Student, Teacher & Admin can save their time. 1.3 Objectives Object's world we live in. when we use these products, we see this existence in nature in human made attribute's, in business. We are categorized it, and combine it, manipulate it, combine it, and also can create it. The technology which we use is a new way of sense about problems using part by part module put in order to around client real-world idea. 3 I used here web-based applied science as PHP and MySQL for database to plan and create my project proposal, routine management system in client server environment.I make this project for developing software manufacture. The details system with at least handbook interposition. I designed the Software to do informational tasks like Teachers, Student and the Admin. I select this software to analyze and design and finally implement it. ? MySQL for Designing the database ? PHP ? Java Script, HTML, CSS, Java Script and jQuery for UI Design 1.4 Expected Outcome The expectation of making this system teacher and student can find out their class room easily and able to booking classroom easily. ? Admin can check all user list ? Identify them ? Manage all users ? Teacher can watch class routine ? Teacher can book classroom ? Student can watch class routine 4 CHAPTER 2 BACKGROUND 2.1 Introduction Technology is necessity of the modern lifestyle. Technology has improved human lives significantly by providing convenience and efficiency. It has made easily possible for us to access education, communication, medicine, transportation, sports etc. Our aim is student & teacher can find out their class routine easily. Student & teacher can save their time. The following figure 2.1 shows Business Process Modeling. Figure 2.1: BPM diagram for total System 2.2 Related Works Requirement gathering techniques are like brainstorming, questionnaires, interviews, user observation and document research are used for problem area to identify requirement. 5 By meeting with our supervisor first we list the requirements from our perspective. To collect real life requirements, we talk with student and teacher what type of problem they face. Collecting them we have find out this solution." Class organizer apps" is similar and it the motivation of my project. But in this apps Teacher can't directly book a room for when he/she needed extra class. Here in my project I add this facility extra here. In future I want to make it as android view. 2.3 Comparative Studies Our University schedule is way of distributing assets such as teachers and classrooms over a fixed period of time. This task can be difficult and very long-delayed. If the operation of generating timetables is automated with the help of algorithms then this can help save both time and money for the educational institute.In this project a general schedule is presented along with a set of constraints commonly used in varsity scheduling. Two meta fact-finding algorithms with previous fulfilling results, pretended Annealing and Tabu Search, are implemented and benchmarked against each other in order to evaluate the performance of these.The results show that although both algorithms are good candidates for creating timetables, Simulated Annealing has the edge both in run time and the quality of the schedule. 2.4 Scope of the Problem The dissimilarity area where we can use this implementation as: ? Any education institution makes use of it providing class schedule ? It can be utilize in offices and moderations can be easily done as stated by requirements. 2.5 Challenges The system has been developed for research purpose to see how much impact it has. A product has been built without compromising its main goal. If we can get support from larger team and this project then lots of advanced and rich feature can be implemented. If the all goals of this project can be implemented with more new goals it will become one of the most popular and grateful projects in our country. We have tried our best to build this system better and fulfill all the requirements but some of them were not possible built for time shortage. 6 Major weakness or missing feature of this system

are- ? Log of every action on this system. 7 CHAPTER 3 REQUIREMENT SPECIFICATION 3.1 Requirements Gathering Requirement gathering techniques are like brainstorming, questionnaires, interviews, user observation and document research are used for problem area to identify requirement.By meeting with our supervisor first we list the requirements from our perspective. To collect real life requirements, we talk with student and teacher what type of problem they face. Collecting them we have find out this solution. 3.2 Requirement Collection and Analysis User exceptions for new or modified product need analysis for determine their actual requirement, this process is known as Requirement Analysis. Requirements must be quantifiable, relevant and detailed. Functional and Non-functional requirements are available. By requirements gathering and specifying them requirements analysis done. 3.2.1 Functional Requirement: Functional requirement is given below ? ? The semester, course teacher information, routine is stored Course allotment is managed ? Class schedule is managed ? Users is also managed ? Class schedule status is viewed by routine ? The system is logged in ? Change the password 3.2.2 Non-functional Requirement: 8 ? Font size and face acceptable for all device and user. ? Admin, Teacher and Student are responsible. ? Time and Frame through the development phase ? Include in all feature. ? The reason to choose for better usability and readability. ? Background color and font color acceptable for all devices and user. ? Security is also a major requirement of this system. 3.3 Use-Case Modeling and Description: The following figure 3.1 shows the total system Use Case Diagram. Figure 3.1: Use Case Diagram For details Use Case Documentation Check Appendix A Section. 9 3.4 Logical Data Model: The complete schema design for proposed system. Figure 3.2: Database Design 10 3.5 Design Requirements Sequence diagram illustrate how the different parts of a system interact with each other to carry out a function, and the order in which the interactions occur when a particular use case is executed. Figure 3. 5: Sequence Diagram for admin 11 The following figure 3.5 shows the teacher's diagram system which is Sequentially arranged. Figure 3. 5: Sequence Diagram for teacher The following figure 3. 5 shows the student's diagram system which is Sequentially arranged. Figure 3.5: Sequence Diagram for student 12 CHAPTER 4 DESIGN SPECIFICATION 4.1 Front-end Design ? Java Script ? HTML ? CSS ? jQuery 4.2 Back-end Design ? MySQL database ? PHP as server-side scripting language 4.3 Interaction Design and UX To achieve a better system to make development work easier, time consume and less error prone system design is must needed. System design is way to design structure of a system like architecture, component and database design etc. 13 4.3.1 Component Diagram The relationship between different components in this system. 14 4.3.2 Class Diagram The purpose of the class diagram describes the attributes and operations of a class and also the constraints imposed of this system. Figure 4.2: Class Diagram 15 4.3.3 Crow's Foot Entity Relationship Diagram Figure 4.3: Crow's foot entity relationship diagram 16 UX Design Figure 4.4: UX diagram [5] 4.4 Implementation Requirements I have executed this system rely on my above investigation and database designing. To make this system, I have isolated the full system in some particular division such as user registration and login, add/edit semester, course, teacher information, handle course allotment, handle schedule and view routine. These division have been briefly discussed below. ? Login Teacher can login ? Dashboard ? Add/Edit syllabus information Admin like chairman can add/edit syllabus information 17 ? Add/Edit Semester Information Admin can add/edit semester information by the form. ? Course List Admin can add/edit course information by this form. ? Weekly Routine We can view this module as weekly routine. ? Specific Routine ? Class Schedule ? Update Routine ? Course Allotment 18 CHAPTER 5 IMPLEMENTATION AND TESTING 5.1 Introduction Without testing it is not

possible to trust the system functionality if it works properly or not. We will test the system that has been already built by following testing rules. Every parts of the system will be tested. 5.2 Test Case For testing this system for any kind types this test case standard will follow No Test Scenario Test Steps Test Data Expected Results Actual Results Pass/Fail 5.3 Black Box Testing In unit testing, here all the system units will be tested according to the test case and if it does not match the expected result, further action will be taken to fix the problem. ? For Admin Table 5.1: Black box testing for admin No Test Scenario Test Steps Test Data Expected Results Actual Results Pass/F ail 01 Login (valid data) ? ? ? Go to/ login URL Enter Email id & password Click submit admin@email.com & 123456 Logged In As expected, Pass 19 02 Admin (invalid data) ? Go to/login URL ? Enter email id & password ? Click submit admin@email.com & 989482 Cannot browse any URL and redirect to login page As expected, Pass 03 Admin related URL check without login Try to browse any admin related URL Cannot browse any URL and redirect to login page As expected, Pass 04 Add new department Click on "add department" button and fill up form Provide information all Save on DB and show in a page As expected, Pass 05 Add new course Click on "add course" button and fill up form Provide information all Save on DB and show in a page As expected, Pass 06 Add teacher Click on "add teacher" button and fill up form Provide information all Save on DB and show in a page As expected, Pass 07 Add room Click on "add room " button and fill up form Provide information all Save on DB and show in a page As expected, Pass 08 Add section Click on "add section" button and fill up form Provide information all Save on DB and show in a page As expected, Pass 09 Add time Click on "add time " button and fill up form Provide information all Save on DB and show in a page Ax expected Pass 10 Add days Click on "add days " button and fill up form Provide information all Save on DB and show in a page As expected, pass 11 Verify teacher Click on "verify " button and fill up form Provide information all Save on DB and show in a page As expected, Pass ? For Teacher (public) section Table: 5.2 Black box testing for Teacher No Test scenario Test steps Test Data Expected Results Actual Result Pass/Fail 11 Add new teacher Click on "add teacher " button and fill up form Mr. x & mrx@gamil.com & 123456 etc. Save in DB and show in a page As expected Pass 20 12 Add extra class Click on "add extra class " button and fill up form Provide information all Save in DB and show in a page As expected Pass 13 Search routine Click on "search " button and fill up form Provide information all Save in DB and show in a page As expected Pass ? For Student (public) section Table: 5.3 Black box testing for Student No Test Scenario Test steps Test data Expected results Actual results Pass/Fail 14 Search routine Click on "search" button and fill up form Provide information all Save in DB and show in a page As expected Pass 5.4 Module Testing Modules are the combination of units. Here all the modules that are tested to prove that they are working as expected. ? For admin section Table 5.4: Module testing for admin No Test scenario Test steps Test data Expected results Actual results Pass/fail 01 Login (valid data) 1. 2. Go to/ login URL Enter email id &password admin@email.com & 123456 Logged in As expected Pass 02 Admin (invalid data) 1. Go to/login URL admin@email.com & 9832445 Logged in As expected, Pass 21 03 Admin related URL check without login Try to browse any admin related URL Cannot browser any URL and redirect to login page As expected, Pass 04 Teacher information click on "new teacher" button to create new teacher button to create new teacher by fill up form and from auto generated id for teacher verification possible Mr. Kamal & Mr. Hasan & Mr. Jaman Save in DB and show in a page As expected, Pass 05 Department information Click on "new department" button and give all information and view the department list CSE & SWE Save in DB and show in a page As expected, Pass 06 Course information Click on "new course "button and give all information and view

the course list Computer fundamental & Learning c programming Save in DB and show in a page As expected, Pass 07 Room Information click on "new room" button and give all information and view the room list AB-505 & CSE505 Save in DB and show in a page As expected Pass 08 Section information click on "section" button and give all information and view section list A & B & c Save in DB and show in a page As expected Pass 09 Time information click on "time" button and give all information and view time list 8:30 – 10:00 & 10:00 – 11:30 Save in DB and As expected Pass 22 show in a page 10 Days information Click on "days" button and give all information and view days list Sunday & Monday Saturdays & Save in DB and show in a page As expected Pass ? For teacher section Table 5.5: Module testing for teacher No Test scenario Test step Test data Expected results Actual results Pass/Fail 11 Teacher view 1. Go to /teacher URL 2. view information Mr. Hasan Show teacher basic info As expected, Pass 5.5 All Other Testing ? Acceptance testing – By unit and module testing its working as its expected which are already fulfill the acceptance testing so no need separately do it. ? Performance testing – We input lots of data and tested in so many ways to read and write data but system was stable and reliable. ? Security testing – By unit and module testing its working as its expected which are already fulfill the security testing so no need separately does it. 23 CHAPTER 6 CONCLUSION AND DEVELOPMENT 6.1 Discussion and Conclusion This system is a very sensitive because of the information it collects and for its functionality so hand on experience will best option for it. One day training mandatory for all admin level user. The main goal is to make a central database and give web application to access them. 6.2 Scope for Further Developments This is a developing software and it is good standard software. I am just trying to please the actual need of the department system need. There is always a scope for further development. I have made the software and try my level best to deliver an excellent system though it has some bug i will try to recuperate them. 24 25