



# **Management Information System**

## **For**

# **BIT Final Year Projects**

**A dissertation submitted for the Degree of Master of  
Information Technology**

**K.W.N.S.Anuradhani**  
**University of Colombo School of Computing**  
**2019**



# Declaration

The thesis is my original work and has not been submitted previously for a degree at this or any other university/institute.

To the best of my knowledge it does not contain any material published or written by another person, except as acknowledged in the text.

Student Name: K W N S Anuradhani

Registration Number: 2016MIT004

Index Number: 16559943

---

Signature:

---

Date:

This is to certify that this thesis is based on the work of Ms. K W N S Anuradhani under my supervision. The thesis has been prepared according to the format stipulated and is acceptable standard.

Certified by:

Supervisor Name: Mr. W V Welgama

---

Signature:

---

Date:

## **Acknowledgement**

This thesis could not have been written without the help of Mr. W. V. Welgama who not only served as my supervisor but also helped and encouraged me throughout the project and the life.

I wish to express my sincere gratitude to the MIT Coordinator of University of Colombo School of Computing (UCSC) and lecturers of Masters of Information Technology (MIT) for giving me this wonderful opportunity to apply the knowledge gained through the MIT degree program.

I honestly thank all my colleagues a lot in different ways and encouraged me to complete the project successfully.

Last but not the least; I would like to convey my heartiest gratitude to my parents and siblings for their enormous assistance and guidance throughout the project

## **Abstract**

Bachelor of Information Technology (BIT) is one of the major degree program conducted by University of Colombo School of computing (UCSC) for around 25 years. Thousands of graduates were created by this degree program and most of them are engaged in software related jobs in Sri Lanka as well as in other countries. The individual project is by far the most important task they complete in their degree program. All the students who registered for the 3rd year and paid the project fee are eligible to carry out the individual project.

There is a Moodle based Virtual Learning Environment for the students who has enrolled into the final year project course. Currently, most of the works in the evaluation process in BIT are done manually by coordinators. Since coordinators are unable to conduct the whole evaluation process through the VLE, viewing all the submissions in student point of view has become a difficult task. It means, details of students with all the submissions corresponding dates of each students is not shown in the VLE. Thus, If one student is not submitted the proposal to start the project, other submission links should be restricted. However the current system(VLE) is not restricted the submission links with comparing previous links.

Considering above factors, I am motivated to develop a software solution which will support coordinators to effectively and efficiently managing the evaluation process while fulfilling the satisfaction level of students.

This thesis describes an Information Management System covering the aspects relevant to the above mentioned problems and handling submissions, registration, deadlines, notifications, chatting and updating news , guidelines which is to be implemented for BIT final year project at University of Colombo School of Computing.

# Contents

Acknowledgement .....	ii
Abstract.....	iii
List of Figures .....	vi
List of Tables .....	vii
Abbreviations.....	viii
Chapter 1: Introduction .....	1
1.2 Project Objectives.....	2
1.3 Scope of the Project .....	2
1.3.1 Scope .....	2
1.3.2 Limitations .....	3
1.3.3 Target Users.....	3
1.4 Work Break-Down Chart .....	4
1.5 Structure of the Dissertation .....	5
Chapter 2: Analysis .....	6
2.1 Introduction .....	6
2.2 Gathering Requirements .....	6
2.2.1 Interviews .....	6
2.2.2 Self-Experience .....	6
2.2.3 Observation of the Work Environment .....	6
2.3 Current Manual System Analysis .....	7
2.4 Review of the Similar Works .....	9
2.4.1 Virtual Learning Environment-University of Colombo, School of Computing .....	9
2.4.2 Computerized Assignment/Project Submission System For F.S.K.T.M (PHILLOMEANA ANAK LIMBANG).....	10
2.4.3 VLE- Faculty of Management and Finance, University of Colombo .....	10
2.4.4 VLE, IHRA, University of Colombo .....	11
2.4.5 Learning Organizer, University of Moratuwa.....	11
2.5 Similar Systems Comparison Matrix .....	12
2.6 Comparison of Alternative Design Strategies .....	13
2.7 Functional Requirements .....	13
2.8 Non-Functional Requirements.....	14
Chapter 3:Design .....	16
3.1 Introduction .....	16

3.2 Use Case Diagram for the Proposed System .....	16
3.3 Methodology and Implementation Considerations .....	21
3.4 The flow of the Interactions.....	22
3.6 Class Diagram for the Proposed System .....	24
3.7 User Interfaces for the Proposed System.....	25
3.7.1 Log in Interface .....	26
3.7.2 Student Dashboard .....	26
3.7.3 Coordinator Dashboard .....	27
3.7.4 Edit Profile .....	27
3.7.5 Guidelines and Templates .....	28
3.7.6 Insert eligible list.....	29
3.7.7 Student Registration .....	29
Chapter 4:Implementation .....	30
4.1 Introduction .....	30
4.2 Development Environment.....	30
4.2.1 Hardware Requirements.....	30
4.2.2 Software Requirements.....	30
4.2.3 Technologies and Languages used .....	30
4.3 Code structure to connect the database .....	31
4.4 Reused Components .....	32
Chapter 5 : Evaluation .....	33
5.1 Introduction .....	33
5.2 Manual Test Plan .....	33
5.2.1 Test cases.....	33
5.1.3 Automated Test plan.....	38
Chapter 6: Conclusion.....	39
6.1 Introduction .....	39
6.2 Future Improvements.....	39
6.3 Lessons Learnt .....	39
References.....	40

## List of Figures

Figure 1.1: WBS Chart for the building workable web based system.....	4
Figure 2.1: Current manual system.....	8
Figure 2.2: Virtual Learning Environment - UCSC.....	9
Figure 2.3: Virtual Learning Environment, FOMF.....	10
Figure 2.4: Virtual Learning Environment, IHRA.....	11
Figure 2.5: Learn.org.UOM.....	11
Figure 3.1: Submission 1, event point of view (UCSC) .....	17
Figure 3.2: Submission 2, event point of view (UCSC) .....	18
Figure 3.3: All submission name with grades (UCSC).....	18
Figure 3.4: Submission, view point of view (FMF, UOC) .....	19
Figure 3.5: Submission 2, Event point of view (FMF, UOC).....	19
Figure 3.6: Use Case Diagram for the Proposed System.....	20
Figure 3.7: Client Server Architecture .....	22
Figure 3.8: Flow of the 1st Interaction with the System.....	22
Figure 3.9: ER diagram for the Proposed System.....	23
Figure 3.10: Class Diagram for Proposed System .....	24
Figure 3.11: Logging interface.....	26
Figure 3.12: 1st Interacting dashboard.....	26
Figure 3.13: Coordinator Dashboard .....	27
Figure 3.14: Edit student profile .....	27
Figure 3.15: Insert guidelines and upload templates .....	28
Figure 3.16: View Guidelines .....	28
Figure 3.17: Create eligibility list .....	29
Figure 3.18: Student Registration .....	29
Figure 4.1: Code structure to Connect Database .....	31
Figure 5.1: Katalone Recorder Interface.....	38

## List of Tables

Table 2.1: Similar System Comparison Matrix.....	12
Table 5.1: Test cases for the Logging Module .....	34
Table 5.2: Test cases for the Submission Module for Students .....	35
Table 5.3: Test cases for Submission Module.....	36
Table 5.4: Test Cases for Student Enrollment and Student Registration .....	37



## **Abbreviations**

BIT- Bachelor of Information Technology

IHRA- Institute of Human Resource Advancement

MGT- Management

MRT- Moratuwa

PHP - Hypertext Pre- processor

SQL - Structured Query Language

UCSC- University of Colombo School of computing

VLE- Virtual Learning Environment

# **Chapter 1: Introduction**

University of Colombo School of Computing (UCSC) is one of the main IT Universities in Sri Lanka. Currently, UCSC is conducting external degree program named Bachelor of Information Technology (BIT) and more than thousand students registered for the degree in each year. The individual project is by far the most important task they complete in their degree program. All the students who registered for the 3<sup>rd</sup> year and paid the project fee are eligible to carry out the individual project. Final year project is compulsory for all the students to obtain the BIT degree. This system is developed to help the BIT 3<sup>rd</sup> year students' project submissions.

## **1.1 Problem**

Currently, most of the works in the evaluation process are done manually by coordinators. Only the reports submission, displaying relevant guidelines and providing useful resources are managed through the project VLE (Virtual Learning Environment). Students submit their project proposal, interim report, progress reports, client agreement form, supervisor agreement form, tentative dissertation, and final dissertation periodically over the whole year. Project VLE is used only as a place to collect all those reports submitted by students. Since coordinators are unable to conduct the rest of the evaluation process through the VLE, scheduling the viva and dissertation evaluations and providing their grades calculating the evaluators' marks have become a difficult task. The major problem is, students details with all the submissions corresponding dates of the each students is not shown in the VLE. Thus, If one student is not submitted the proposal to start the project, other submission links should be restricted. However the current system is not restricted the submission links with comparing previous links.

Considering above details, I am motivated to provide a software solution which will support coordinators to effectively and efficiently managing the evaluation process while fulfilling the satisfaction level of users.

## 1.2 Project Objectives

The main objective of the proposed system is to automate the above-mentioned problem using Information Technology to help all the users of the system. Following are the other objectives of the system.

- To collect all the project relevant details in to one location.
- To provide the web-based system which can access information from any location.
- To track the students' progress and performance easily.
- To manage the students' details and easily access the student information
- To ensure data integrity, privacy, and security.
- To replace the existing manual system which is time consuming, when coordinator is organizing the evaluation process

## 1.3 Scope of the Project

### 1.3.1 Scope

Proposed system is capable of managing, processing, maintaining, searching, and viewing the project related details of the system. More of the scope details are given below.

- **Providing resources** - The system provides templates, guidelines, and other details relevant to their submissions. However, the system does not create an e-learning environment for self-study.
- **Managing user accounts and maintaining their biographical details** (profile management) - The system manages user accounts and maintains their biographical details for the scheduling and evaluation process. However, the system does not allow users to register to the system by themselves and does not consider the payment method or fee.
- **Managing submissions-** More than ten submissions have to be made by the each students periodically over the year.

- **Keep tracking the progress of students** - The system keeps each student progress individually. So, admin can generate reports based on the student progress.
- **Scheduling** - Creating schedules for viva and dissertation evaluation is one of the main functions of the system. However, the system does not support to create the list of staff allocation.
- **Generating reports according to the needs**
- **Communication** - Students have facility to communicate with the coordinator regarding their project relevant issues.

### 1.3.2 Limitations

- The proposed system handles only the projects and does not direct to other subjects of the students
- The proposed system is not handled the student registration process and payments
- The proposed system is not link with the existing VLE

### 1.3.3 Target Users

- **BIT 3<sup>rd</sup> year students**  
Students who are eligible for the project are main users of the system.
- **Project coordinator**  
Project coordinator handles the coordination of the BIT evaluation process and plays the admin role of the system.
- **Project Assistant**  
Project assistant handles the assist tasks for the BIT evaluation process

## 1.4 Work Break-Down Chart

The purpose of this chart is to break down complex activities into smaller, more management components. All the stages with including the deliverables relevant to each stage are depicted in the chart.

According to the project, WBS chart for the build Workable web based system is shown in the Figure 1.1.

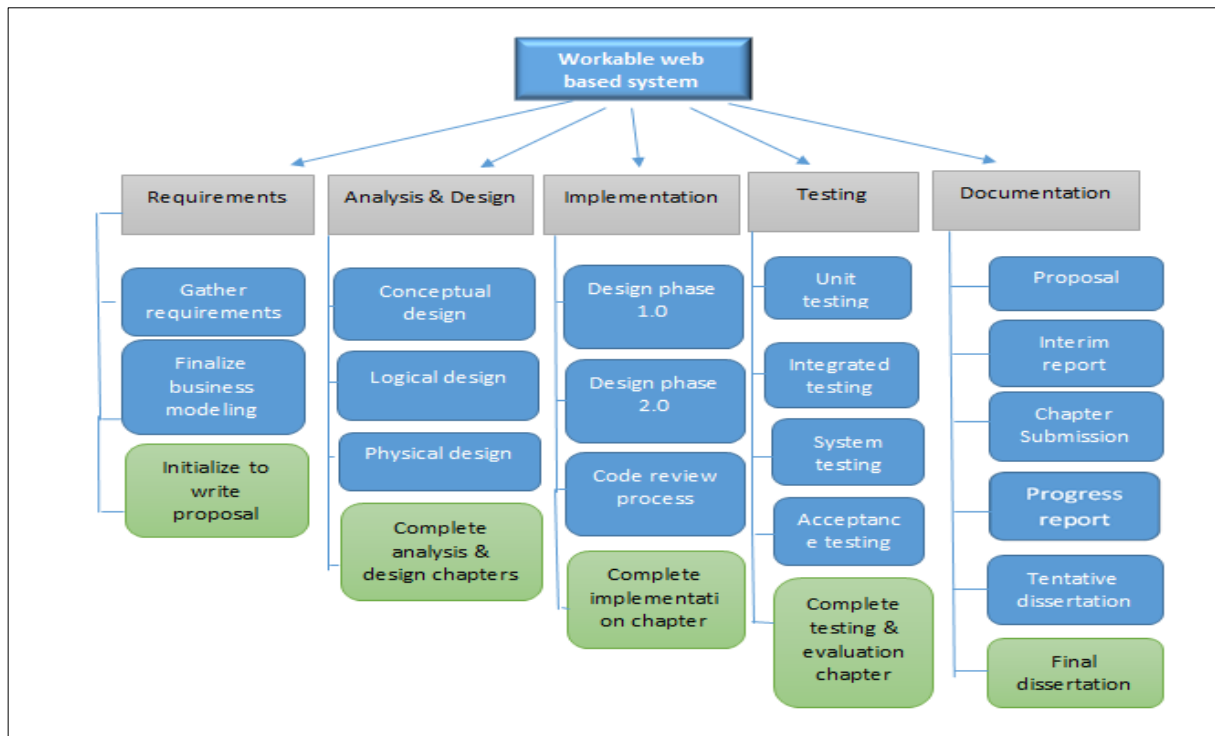


Figure 1.1: WBS Chart for the building workable web based system

## **1.5 Structure of the Dissertation**

This Dissertation contains of 6 chapters regarding the process of Information Management System for BIT final year projects. Those chapters are listed below.

Chapter 02 is the Analysis chapter and it explains the requirement gathering techniques, details of the manual system, functional nonfunctional requirements and details of the existing similar systems.

Chapter 03, Design chapter mainly focuses the designed methodology, alternate solutions, designed diagrams and interface designs. During the design phase, the system is designed to satisfy the requirements identified in the analysis phase. Chapter 04, Implementation chapter explains the hardware software requirements, development tools which is used for the system, code features and reused existing codes of the system.

Evaluation chapter is the Chapter 05 and it explains the techniques of testing, details of software testing, high level test plan of the system. Chapter 06, Conclusion chapter explains the future enhancements of the system and lesson learnt of the overall project work.

## **Chapter 2: Analysis**

### **2.1 Introduction**

“System analysis is one of the main phases in the software development life cycle. System analysis will help to get an overall image of the system and will be able to produce a high-level description of the system through this phase. Main objectives of this phase are what services system should provide, required performance of the system. Before analyzing the system, first the requirements should be gathered by using the fact-finding techniques, such as interviews, observations, sample documentations etc” [1].

Analysis stage helps to understand currently working system and it can identify how overcome the current problems introducing new system. Further, it is a way to collect functional and non-functional requirements of the system.

### **2.2 Gathering Requirements**

Requirement gathering is an essential and critical stage to provide a better system. There are so many techniques are used to gather requirements of software development industry.

#### **2.2.1 Interviews**

Clarification of the problems was done by discussing with the supervisor throughout this project. Thus, Interviews are good methods to gather requirements to develop the system perfectly.

#### **2.2.2 Self-Experience**

All the project assistant tasks are done by me when I was a project assistant of BIT projects. Self-experiences are helped a lot to gather and clarify the requirements.

#### **2.2.3 Observation of the Work Environment**

Observation of the working environment is the one of major points in fact finding methods and is used to gather requirements of the MIS for BIT Final Year Projects.

## **2.3 Current Manual System Analysis**

At present, all the activities are handled by manual system without facilitating the submission and providing project guidelines and templates. It does not provide a place to collect all the project relevant details in to one location. Further, it is consuming time to create the schedules in manually. Also current system does not have a proper method to manage students' details with their submissions.

### **Submit Document**

- Student can submit documents. Further it can be extended as upload documents and download documents.

### **Forum Discussion**

- Teacher and students can join with the forum discussion if they want.

### **Submission**

- Teacher can activate submission links and deactivate submission links referencing their timelines.

### **Download Files**

- Teacher can download the submission as bulk or individual submission for evaluation process

### **Upload Files**

- Teacher can also upload templates, guidelines to offer awareness based on the requirements of the students.

### **Create Group of Students**

- Teacher can create group of students based on the purpose of having groups.

### **Manage Users**

- Administrator of the system can manage the users such as students and the teachers who are the main users in the system. Admin can enroll users and delete users

### **View Submission as Event Wise**

- Each and every submission are viewed as an event. All submissions cannot be viewed on the page with the all previous submissions in students' point of view.



## Manage Users

- Users are managed by admin as all the types of systems.

## Set Privileges

- Admin can create privileges for the users

## Create Separate Courses

- The current system is not for the one course. It consists with the several course modules. Admin can create separate courses and enroll users as their qualifications.

Figure 2.1 represents the high-level architecture of the current system using use case diagram as above explicated.

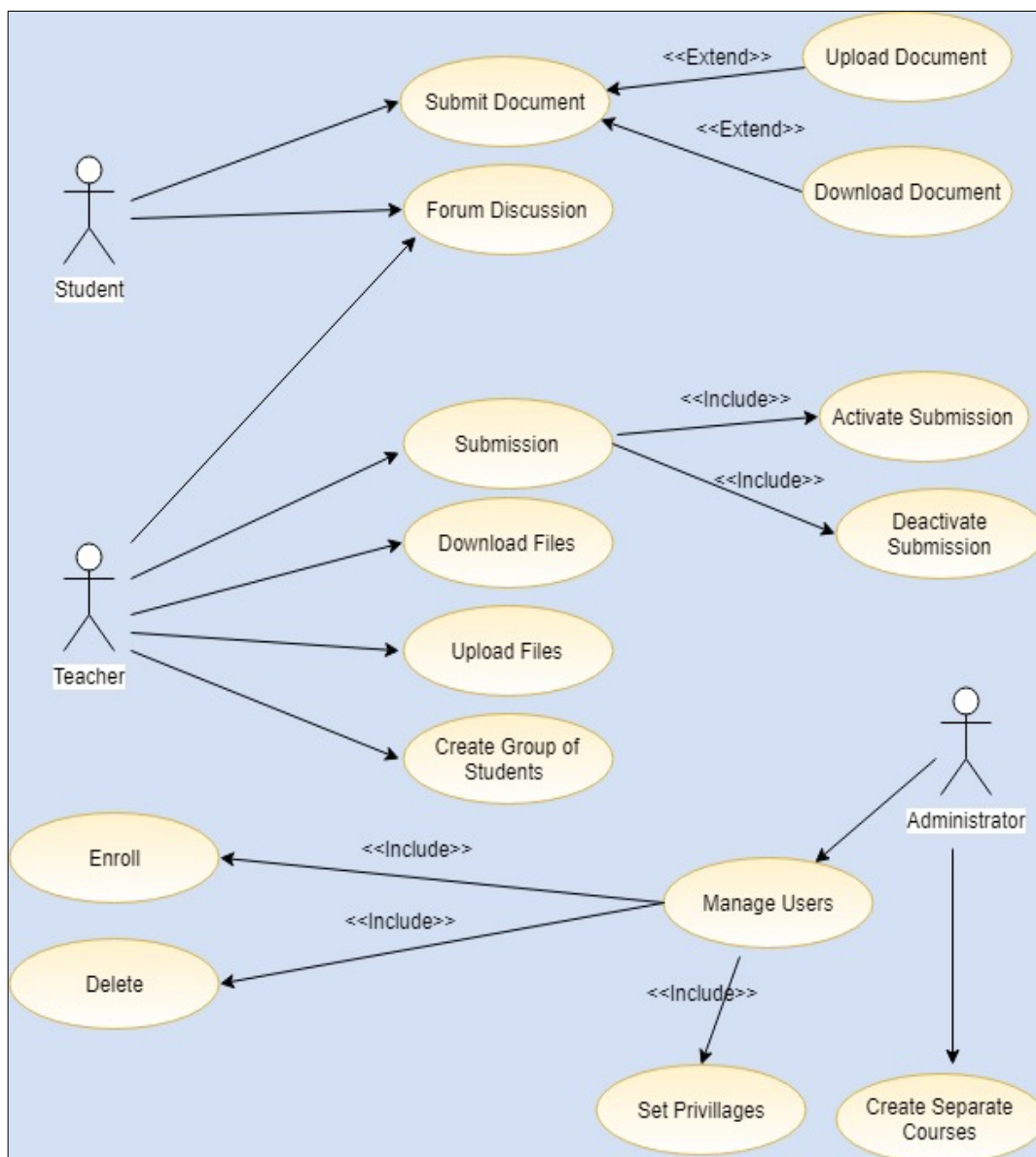


Figure 2.1: Current manual system

## 2.4 Review of the Similar Works

Following are the few similar systems which have reviewed for building a successful system with knowledge of existing work.

### 2.4.1 Virtual Learning Environment-University of Colombo, School of Computing

UCSC VLE is an online system which is designed to support teaching and learning activities through the Internet [2]. Using this VLE all can access resources (teaching materials, past papers etc.) to learn the courses online. In addition, all can connect with other facilitators through the forums and chat sessions. Virtual Learning Environment-UCSC is depicted in Figure 2.2.

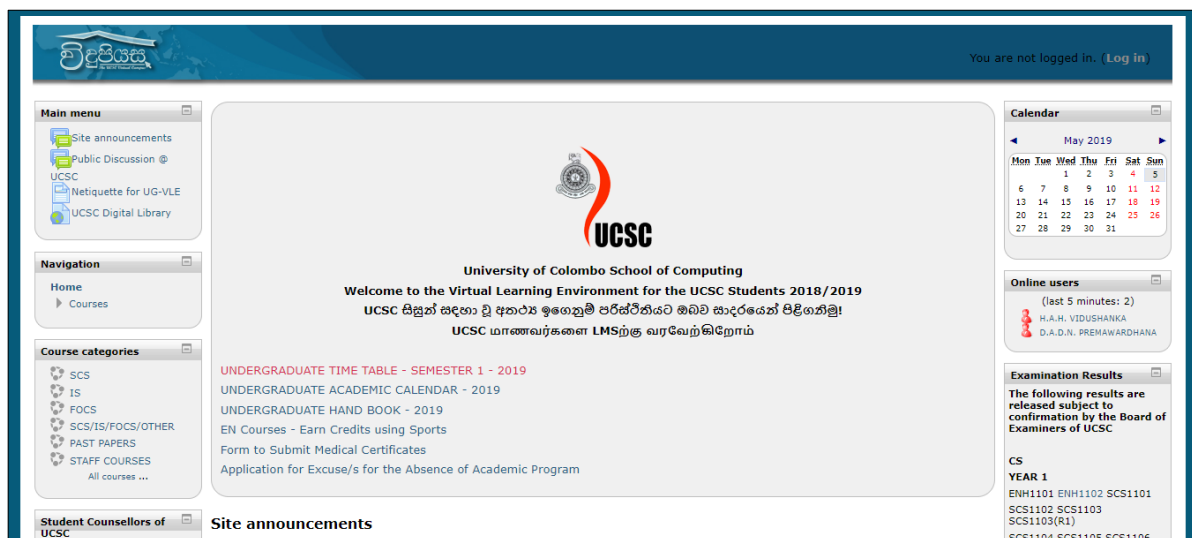


Figure 2.2: Virtual Learning Environment - UCSC

### 2.4.2 Computerized Assignment/Project Submission System For F.S.K.T.M (PHILLOMEANA ANAK LIMBANG)

This project is about designing and developing a client-server system to computerize assignment/project submission process in FSKTM. The system is able to connect to the email server of lecturers and students to alert them on the current status of the submission process.

Following tasks are done by the above system [3].

- Submission management
- Notification handling
- Chatting
- Tracking progress of the submissions

### 2.4.3 VLE- Faculty of Management and Finance, University of Colombo

This is also an online system which is designed to help for teaching and learning activities through the Internet for the lecturers and students in the faculty of management and finance [4], University of Colombo. Figure 2.3 depicts the VLE, FOMF.

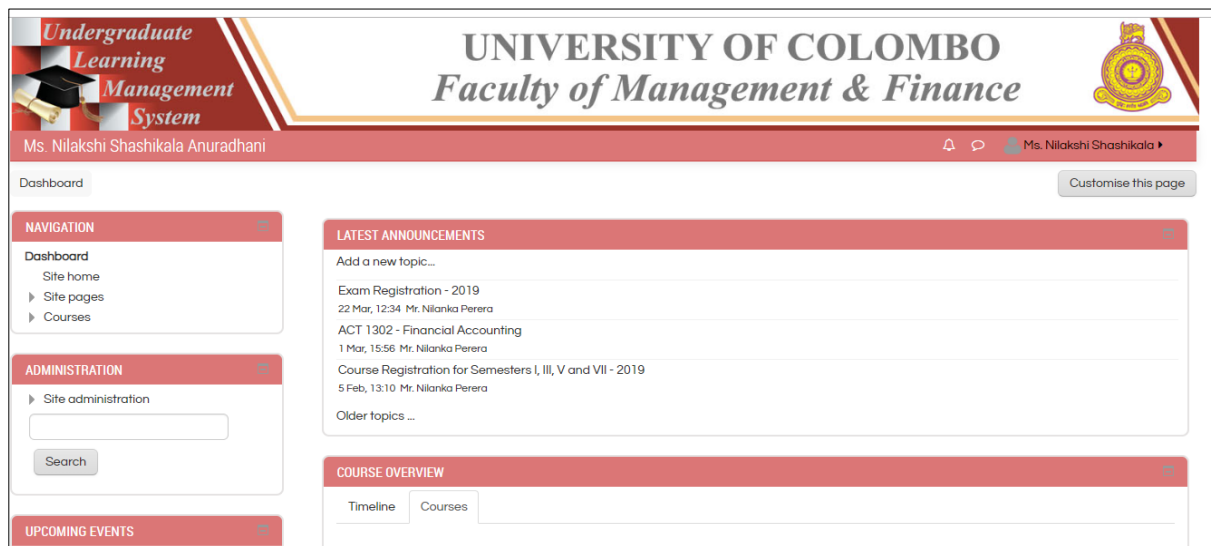


Figure 2.3: Virtual Learning Environment, FOMF

## 2.4.4 VLE, IHRA, University of Colombo

This is also a virtual learning environment created using Moodle. All the notices are published on the VLE and notices can be seen without log in to the system[5]. It is linked with the UOC library. FIGURE 2.4 depicts the Virtual Learning Environment, IHRA

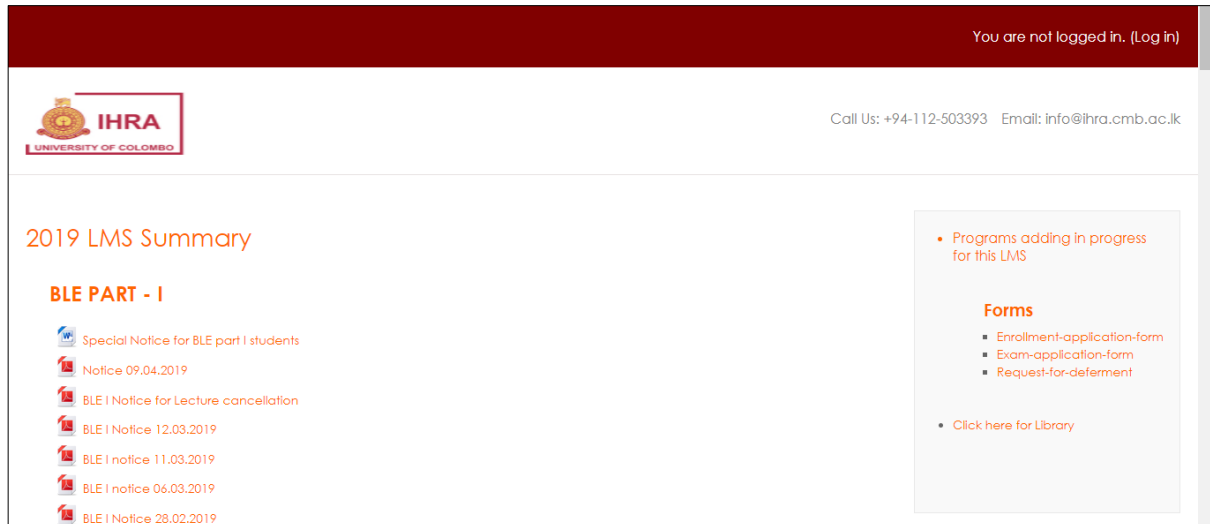


Figure 2.4: Virtual Learning Environment, IHRA

## 2.4.5 Learning Organizer, University of Moratuwa

Learning organizer is a document management system created using Moodle [6]. Mainly, all the submissions are managed by this system and Especially SMS facility is integrated with this MIS platform. Figure 2.5 depicts the learn.org

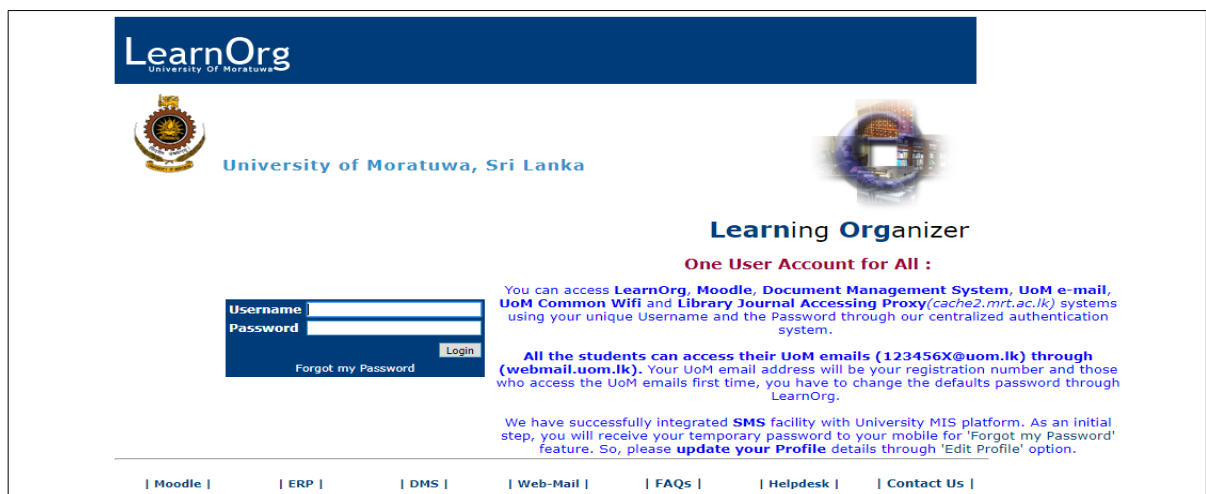


Figure 2.5: Learn.org.UOM

## 2.5 Similar Systems Comparison Matrix

With comparing similar systems with the proposed system is depicted using the table 2.4

	UCSC VLE	Computerized assignment/project submission system For F.S.K.T.M	MGT VLE	IHRA VLE	MRT VLE	Proposed System
Submission management	✓	✓	✓	✓	✓	✓
Notification handling	✓	✓	✓	✓	✓	✓
Chatting	✓	✓	✓	✓	✓	✓
Tracking progress of the submissions		✓				✓
Send reminders for deadlines						✓
Upload templates and guidelines	✓	✓	✓	✓	✓	✓
Schedule the evaluation dates						✓
Send evaluation dates as Notifications					✓	✓
Add restrictions to the students' submissions						✓
Integrated with SMS components					✓	

Table 2.1: Similar System Comparison Matrix

## **2.6 Comparison of Alternative Design Strategies**

Alternative solutions can be categorized into various subtopics

### **Maintain system with old method**

By choosing this method the coordinator should waste a lot of time to access the students' data and filter out the necessary details of the submissions.

### **Plugin development to the VLE**

Developing a plugin for a LMS is not an easy task. It is included kind of research work. More effort is needed to implement the plugin for the VLE.

### **System as a web solution**

Implementing a web-based solution is more beneficial than deploying a plugin for the VLE. The cost of software and the effort of the developer can be reduced by the implementation of a web-based solution. Also, Web-based systems are not platform dependent and any different platform can be used. Access (24\*7) to the system can be very easy and no need to be in UCSC premises to do an activity. When comparing with the plugin for the VLE and web-based system is easy to deploy.

## **2.7 Functional Requirements**

### **Create Logins**

- Admin can create logins for students. The students are not allowed to create accounts by themselves.
- Admin can create login for the project coordinator and the assistant

### **Submit Reports/ Templates**

- Students can submit reports as MS Word documents or PDF documents.
- Admin can upload guidelines and templates
- Students can view guidelines and templates as their relevance.
- Admin can activate and deactivate the submission importance of the deadline. If a student has requested to extend the deadline for fair reasons, the system is flexible enough to extend the deadline.

### **Generate Reports**

- System provides an option for generating reports.

### **Communication**

- Students have queries when they are doing projects. So, students can chat with the coordinator to get solutions to those queries.
- The coordinator can participate in the chat and keep the history of chatting.

### **Send Notifications**

- When admin sets the submission link, students can see the significance notification with reminders.
- When coordinator schedules date for viva and dissertation evaluation, the student can get appropriate dates for the evaluation process.

### **Create Eligibility List**

- Admin can create the final eligibility list for the evaluation process.

### **Schedule the Viva and Dissertation Evaluation**

- The coordinator can schedule viva and dissertation evaluation for the list of eligible students.

## **2.8 Non-Functional Requirements**

According to the build system, Identified non-functional requirements are given below.

- **Reliability**

The system should be reliable. That is, it should provide correct output belongs to correct inputs. It should not provide redundant data. Accuracy of the results should be confirmed by the system.

- **User Friendliness**

The system should be more user-friendly. It should contain interfaces that are more convenient and easy to handle.

- **Performance**

All pages should load within 5-60 seconds at any load condition.

- **Availability**

Information should be available whenever the user (student, coordinator, Assistant) need.

- **Security**

The system should be protected against unintended and inadvertent correction of information by the user himself.



## **Chapter 3:Design**

### **3.1 Introduction**

Design Phase is the hardest and important part of the software development lifecycle. This phase provides the detail and description of the system, including components, interface, modules, architecture, and data, for the functional and nonfunctional requirements gathered in the analysis phase. The goal of the design process is to produce a model or representation of a system, which can be used later to build that system [7].

### **3.2 Use Case Diagram for the Proposed System**

A UML high level essential use case is a brief description of the main processes used to accomplish the system functions. In here, Use cases are depicted according to the user levels.

#### **Submit Documents**

- Students can submit their submission and coordinators can submit the templates and guidelines to fill users' requirements
- Students and coordinators can download files as their necessity

#### **Chat**

- Chat facility is implemented and history of the chat is kept to create a connection between users.

#### **Manage Students**

- Coordinator can manage the students including enroll, delete, categorize and filter.

#### **Submission**

- Coordinator and Assistant can activate and deactivate the submission links

#### **Send Notifications**

- Coordinator and assistant can send the notifications for deadlines and important news

## Generate Reports

- Coordinator and assistant can generate reports to get students submission details, students' details, eligible list to complete the project and final eligibility list to submit their project.

## View all Submissions as Students' Point of View

- In the current system there is no way to see all the submissions in a one page with corresponding submission dates. However the implemented system provides facilities to see all the submissions on the students profile as their relevance.

Following figures are depicted to show how the submissions are seen by the students and teachers in event point of view in the considering the current system.

Following figure 3.1 shows the assignment 1 submission to the students. However, if students want to see all the submissions they have done, all the submissions cannot be viewed on one page. Students have to go through the event by event to see the submissions.



Assignment 1	
 Assignment 01.docx	
<b>Submission status</b>	
Submission status	Submitted for grading
Grading status	Not graded
Due date	Tuesday, 26 February 2019, 6:00 PM
Time remaining	Assignment was submitted 2 days 3 hours early
Last modified	Sunday, 24 February 2019, 2:32 PM
File submissions	 16550043.rar
Submission comments	<a href="#">Comments (0)</a>
<div><a href="#">Edit submission</a></div> <div>Make changes to your submission</div>	

Figure 3.1: Submission 1, event point of view (UCSC)

Following figure 3.2 shows the assignment 2 submission view to the students. This is a one of the interfaces of the UCSC VLE.

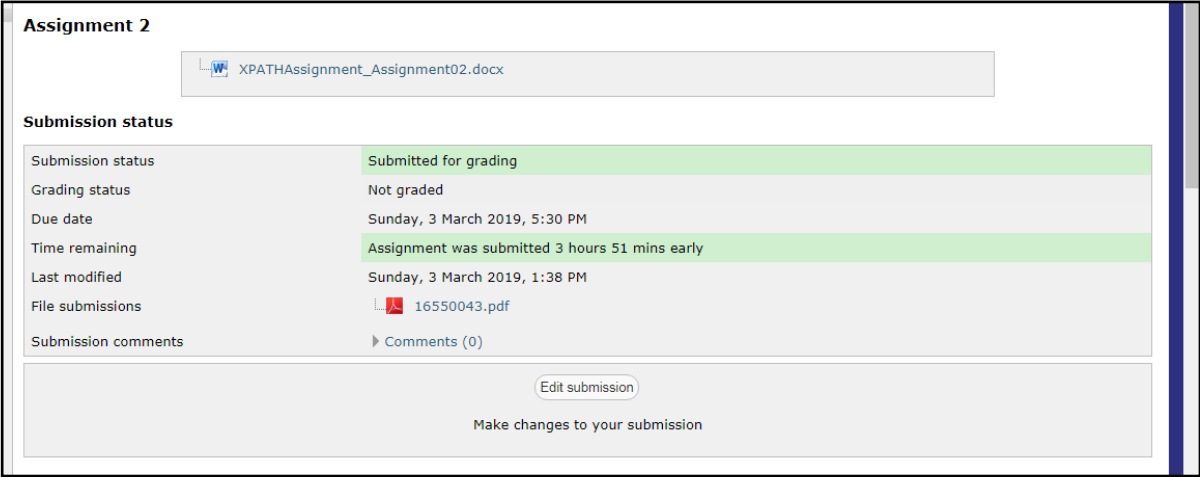


Figure 3.2: Submission 2, event point of view (UCSC)

However, the VLE of the UCSC is helped to view grades for all the submissions at once to the students. Figure 3.3 is depicted the all submission name with grades.

The screenshot shows a 'User report' for 'MIT4201'. It contains a table with columns: Grade item, Calculated weight, Grade, Range, Percentage, Feedback, and Contribution to course total. The table lists 'Assignment 1', 'Assignment 2', and 'Course total' with their respective grades and weights.

Grade item	Calculated weight	Grade	Range	Percentage	Feedback	Contribution to course total
MIT4201 Software Quality Assurance						
Assignment 1	0.00 % ( Empty )	-	0-100	-		0.00 %
Assignment 2	0.00 % ( Empty )	-	0-100	-		0.00 %
Course total	-	-	0-100	-		-

Figure 3.3: All submission name with grades (UCSC)

According to the given references, Faculty of management, UOC follows the same process as above mentioned. This interface (Figure 3.4) is helped to show the group assignment submission

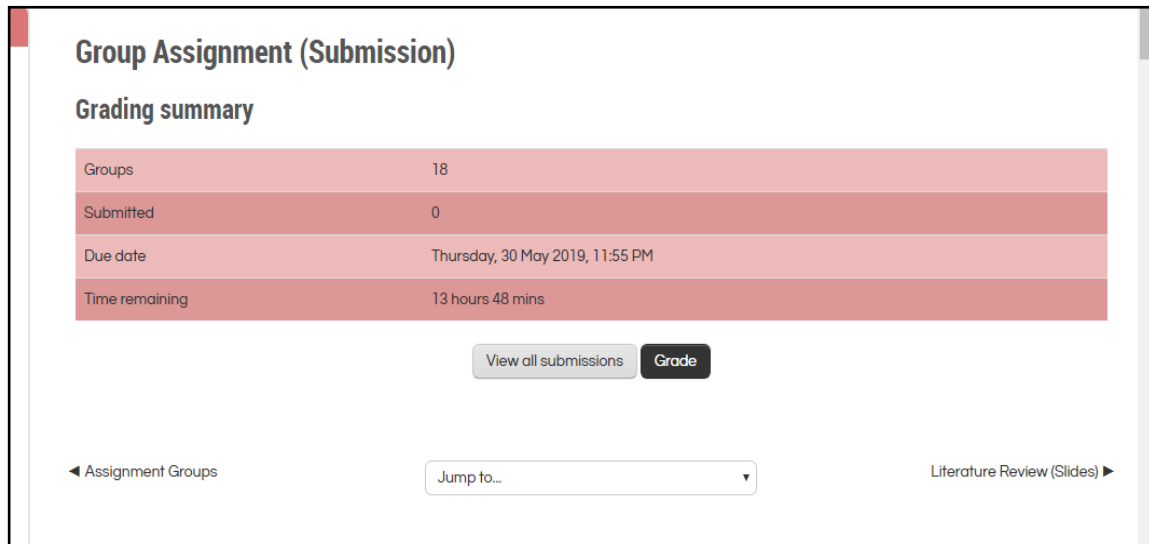


Figure 3.4: Submission, view point of view (FMF, UOC)

Figure 3.5 is depicted another submission as event point of view.

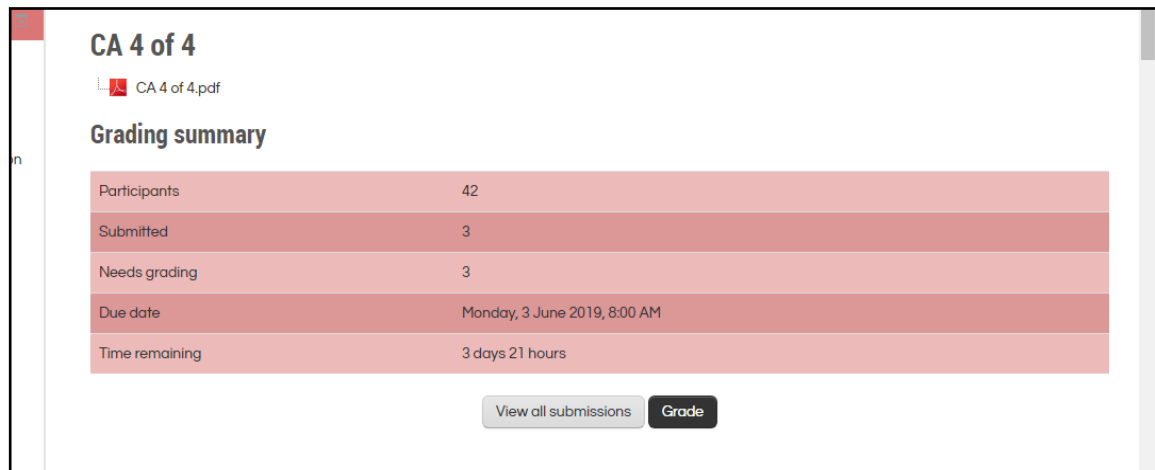


Figure 3.5: Submission 2, Event point of view (FMF, UOC)

### Schedule Evaluation Process

- Scheduling evaluation process is completed using the implemented system with considering the eligibility of the students.

### Send Scheduling Dates

- Sending final evaluation schedule details as reminders is the important module done by the system.

### Create Schedules

- Coordinator can create schedule filtering the students according to the process going on.

The use case diagram for the proposed system is shown below in Figure 3.6.

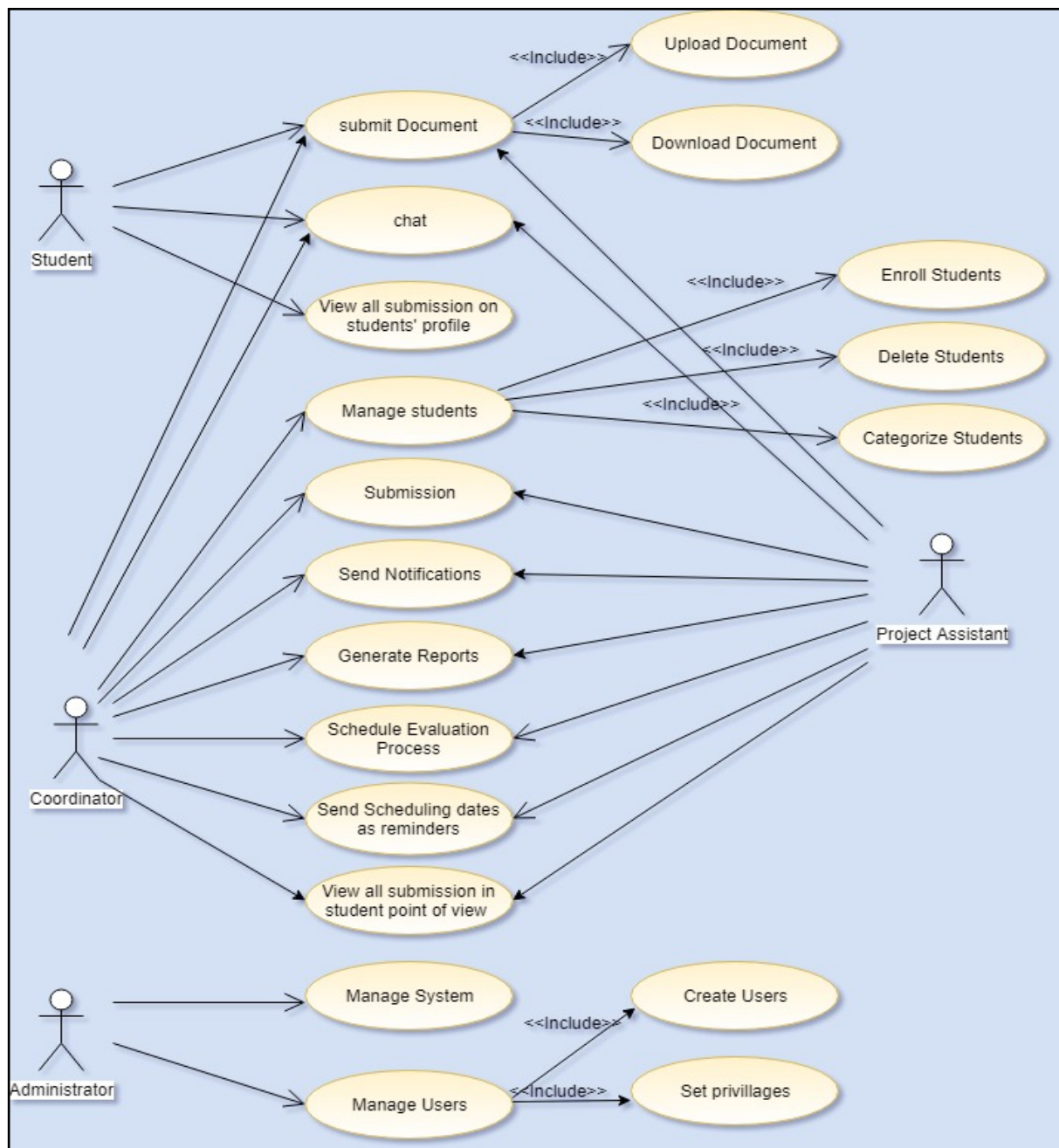


Figure 3.6: Use Case Diagram for the Proposed System

### 3.3 Methodology and Implementation Considerations

Methodology and implementation considerations are explained in this section. RUP model was selected to implement the proposed system, because the requirements are based on the feedbacks and the desires of the coordinator, which can get changed during the lifecycle. And the language of modeling in the common knowledge base is the Unified Modeling Language (UML). This is iterative development process.

The last iteration delivers on that the set of iteration objectives must be exactly same for the set of goals specified by the coordinator, that is, all requirements must be met.

The most suitable system type is web-based one for this project. There are so many advantages in the web-based system to my project:

- Access to a Web Based System can be gained from any location as long as the user has access to a web browser and an internet connection.
- No need to separately install on each of the end users workstations.
- It is also easily customizable

Therefore the proposed system will be deployed as a web-based application.

To develop web based system, used technologies are in the following.

- My SQL
- Navicat Tool
- Html
- CSS
- Java Script
- PHP
- Bootstrap

Proposed system architecture is **client server architecture**. According to that architecture, central authorization can be kept as all the data is stored on server. And it is easy to make a back-up of it as well. It shows in Figure 3.7.

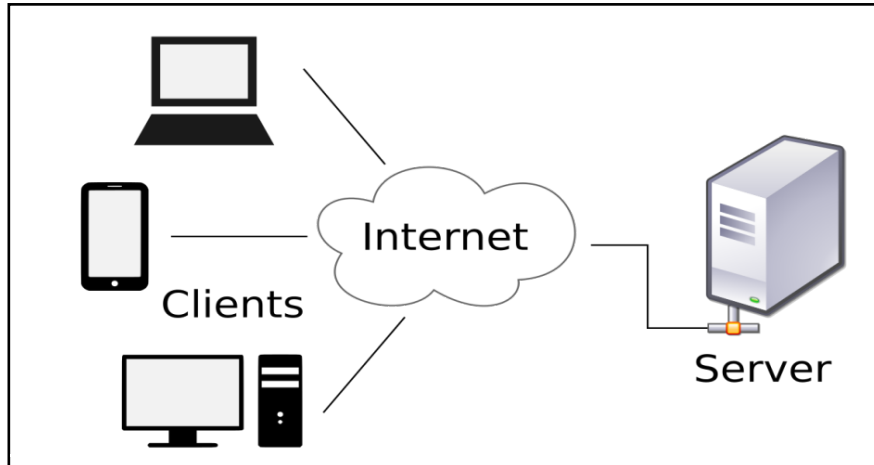


Figure 3.7: Client Server Architecture

### 3.4 The flow of the Interactions

The flow of the 1<sup>st</sup> interaction with the system is given below in Figure 3.8.

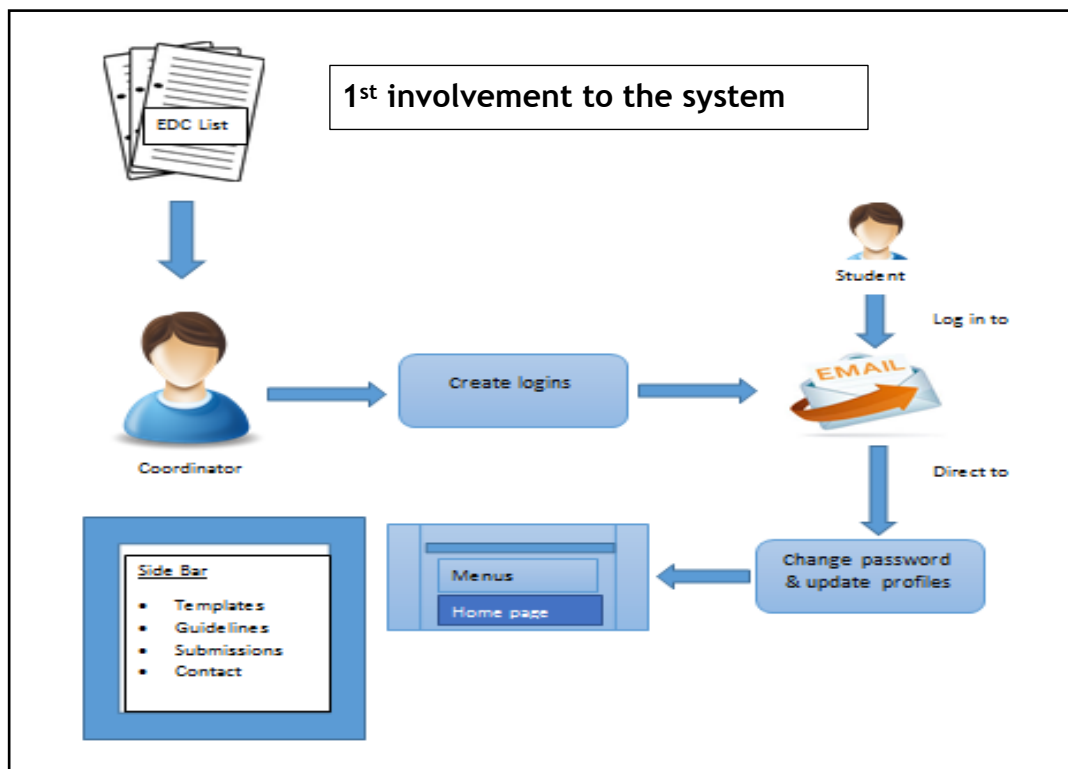


Figure 3.8: Flow of the 1st Interaction with the System

### 3.5 Data Base Design for the Proposed System

Database design is the process of producing a detailed data model of a database. This logical data model contains all the needed logical and physical design choices and physical storage parameters needed to generate a design in a data definition language, which can then be used to create a database.

Database design for proposed system is displayed by Figure 3.9.

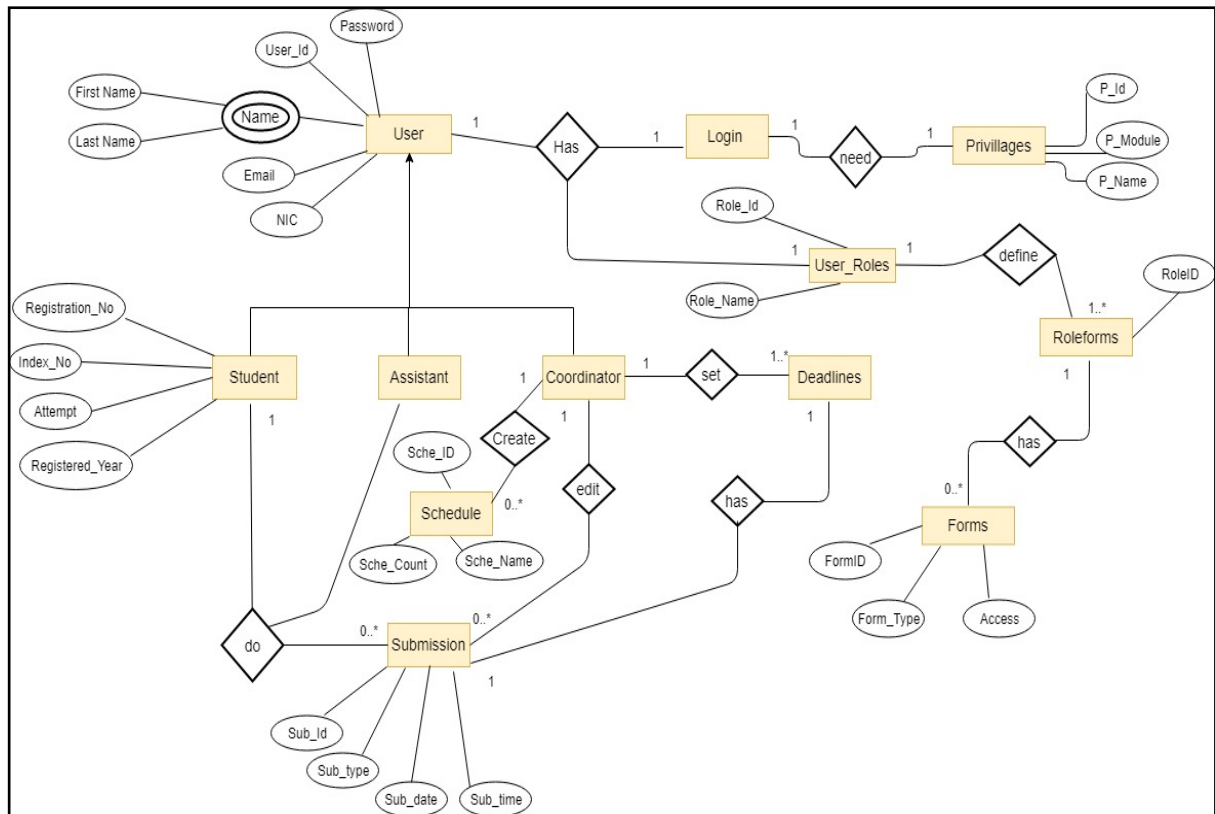


Figure 3.9: ER diagram for the Proposed System



### 3.6 Class Diagram for the Proposed System

“The class diagram is a static diagram. It represents the static view of an application. Class diagram is not only used for visualizing, describing and documenting different aspects of a system but also for constructing executable code of the software application.

The class diagram describes the attributes and operations of a class and also the constraints imposed on the system. The class diagrams are widely used in the modeling of object oriented systems because they are the only UML diagrams which can be mapped directly with object oriented languages” [8].

Class Diagram for the School Management System is depicted in following Figure 3.10.

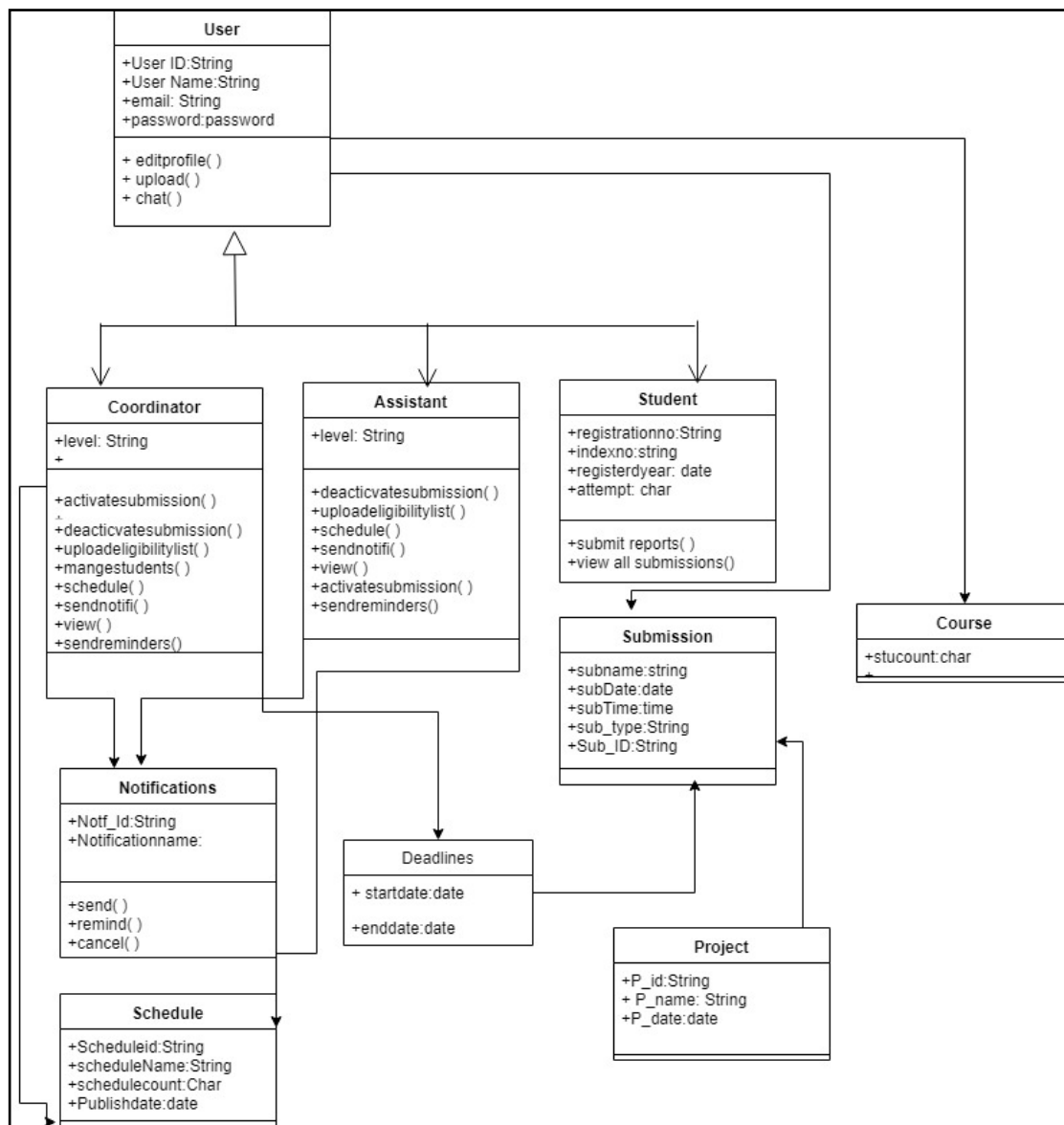


Figure 3.10: Class Diagram for Proposed System

### **3.7 User Interfaces for the Proposed System**

User interface design is an essential part of the overall system design process as interface is the layer that the users are directly interacting with. Well-designed user interface supports user and user training and will decrease the cost. Therefore accomplishment of good user interface will be a basic feature of the design.

- **Clarity**

Clarity is most imported part of user interface design. Task of the system should be certainly shown to the user. If the user cannot identify how this task work and what will done for next step.

- **Attractive**

All user interfaces should be simple and easy to use. This system uses common format, font styles and font size and menu type which will be attractive to the user.

- **Efficiency**

Good interface design should have to be implemented to make the functions of the system faster. An effective system is mostly valuable for users to do their job faster and accurately.

Consistent

Buttons, tabs, icons and other interface elements will be consistent across the entire system.

- **Forgiveness**

This part will be archived in a way to avoid the mistakes done by users. If some information was deleted, then the system should be able to recover that situation. Provision of an Undo button and allowing recovering previous status, displaying a button and message windows would be beneficial for the system.

- **Common format**

All the UI components should have to common format. Such as texts, icons, textboxes.

- **Responsive Design**

It will help to view system pages execute in different size screens like desktop computer, smart phones, and tabs

### 3.7.1 Log in Interface

This is the part where the main confirmation of access to the system is generated. User should logging to their account with provided user name and the password. Figure 3.11 shows the logging interface of the management Information system for BIT final year projects.

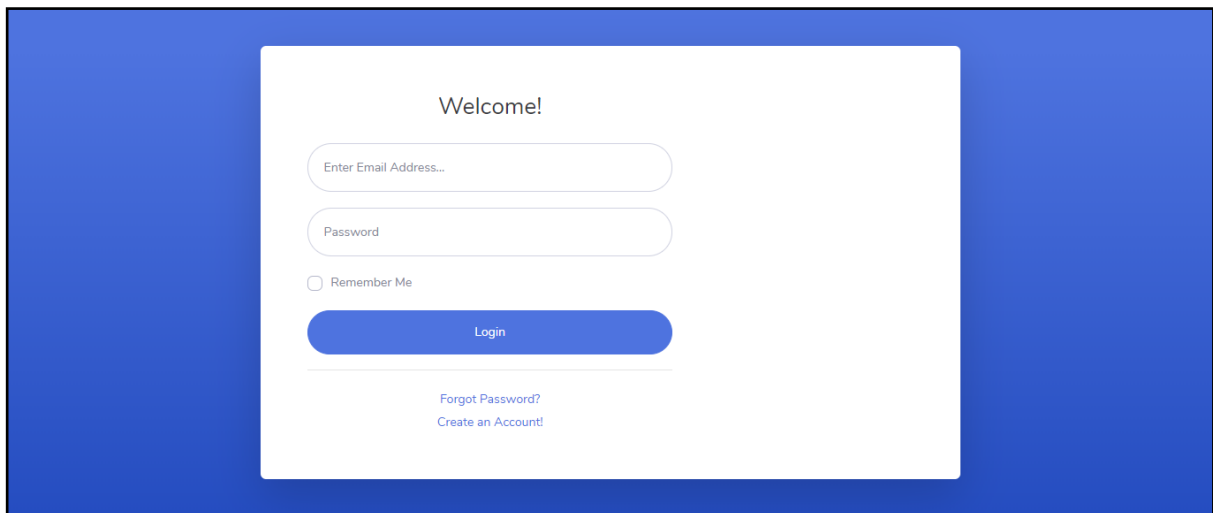


Figure 3.11: Logging interface

### 3.7.2 Student Dashboard

When a student login to the system, the first interface that they are interacting with is the dashboard. The following picture in Figure 3.12 depicts the dashboard page of the proposed system.

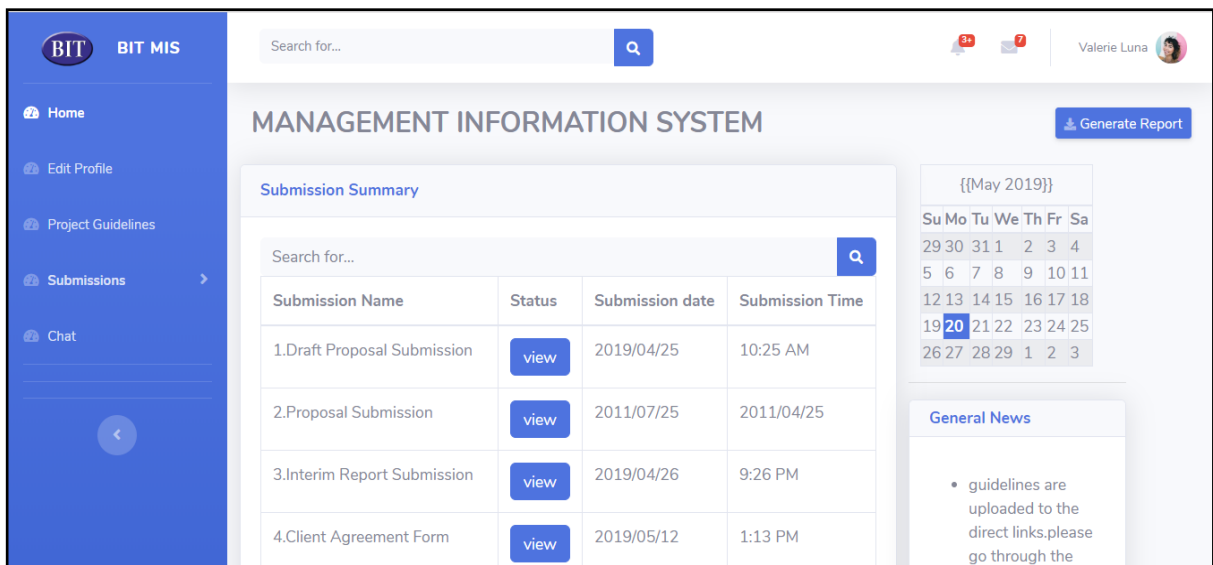


Figure 3.12: 1st Interacting dashboard

### 3.7.3 Coordinator Dashboard

When the coordinator login to the system, the first interface that the coordinator is interacting with the below Figure 3.13 dashboard.

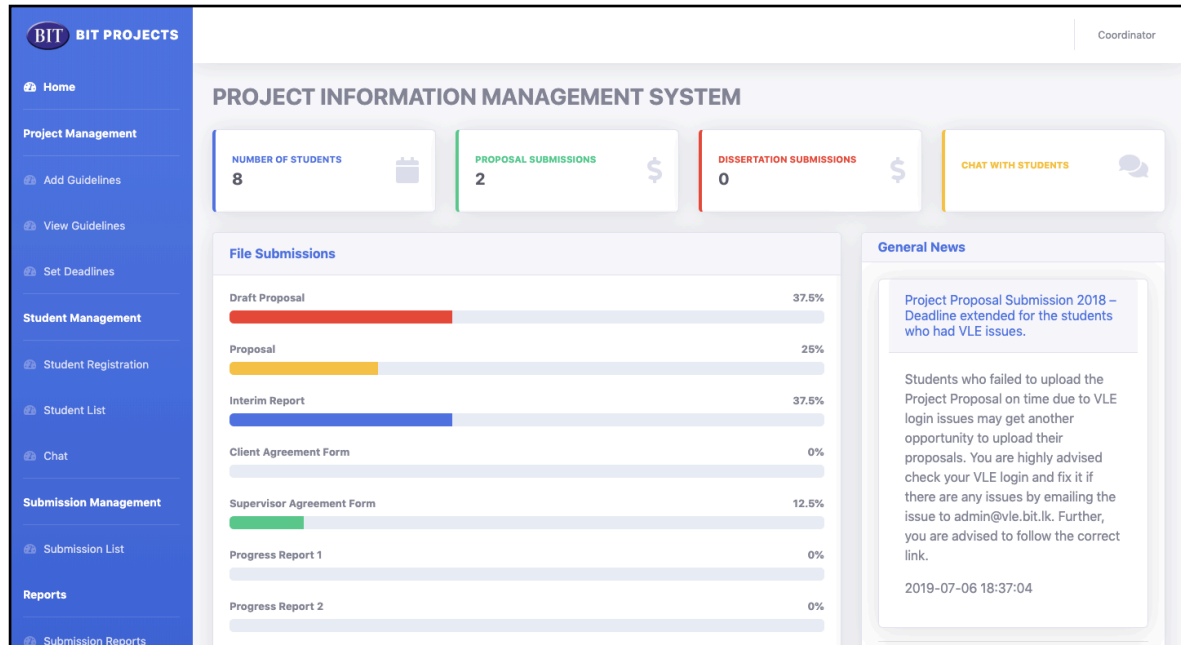


Figure 3.13: Coordinator Dashboard

### 3.7.4 Edit Profile

When a student needs to edit their profile, this interface (Figure 3.14) is helped to edit their details.

The screenshot shows the 'Edit Profile' interface for a student in the 'BIT MIS' system. The top header includes the 'BIT MIS' logo, a search bar, notification icons, and the user's name 'Valerie Luna' with a profile picture. The left sidebar menu contains 'Home', 'Edit Profile' (which is highlighted), 'Project Guidelines', 'Submissions', and 'Chat'. The main content area is titled 'Profile' and contains a 'Student Profile' form with the following fields: 'First Name', 'Last Name', 'NIC', 'Attempt', 'Email Address', 'Registration Number', and 'Index Number'. A blue 'Save' button is partially visible at the bottom left of the form area.

Figure 3.14: Edit student profile

### 3.7.5 Guidelines and Templates

When coordinator log into the system, the place to insert guidelines and upload templates for the students. Figure 3.15 depicts the inserting guidelines and uploading templates to the system.

BIT MIS

Search for...

Valerie Luna

## Guidelines & Templates

Insert Guidelines

Submit

Upload Templates

Select Template to Upload: Choose File No file chosen Upload File

Copyright © Your Website 2019

Figure 3.15: Insert guidelines and upload templates

Figure 3.16 depicts the view interface of the guidelines on the system.

BIT PROJECTS

Coordinator

## Guidelines

Heading	Guidelines	Template	Update	Delete
Sample Proposal	Refer the pdf	Sample Proposal.pdf	Update	Delete
Viva Evaluation Form-2019	see the attached file	Viva Evaluation Form-2019.pdf	Update	Delete
Evaluation Criteria	The marking scheme for the project is as shown using the attached pdf	Evaluation Criteria.pdf	Update	Delete
IT6105-Project Guidelines	IT6105 - Software Development Project Guidelines are attached herewith.	IT6105-Project Guidelines.pdf	Update	Delete

Copyright © BIT Projects 2019

Figure 3.16: View Guidelines

### 3.7.6 Insert eligible list

This system provide a chance to upload eligible list at once including more than 500 records as a format of excel. Figure 3.17 depicts the interface for uploading the eligibility list.

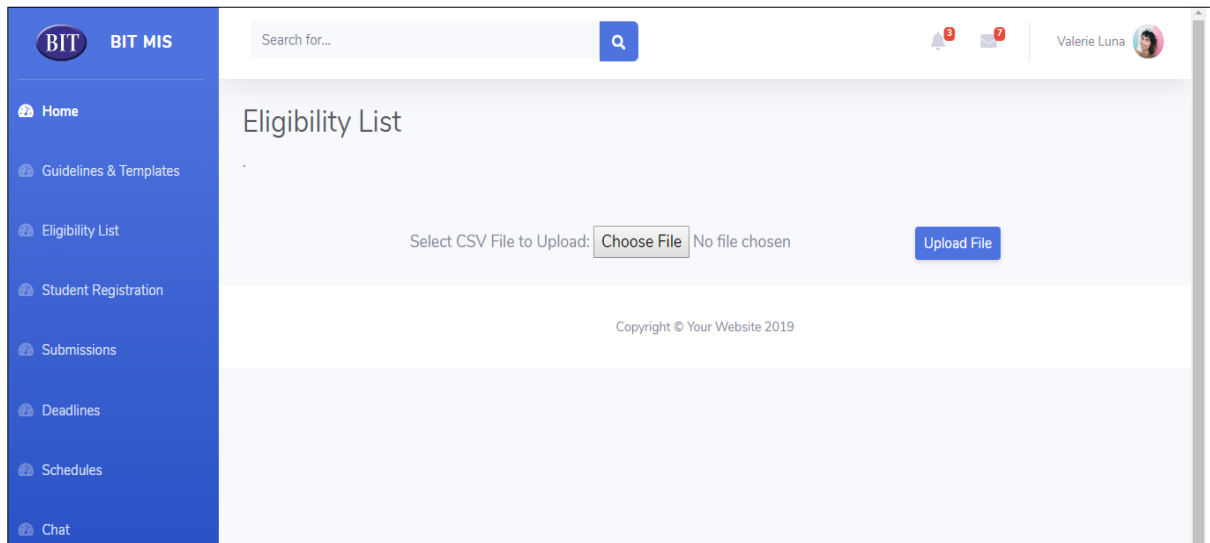


Figure 3.17: Create eligibility list

### 3.7.7 Student Registration

Students are registered to the system using CSV file uploading. The following user interface is helped to register students who missed their names in eligibility list with the fair reason considering the process. Figure 3.18 shows the student registration interface.

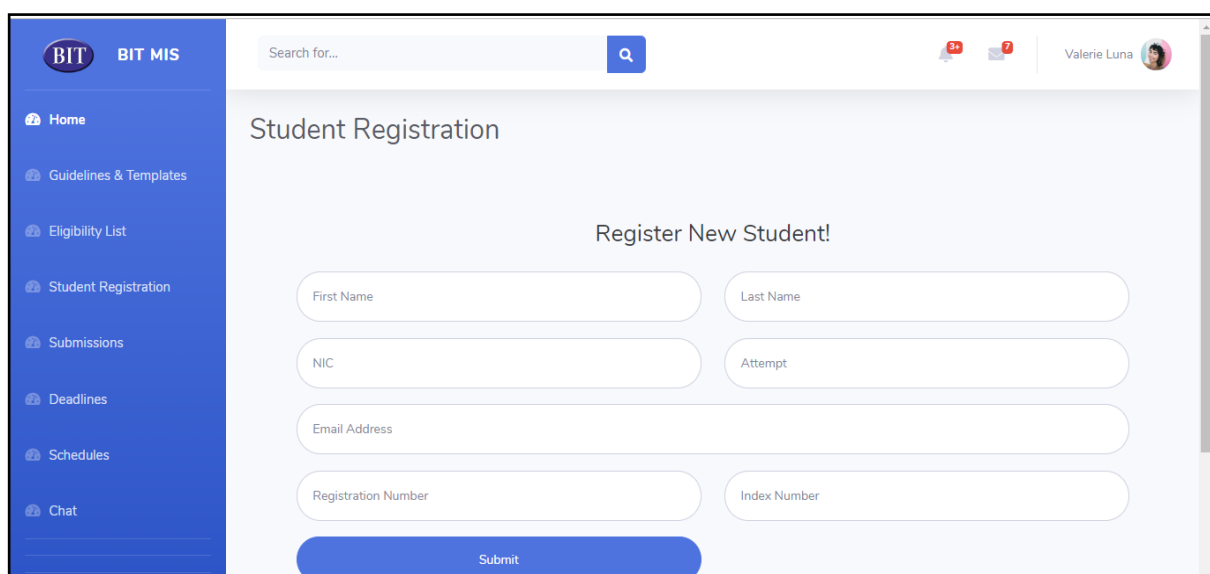


Figure 3.18: Student Registration

## **Chapter 4:Implementation**

### **4.1 Introduction**

The system is developed according to the detailed design specification as required by the Supervisor of the project. Both front-end and back-end coding is necessary in development. A lot of time is consumed at this level for the same reason that high-level code is utilized. When rework is involved in events such as code modification or corrections, it is good practice to have used comments to describe particular code segments. Form validation is a mandatory requirement as well.

### **4.2 Development Environment**

#### **4.2.1 Hardware Requirements**

The system was developed on a machine with the following configuration:

- Processor: 1.6Ghz (Intel core i5)
- RAM: 4 GB
- Hard disk: 1GB

#### **4.2.2 Software Requirements**

The following software is required for the system development.

- Microsoft Windows 7 Professional or above
- MySQL
- XAMMP Server
- Navicat

#### **4.2.3 Technologies and Languages used**

##### **Apache Server**

The web server used for the project was Apache Server. It is open source software which is fast, reliable and secure [9].

##### **MySQL**

The relational database management system Oracle MySQL was used to handle database communications with the application [10].

## Navicat

Navicat for MySQL is the ideal solution for MySQL/MariaDB administration and development. It is a single application that allows you to connect to MySQL[11].

## Netbeans IDE 8.0

Coding the system was done with the Netbeans IDE which is a very popular integrated development environment. A set of software components called modules are allowed to be developed through this platform; an advantage of the IDE being cross-platform compatible so that the code runs on all Microsoft Windows, Max OS X, Linux, Solaris and other platforms supporting the use of a compatible Java Virtual Machine (JVM) [12].

## PHP

Object Oriented PHP was used as the programming language in developing the system. It is also cross-platform software which is open source and free to develop, deploy and use as required. PHP is specially designed for the web; it is powerful, scalable and robust. The language can be object oriented or non-object oriented depending on the functionality, although OOP was considered for the project.

### 4.3 Code structure to connect the database

To access and work with the database, a database connection should be established upfront. A common folder contains the code for it facilitating the inclusion of the connection wherever necessary. Following Figure 23 is shown the code structure to connect the database.



Figure 4.1: Code structure to Connect Database



## **4.4 Reused Components**

### **Header and Footer of students' view**

Header and Footer sections are same in every webpage except some elements. Therefore, Header and Footer components were written in separate files and they are called in every page of the system. Elements of them dynamically change as the loaded page.

### **Header and Footer of every panel of Admin console and User console**

Header and Footer sections are same in every panel of each console except some elements. Therefore, Header and Footer components were written in separate files and they are called in every panel of each console. Elements of them dynamically change as the loaded panel.

### **Side Navigation of every panel of Admin console**

Side Navigation structure is same in every panel of admin console. Therefore, Side Navigation was written in separate files and it is called in every panel of admin console. Tabs of the navigation dynamically change as the loaded panel.

### **Side Navigation of every panel of Student, coordinator and assistant console.**

Side Navigation structure is same in every panel of user console. Therefore, Side Navigation was written in separate files and it is called in every panel of user console. Tabs of the navigation are dynamically changed as the loaded panel.

## **Chapter 5 : Evaluation**

### **5.1 Introduction**

Evaluation is the most critical phase in software development life cycle. It reviews the entire system works with many aspects and determines system quality and achieves the objectives of the project. According to my project, Evaluation process is done as follows.

### **5.2 Manual Test Plan**

- Perform a white-box testing to test the system structure of each program. Then perform a black box testing to test the functionality of the program, at the level of unit-testing.
- According to the system structure and standards in each system user interface define test cases with sample test data with expected output. Those are used to compare with the actual results after the execution of the test case.
- Perform an integration test and a system test after integration of the system modules to check whether the critical functionalities are successfully covered.
- Confirm that all the functional and non-functional requirements have been met with the developed system.
- Conducting an acceptance test at the client site before handover.

#### **5.2.1 Test cases**

A test case is a document, which has a set of test data, preconditions, expected results and post conditions, developed for a particular test scenario in order to verify compliance against a specific requirement. There are essentially two main approaches to designing test cases.

- Black Box approach

It is a testing, either functional or non-functional, lacking reference to the internal structure of the component or system [13].

- White Box approach

It is a testing based on an analysis of the internal structure of the component or system. [14].

## Login Module

Test ID	Test Case	Expected Result	Pass/ Fail
LM 1	Give incorrect user name and incorrect password	Unable to log in to the system	Passed
LM 2	Give incorrect user name and correct password	Unable to log in to the system	Passed
LM 3	Give correct user name and incorrect password	Unable to log in to the system	Passed
LM 4	Give correct user name and correct password	Able to log in to the system	Passed
LM 5	Try to log using 5 times using wrong user details	Unable to log in to the system	Passed
LM 6	Leave all fields empty and then click on “login” button	Show Error message as “Please enter user name and password to login”	Passed
LM 7	Leave user name field empty and then click on “login” button	Show Error message as “Please enter user name and password to login”	Passed
LM 8	Leave password field empty and then click on “login” button	Show Error message as “Please enter password to login”	Passed

Table 5.1: Test cases for the Logging Module

### Submission Module for Students

Test ID	Test Case	Expected Result	Pass/ Fail
SM 1	Upload word document and click submit button	Able to Upload word document	Passed
SM 2	Upload pdf document and click submit button	Able to Upload pdf document	Passed
SM 3	Upload PowerPoint and click submit button	Able to Upload PowerPoint	Passed
SM 4	Upload Image file and click submit button	Able to Upload image file (PNG,JPEG,JPG)	Passed
SM 5	Upload excel, CSV file and click the submit button	Unable to such a format. Send a message “file format should be required form”.	Passed

Table 5.2: Test cases for the Submission Module for Students

### Submission module for the Coordinator and Assistant

Test ID	Test Case	Expected Result	Pass/ Fail
SM-CA 1	Upload word document and click submit button	Able to Upload word document	Passed
SM-CA 2	Upload pdf document and click submit button	Able to Upload word document	Passed
SM-CA 3	Upload PowerPoint and click submit button	Able to Upload word document	Passed
SM-CA 4	Upload Image file and click submit button	Able to Upload word document	Passed
SM-CA 5	Upload excel, csv file and click the submit button	Able to Upload excel format document	Passed
SM-CA 6	Click activate submission button and set the previous date as deadlines	Unable to set the submission date as previous	Passed
SM-CA 7	Click activate submission button and set the today as deadlines	Able to set the submission date as Current date	Passed
SM-CA 8	Click activate submission button and set the future date as deadlines	Able to set the submission date as future date	Passed
SM-CA 9	Click deactivate submission button	Deactivate the submission	Passed

Table 5.3: Test cases for Submission Module

## Student Enrollment and the student Registration

Test ID	Test Case	Expected Result	Pass/ Fail
SE 1	Upload CSV file to create and view the list of eligibility	Able to Upload excel format document	Passed
SE 2	Upload any other file format to create and view the list of eligibility	Unable to Upload any other file format to the system and display error message.	Passed
SE 3	Fill all the information provided in the student registration form providing incorrect NIC and click submit button.	Unable to register the student. Display the error message as “please use the valid NIC”	Passed
SE 4	Fill all the information provided in the student registration form providing incorrect format of email and click submit button.	Unable to register the student. Display the error message as “please use the valid Email”	Passed
SE 5	Fill all the information provided in the student registration form providing incorrect registration number format and click submit button.	Unable to register the student. Display the error message as “your registration number is incorrect. Please try again”	Passed
SE 6	Fill all the information provided in the student registration form providing incorrect index number format and click submit button.	Unable to register the student. Display the error message as “your Index number is incorrect. Please try again”	Passed
SE 7	Leave Required field empty And click save button	Unable to register the student. Display the message as “please fill the required fields”	Passed

Table 5.4: Test Cases for Student Enrollment and Student Registration

### 5.1.3 Automated Test plan

Katalon Automation Recorder is the best automation recorder that helps to export Selenium WebDriver code. Using this katalon recorder, all the actions are recorded as web elements on web applications, played automated test cases, and done reporting quickly and easily [15].

Katalon is added to the chrome browser as add-on and done the test recording. After completing the recording of the test cases, script is run and tested the system.

Figure 5.1 shows the katalon recording interface of the system.

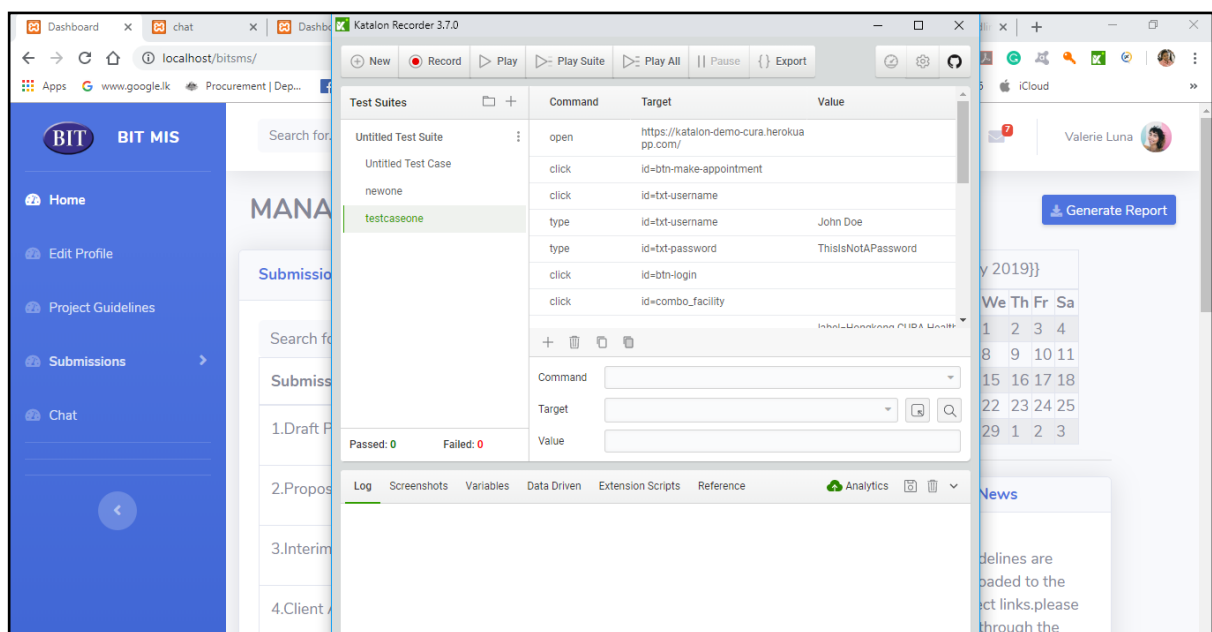


Figure 5.1: Katalone Recorder Interface

## **Chapter 6: Conclusion**

### **6.1 Introduction**

This chapter reviews the works that done throughout the project regarding the challenges faced, mistakes happened and the way corrected them and the knowledge and experience gained. The system was developed to minimize the failures and drawbacks faced with a manual system of the BIT project handling. The existing system was analyzing in the analysis phase. Then the solution was designed using object-oriented techniques. The proposed system was developed in the implementation phase as a web application. Finally, it was evaluated through the testing phase.

### **6.2 Future Improvements**

Following improvements will be added to the System and release it as next version of the System.

- Provide facility to enter the marks at the time of viva
- Make comments on viva
- Send reminders for the examiners to reserve their date for the evaluation process
- Automatic payments calculations for the staff who participated to the evaluation process the payments
- Integrate with the existing VLE

### **6.3 Lessons Learnt**

When submitting the project proposal, and selecting the scope for the project was some kind of critical task I have been achieved. Also, requirement analysis for the selected project was another critical stage of the project. Further, in the implementation process more knowledge acquired about the PHP, JavaScript, Bootstrap coding and MySQL. The most important thing is finally I have been understood about the report writing process and the work in the given schedule is gain proper outstanding knowledge of the system.

The most important lesson I learnt throughout this project work is that the time management as I deviate from the project plan and had to complete the project with a greater difficulty purely due to poor time management. This greater lesson as time management will help me in order to succeed in other matters in my life.



## References

- [1] Ian Sommerville, Software Engineering 7th edition, Addison-Wesley, 2006.
- [2] Virtual Learning Environment, University of Colombo School of Computing (online) Available: <http://ugvle.ucsc.cmb.ac.lk/> and <http://vle.bit.lk/mod/book/view.php?id=3897&chapterid=12> [Accessed: 29th, May 2018]
- [3] Assignment Submission System (online), Available:[http://ir.unimas.my/7568/1/Computerized%20assignment%20%20project%20submission%20system%20for%20F.S.K.T.M.%20\(24%20pages\).pdf](http://ir.unimas.my/7568/1/Computerized%20assignment%20%20project%20submission%20system%20for%20F.S.K.T.M.%20(24%20pages).pdf) [Accessed: 30th , May 2018]
- [4] Virtual Learning Environment, Faculty of Management and Finance, University of Colombo (online) Available: <http://lms.mgmt.cmb.ac.lk/my/> [Accessed: 11th, Feb 2019]
- [5] Virtual Learning Environment. University Of Colombo, IHRA (online) Available: <http://lms.cmb.ac.lk/ihra/> [Accessed: 27th, Jan 2018]
- [6] Learning Organizer, University of Moratuwa(online) Available:[https://lms.mrt.ac.lk/login\\_index.php](https://lms.mrt.ac.lk/login_index.php) [Accessed: 08th , Jan 2019]
- [7] Software Processess (online) Available: . [http://moodle.autolab.uni-pannon.hu/Mecha\\_tananyag/szoftverfejlesztési\\_folyamatok\\_angol/ch03.html](http://moodle.autolab.uni-pannon.hu/Mecha_tananyag/szoftverfejlesztési_folyamatok_angol/ch03.html) [Accessed: 12th , Jan 2019]
- [8] Class Diagram (online) Available: [https://www.tutorialspoint.com/uml/uml\\_class\\_diagram.htm](https://www.tutorialspoint.com/uml/uml_class_diagram.htm) [Accessed: 6th Feb, 2019]
- [9] Apache Web Server(Online) Available: <https://www.wpbeginner.com/glossary/apache/> [Accessed: 6th Feb, 2019]
- [10] What is Relational database Management system(online) Available: <https://www.codecademy.com/articles/what-is-rdbms-sql> [Accessed: 6th Feb, 2019]
- [11] Navicat For MySQL(online) Available: , <https://www.navicat.com/en/products/navicat-for-mysql> [Accessed: 27th April, 2019]
- [12] What is Integrated Development Environment (IDE), Available: <https://searchsoftwarequality.techtarget.com/definition/integrated-development-environment> [Accessed: 7th May, 2019]
- [13] Black box Testing (online) Available: <http://softwaretestingfundamentals.com/black-box-testing/> [Accessed: 7th May, 2019]

[14] White Box Testing (online) Available: <http://softwaretestingfundamentals.com/white-box-testing/> [Accessed: 7th May, 2019]

[15] katalon recorder (online) Available: <https://chrome.google.com/webstore/detail/katalon-recorder/ljdobmomdgdlniojadhoplhkpialdid> [Accessed: 21st Mar, 2019]

## Appendix A: Code Listing

### A.1 Submission Model

The following figure A.1 is used to submit the one of categories in students' submissions to the database . It shows the example code for the client agreement form submission.

```
?PHP
INCLUDE './DBCONNECTION.PHP';
SESSION_START();
IF (ISSET($_POST['SUBMIT'])) {
    $STUREG = $_SESSION['REGISTRATIONNo'];
    //HTTPS://WWW.CODEMILES.COM/PHP-TUTORIALS/UPLOAD-PDF-FILE-IN-PHP-T1486.HTML
    DEFINE("FILESPLACE", "../FILES/CAF");
    $SQL = "INSERT INTO `SUBMISSION` (`REGISTRATIONNo`, `SUB_TYPE`, `SUB_ID`) VALUES
('STUREG', '4', '4')";
    $SQL2 = "UPDATE `STUDENT` SET `CAF`='1' WHERE
`REGISTRATIONNo`=`REGISTRATIONNo`=`$STUREG`";
    IF ($CONN->QUERY($SQL) == TRUE && $CONN->QUERY($SQL2)) {
        IF (IS_UPLOADED_FILE($_FILES['FILETOUPLOAD']['TMP_NAME'])) {
            IF ($_FILES['FILETOUPLOAD']['TYPE'] != "APPLICATION/PDF") {
                ECHO "<P>CLASS NOTES MUST BE UPLOADED IN PDF FORMAT.</P>";
            } ELSE {
                $RESULT = MOVE_UPLOADED_FILE($_FILES['FILETOUPLOAD']['TMP_NAME'], FILESPLACE . "/"
$STUREG.PDF");
                IF ($RESULT == 4)
                    ECHO "<SCRIPT TYPE='TEXT/JAVASCRIPT'>WINDOW.ONLOAD = FUNCTION()
{IF(CONFIRM('SUCCESSFULLY UPLOADED THE CAF!')==TRUE){WINDOW.LOCATION.HREF = '../
CAF_SUBMIT.PHP';}};</SCRIPT> ";
                ELSE
                    ECHO "<SCRIPT TYPE='TEXT/JAVASCRIPT'>WINDOW.ONLOAD = FUNCTION()
{IF(CONFIRM('ERROR OCCURED')==TRUE){WINDOW.LOCATION.HREF = '../CAF_SUBMIT.PHP';}};</SCRIPT>
"; // CHANGE THIS TO REDIRECT TO THE SAME PAGE
            } #ENDIF
        } #ENDIF
    } ELSE {
        ECHO "<SCRIPT TYPE='TEXT/JAVASCRIPT'>WINDOW.ONLOAD = FUNCTION() {IF(CONFIRM('ERROR
OCCURED')==TRUE){WINDOW.LOCATION.HREF = '../CAF_SUBMIT.PHP';}};</SCRIPT> "; // CHANGE THIS TO
REDIRECT TO THE SAME PAGE
    }
    $CONN->CLOSE();
}
?>
```

Figure A.1: CAF Submission Model

## A.2 Submission View

Below Figure A.2 shows the view of the submission of Client Agreement Form Submission as an example.

```
<DIV CLASS="ROW">

    <div class="card shadow mb-4 col-md-12">

        <DIV CLASS="CARD-HEADER PY-3">
            <H6 CLASS="M-0 FONT-WEIGHT-BOLD TEXT-PRIMARY">SUBMISSION
STATUS</H6>
        </DIV>
        <DIV CLASS="CARD-BODY">
            <DIV CLASS="TABLE-RESPONSIVE">
                <TABLE CLASS="TABLE TABLE-BORDERED" WIDTH="100%"
CELLSPACING="1">

                    <THEAD>
                        <TR>
                            <TH>REGISTRATION NUMBER</TH>
                            <TH>SUBMISSION DATE</TH>
                            <TH>DOWNLOAD FILE</TH>
                        </TR>
                    </THEAD>
                    <TBODY>
                        <?PHP
                        $RESULT = GET_DP_LIST();
                        WHILE ($ROW = $RESULT->FETCH_ASSOC()) {
                            EXTRACT($ROW);
                            ECHO "<TR>";
                            ECHO "<TD>" . $REGISTRATIONNO . "</TD>";
                            ECHO "<TD>" . $SUB_DATE . "</TD>";
                            ECHO "<TD><A HREF='./FILES/DP/' . $REGISTRATIONNO .
.PDF'>" . $REGISTRATIONNO . ".PDF</A></TD>";
                            ECHO "</TR>";
                        }
                        ?>
                    </TBODY>
                </TABLE>
            </DIV>
        </DIV>
    </DIV>
</DIV>
```

Figure A.2: View of the Submission (Client Agreement Form Submission)

### A.3 Session Handling

PHP is one of the most popular and widely used Server side scripting language. Session handling is one of the key thing which most of web applications and projects need. Below figure A.3 depicts the session handling made in the Information Management System for BIT final year projects.

```
<?PHP
    session_start();
    IF ($_SESSION['TYPE'] == '1' || $_SESSION['TYPE'] == '2') {

    } ELSE {
        HEADER("LOCATION: LOGOUT.PHP");
    }

    INCLUDE './INCLUDE/HEADER.PHP';
    INCLUDE './MODEL/VIEWCAFMODEL.PHP';
    $PAGENo = 20005;
?>
```

Figure A.3 Session Handling on page

### A.4 Update Student Details

The following figure A.4 depicts the update student details code.

```

<?PHP
INCLUDE './DBCONNECTION.PHP';
IF (ISSET($_POST['SUBMIT'])) {

//HTTPS://WWW.CODEMILES.COM/PHP-TUTORIALS/UPLOAD-PDF-FILE-IN-PHP-T1486.HTML
$ID = $_POST['ID'];
$NIC = $_POST['NIC'];
$REGISTRATIONNo = $_POST['REGISTRATIONNo'];
$INDEXNo = $_POST['INDEXNo'];
$FIRSTNAME = $_POST['FIRSTNAME'];
$LASTNAME = $_POST['LASTNAME'];
$EMAIL = $_POST['EMAIL'];
$CONTACTNo = $_POST['CONTACTNo'];

$SQL = "UPDATE `STUDENT` SET
`NIC`='$NIC', `REGISTRATIONNo`='$REGISTRATIONNo', `CONTACTNo`='$CONTACTNo', `INDEXNo`='$INDEXNo', `FIRSTNAME`='$FIRSTNAME', `LASTNAME`='$LASTNAME', `EMAIL`='$EMAIL' WHERE `AUTOID`
= '$ID'";
IF ($CONN->QUERY($SQL) == TRUE) {

    ECHO "<SCRIPT TYPE='TEXT/JAVASCRIPT'>WINDOW.ONLOAD = FUNCTION()
{IF(CONFIRM('SUCCESSFULLY UPDATED THE RECORD')==TRUE){WINDOW.LOCATION.HREF = '../STUDENTLIST.PHP';}};</SCRIPT> ";
}ELSE

    ECHO "<SCRIPT TYPE='TEXT/JAVASCRIPT'>WINDOW.ONLOAD = FUNCTION() {IF(CONFIRM('ERROR OCCURED')==TRUE){WINDOW.LOCATION.HREF = '../EDITPROFILE.PHP';}};</SCRIPT> ";
#ENDIF
$CONN->CLOSE();
}
?>

```

Figure A.4 Update Student Details