



General Sir John Kotelawala Defense University

Faculty of Computing

Department of Computer Science

Group Project Undertaken in partial fulfillment of the requirement for the
BSc Computer Science/ Computer Engineering/ Software Engineering Degree

Intake 36

DESIGN REPORT

Group Details		
Group Number	Student Number	Student Name
8	D/BCS/19/0013	S.V. Wijesekara
	D/BCS/19/0018	L.R.T.D.M. Bandara
	D/BCS/19/0015	D.M.S.S. Dissanayake
	D/BCE/19/0004	O.R. Zain
	C/BSE/18/5706	K.A. Jayashanka
Project Details		
Project Title	Final Year Project Coordinating System	
Supervisor	Miss W.K.M.S. Ilmini	
Co-Supervisor		

Contents

1.0 Introduction.....	3
1.1 Purpose.....	3
1.2 Scope	3
1.3 Definitions, Acronyms, abbreviations	3
2.0 System architecture	3
2.1 Application Layer	4
2.2 Data Layer	5
2.3 Presentation Layer.....	5
3.0 Software Architecture	5
3.1 Overall Software Architecture.....	6
3.2 Module Architecture.....	7
3.3 Module Architecture in Detail.....	8
4.0 Design Overview.....	10
4.1 Description of Problem.....	10
4.2 Technologies Used.....	11
4.3 System Operation.....	12
4.4 Process.....	13
5.0 Sample GUI Designs for The Web-app.....	14
6.0 Data Design.....	16
6.1 Conceptual Database Design.....	17
6.2 Mapping of Logical Database to Relations	18
7.0 References.....	21

1.0 Introduction

1.1 Purpose

The purpose of this document is to describe the implementation of the final year project coordinating system created for the FOC of Kotelawala Defence University. The FYPCS is designed to efficiently coordinate the final year projects through a computerized system.

1.2 Scope

This document describes the implementation details of the FYPCS. The software will consist of a list of step by step approaches till the final thesis submission and the final presentation submission of the final year projects of FOC in KDU. First, it will start by the normal form submissions and then project proposals and it will also provide the final year students of FOC the deadlines, submission links and the results of their past submissions, so they can get a rough idea of their projects' marks.

1.3 Definitions, Acronyms, abbreviations

FOC – FACULTY OF COMPUTING

KDU – KOTELAWALA DEFENCE UNIVERSITY

FYPC- FINAL YEAR PROJECT COORDINATING SYSTEM

2.0 System architecture

Architectural design defines the overall structure of the system. The architectural design is given according to the three-tier-architecture where overall design is split into three layers of presentation tier, application tier and data tier.

The overall system architectural design for the proposed system is as follows;

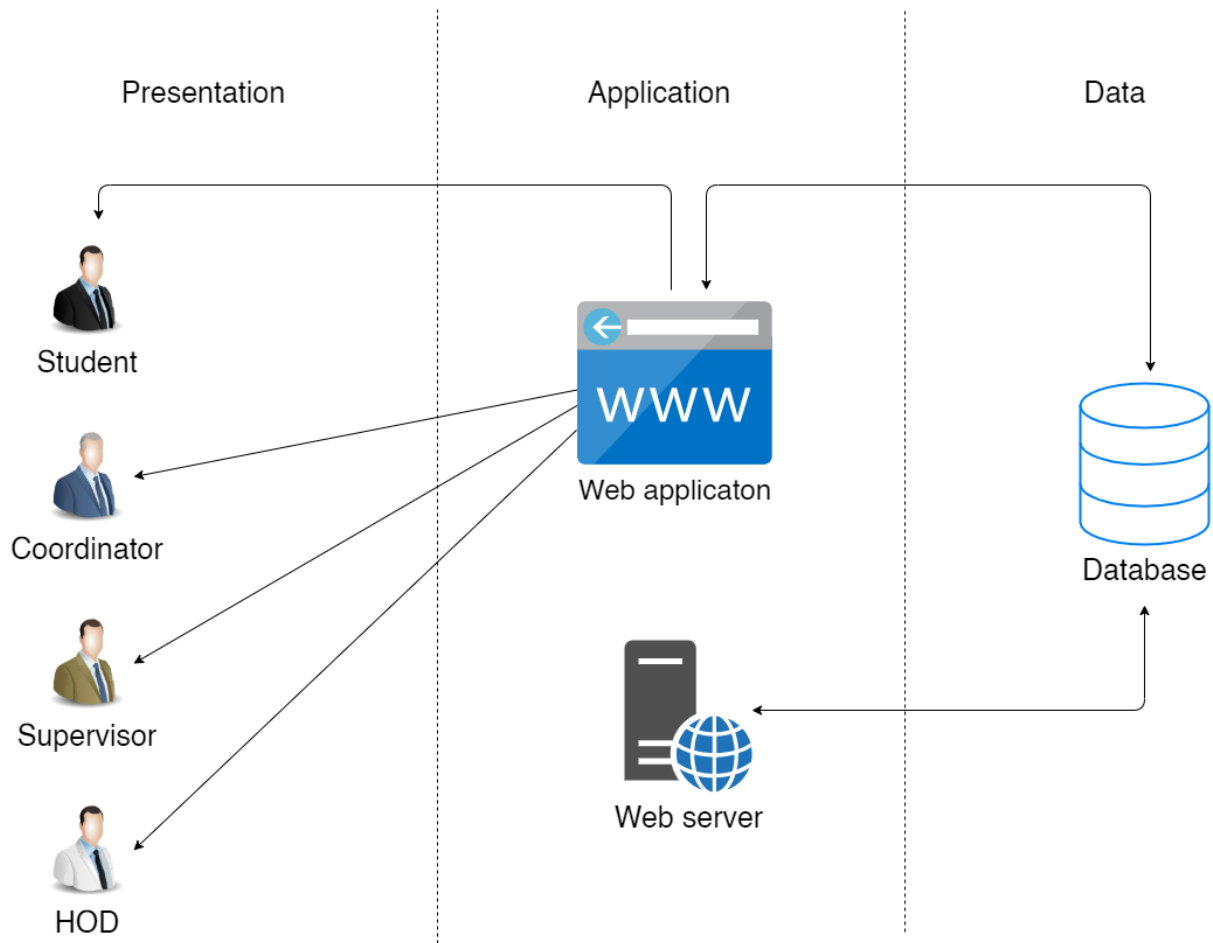


Figure 2.0

2.1 Application Layer

Application layer shows us that this is a web platform. Application layer connects with presentation layer and data layer. web application is connected with application layer and the admin has ability to control this layer. Application layer is the center of the entire system. The information about the students and the tasks of the both parties will be displayed in the application layer. Users have ability to update the information in this section. Application layer interacts with both other layers. The changes have done in this section affects for both other presentation and data layers.

2.2 Data Layer

Data layer manages the database of the web application. This layer helps to manage the data storage operations of the overall system. This is the section where the all database management applications are running. Firebase is the platform to store database information in our project. We are managing a single database and it is connected with several tables. Admin(coordinator) has the ability to include and remove the details from the tables.

2.3 Presentation Layer

Presentation layer helps to get the interaction of the users. This monitors the interfaces of the application. This is the only layer that user can see. Coordinator signing in, Student signing in, Supervisor assignment, Files uploading and downloading, and process viewing are happening in this layer. User has the ability to enter details in the presentation layer and the details entered in the presentation layer have delivered to the application layer.

3.0 Software Architecture

Software architecture was based on modularized approach where the software is divided into parts. Each module has more than one tasks of the overall system to achieve the ultimate objectives expected. The software system is developed by the use of the Visual Studio Code and the databases are maintained by the use of phpMyAdmin software.

3.1 Overall Software Architecture

Following figure represents the overall software architecture of the developing system.

Figure 3.0 : Overall Software Architecture

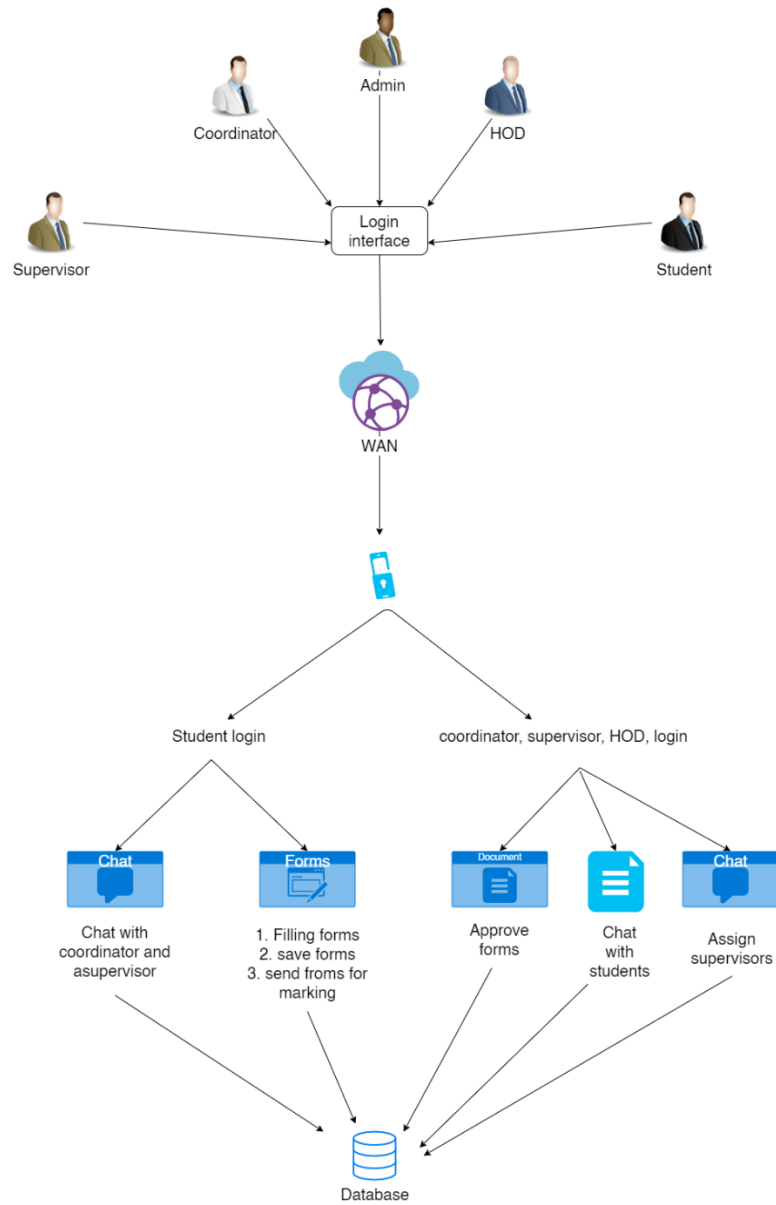


Figure 3.0

3.2 Module Architecture

The overall software system has been divided into several modules. This section will describe about the organization of the modules that it consists.

An Overview of Module Architecture

In the below shown is an enumerated list of requirements for the new system development.

Module 1 : “ Login Module”

1.1 : Admin / Supervisor's / Student's login

Module 2 : “System Administration”

2.1 : Manage supervisor / student's records.

2.1.1 Register a new Student / Supervisor

2.1.2 Edit user details

2.1.3 Remove users

Module 3 : “Student module”

3.1 Fill the relevant forms

3.2 save the filled forms

3.3 send for approval

3.4 acknowledge the response

Module 4 : “Coordinator module”

4.1 Respond to filled forms by students

4.2 assign supervisors

Module 4 : “View Tasks & Students Records”

4.1 View Student profile

4.2 View Supervisor profile

3.3 Module Architecture in Detail

There are several modules for various functions in the system. In the below it is described in detail about those modules.

Module 1 : Login Module”

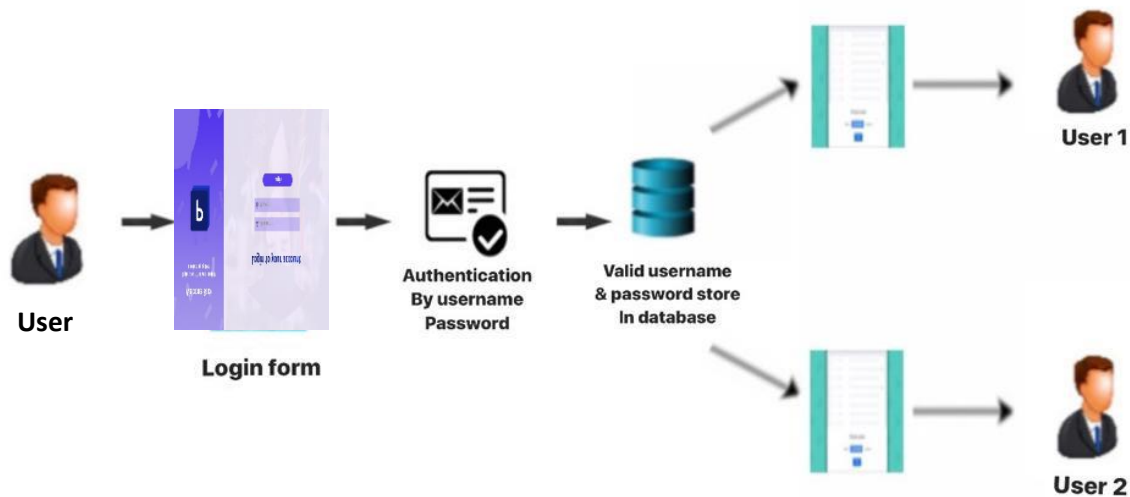


Figure 3.0: Supervisor’s /Students’ Login Module

Users and students have to login to the system. The authentication will be done by using sessions. Then we verify username and password from the details stored in the database. Then users can access the relevant parts of the software.

Module 2 : “System Administration”

Mod 2.1 Manage Supervisor / Students' records

Mod 2.1.1 Add a new student / lecturer

The admin (Coordinator) can add students and lecturers to the system. The above figure (figure 3.0) shows how the administration will be happen in our system.



Figure 4.0 Edit user details

Only the admin has access to edit the user details. All the edited details are stored in the database. Above figure shows how the editing process happens in the whole program.

Removing can only be happened by Admin



Figure 5.0 Remove users

4.0 Design Overview

4.1 Description of Problem

The existing manual project coordination process does not support the essential needs of the university undergraduates and the lecturers. It is quite difficult to update a file based approach as the data processing and event planning takes more time than anticipated and normally it showed that many conflicts had been faced by the students and the staff annually regarding the time limitation and direct contact between the lecturers and the students.

The FYPCS will replace the current manual process with a computerized approach and will make the project coordinating more efficient.

4.2 Technologies Used

- phpMyAdmin
- Adobe Photoshop CC 2018
- PHP
- JavaScript
- Bootstrap
- HTML
- Visual studio code

The FYPCS will be coded in PHP and HTML and will use Visual Studio Code, the interfaces are designed with the help of HTML and Adobe Photoshop and it is connected to the database phpMyAdmin.

4.3 System Operation

Figure 1 shows the sequence the FYPCS events will occur.

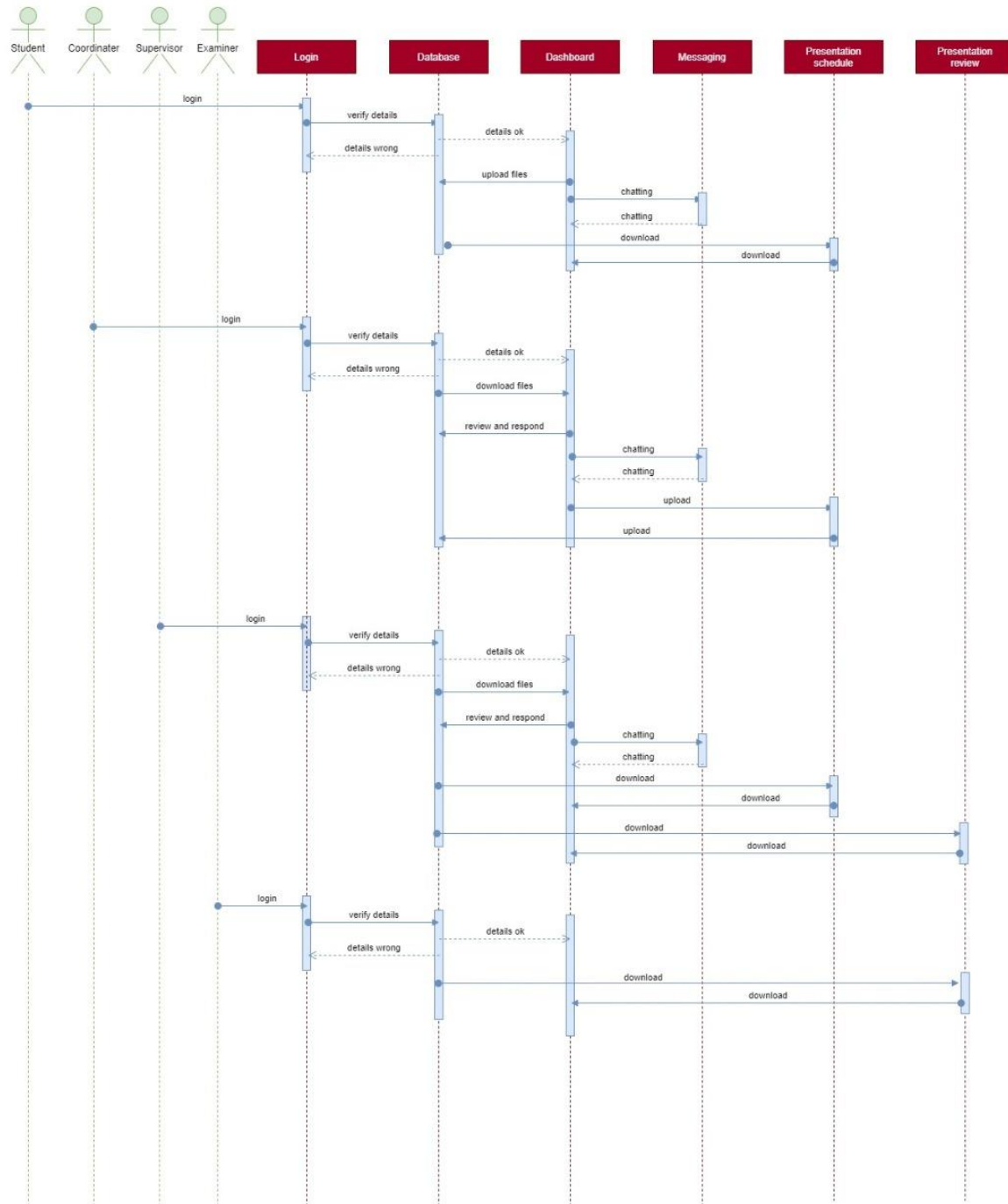


Figure 1

4.4 Process

Given below is the step by step process description described for each process that will be done from the FYPCS.

Forms for students

1. Fill the form
2. Save for further editing
3. Able to upload excel files to necessary forms
4. Send for marking
5. Get the “form accepted” message from coordinator

Assigning supervisors

1. Read the submitted norm forms and assign respective supervisors according to the project area
2. Send details about chosen supervisors to respective students

Chat

1. Chat between student - coordinator and student - supervisor

View submitted forms

1. View forms submitted by students to respective parties marked by viewed/not viewed/responded/not responded

Forms for supervisors/coordinators

1. Approve/disapprove and alert students.

Home page for coordinator

1. View the number of different forms submitted by students. And percentage of submitted forms
2. Differentiate between new forms, marked and respond sent forms, viewed yet not marked forms.
3. View messages by student name.

5.0 Sample GUI Designs for The Web-app

Figure 3 and figure 4 shows the supervisor assigning page.

Figure 5 shows the view submitted forms page

Figure 6 shows the login page

localhost/backen/coordinat

localhost/backen/coordinat

localhost/backen/coordinat

localhost/backen/coordinat

localhost/backen/coordinat%20interface/assigning%20supervisors/index%20ver%202.0.php

Student name	Supervisor 1	Supervisor 2	Supervisor 3
research area -> 1			
1 1	1	1	1
ddddddddd ddddddddd			
ll ll			
tt tt			
oo oo			
oopm ooopm			
oopp oopp			
ww ww			

Supervisor's name	Total	s1	s2	s3
Commodore JU Gunaseela	0	0	0	0
Dr. HRWP Gunathilaka	0	0	0	0
Dr. LP Kalansooriya	0	0	0	0
Maj. RMM Pradeep	0	0	0	0
Miss. CP Waduge	0	0	0	0
Miss. WMSK Ilmini	0	0	0	0
Mr. ADAI Gunasekara	0	0	0	0
Mr. Buddhitha Hettige	0	0	0	0
Mr. DMR Kulasekara	0	0	0	0
Mr. GIF De Silva	0	0	0	0
Mr. L Wicramasinghe	0	0	0	0
Mr. PPNV Kumara	0	0	0	0
Mr. RPS Kathiriarachchi	0	0	0	0
Mr. WAAM Wanniarachchi	0	0	0	0
Mr. WMSRB Wijerathne	0	0	0	0
Mrs. BHD Perera	0	0	0	0
Mrs. DU Vidanagama	0	0	0	0
Mrs. ERC Sandamali	0	0	0	0
Mrs. IA Wijethunga	0	0	0	0
Mrs. N Wedasinghe	0	0	0	0
Mrs. SCM De Silva Sirisooriya	0	0	0	0
Mrs. WJ Samaraweera	0	0	0	0
Mrs. WPJ Premarathna	0	0	0	0

Figure 3

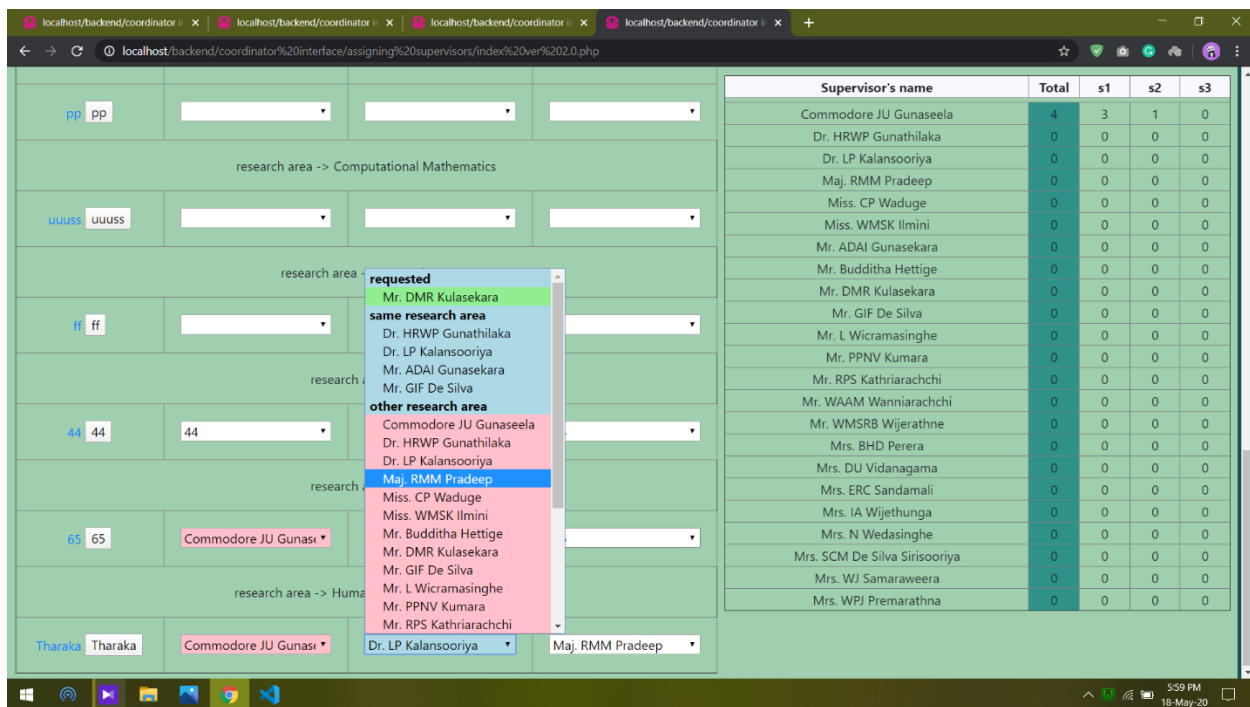


Figure 4

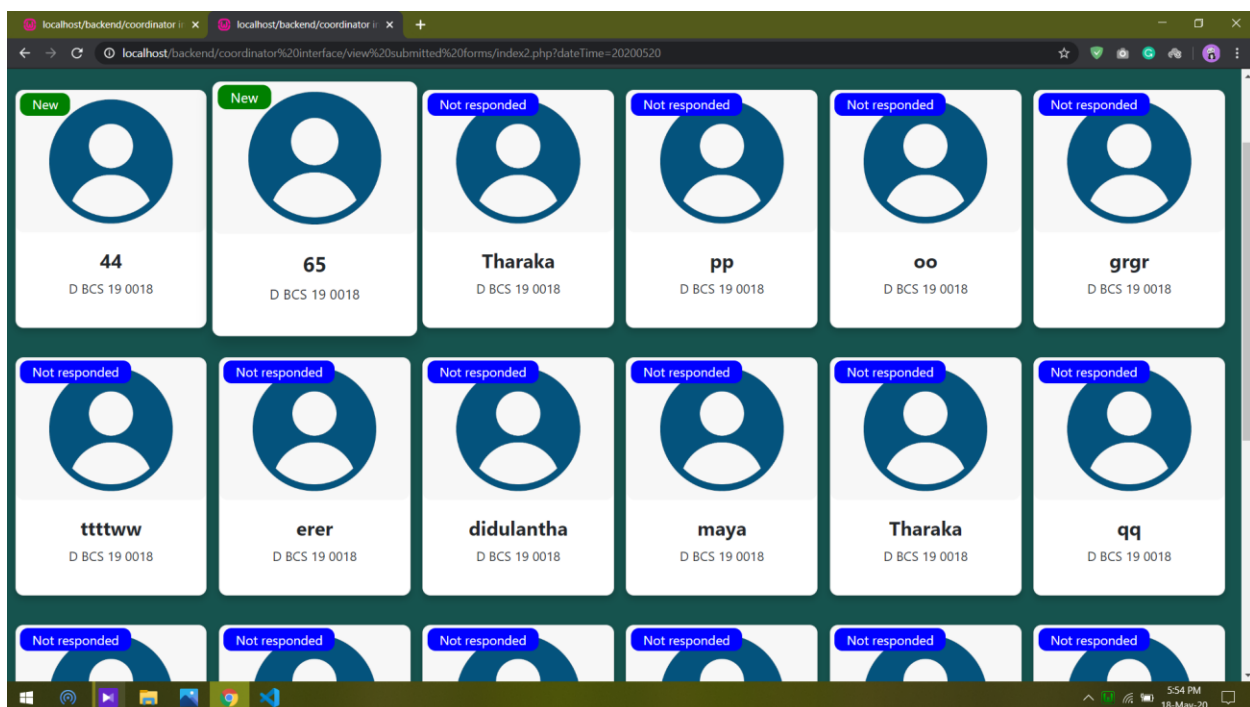


Figure 5

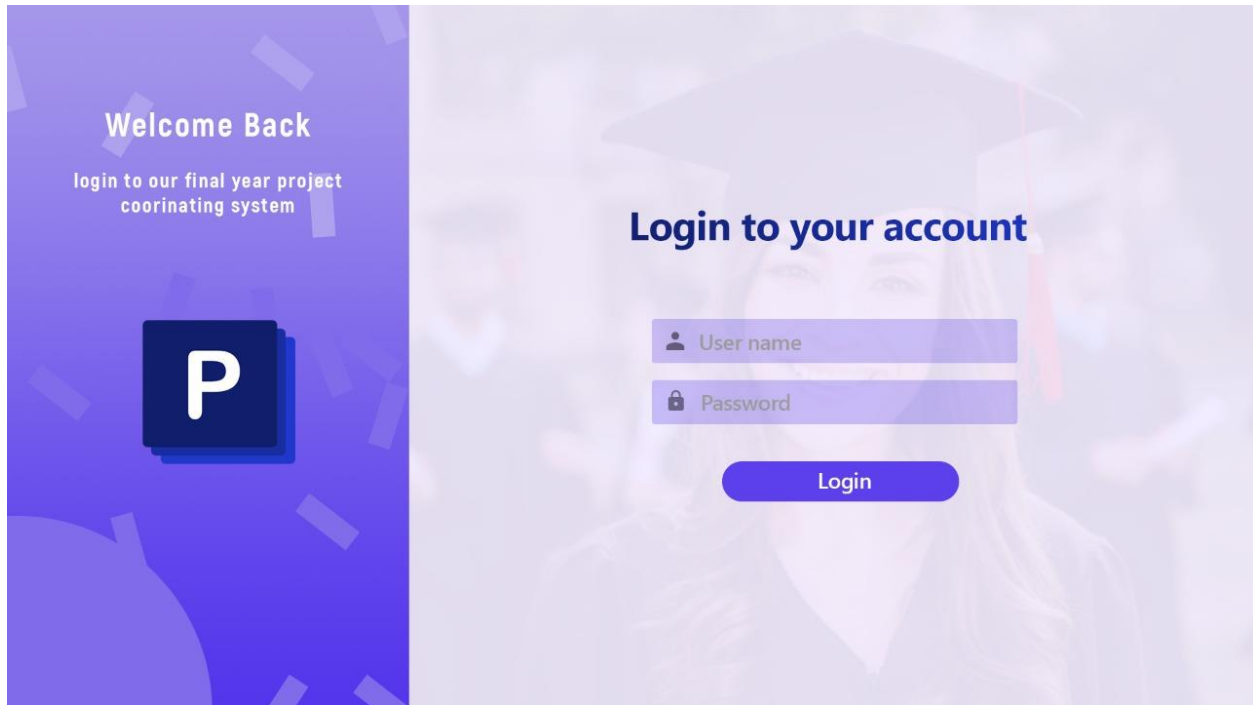


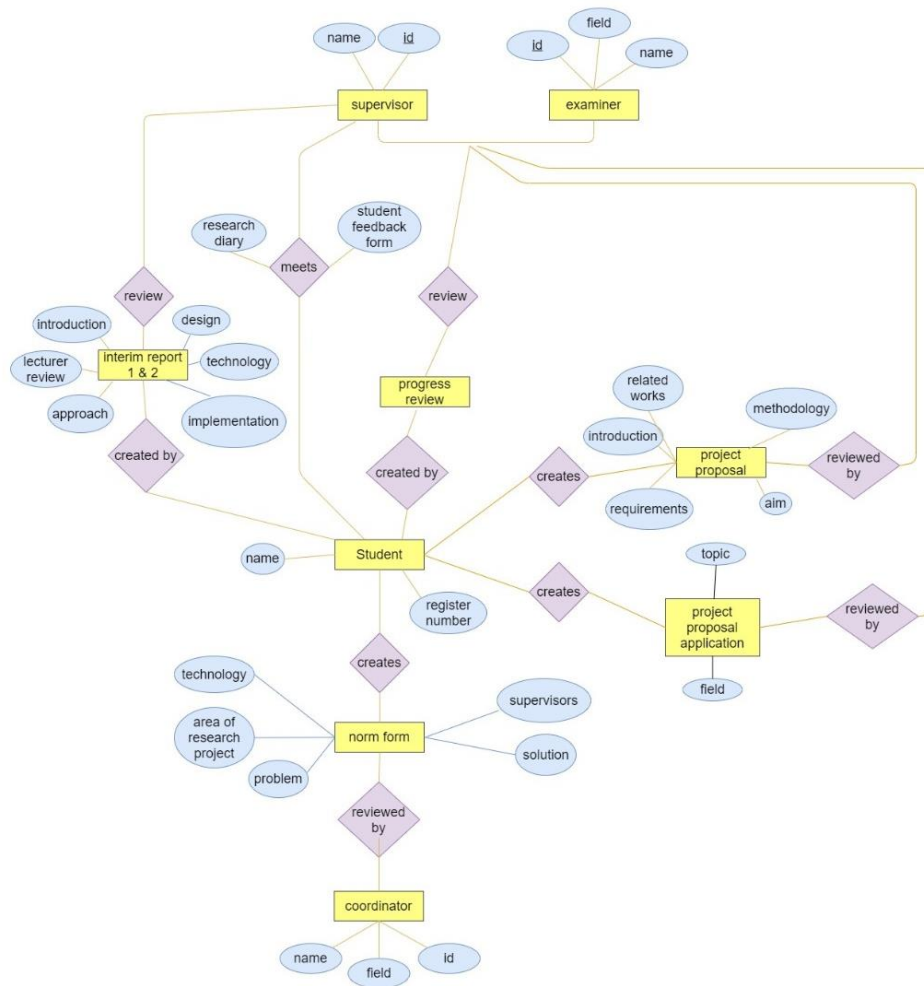
Figure 6

6.0 Data Design

In our system we will have a centralized data base in the server. phpMyAdmin will be used for this. This section of the document represents the conceptual data design of the system, the process of converting EER diagram into table and the database relationship diagram. Furthermore, the table of the database are also stated with their attributes and data types.

6.1 Conceptual Database Design

Following figure shows the EER model.



6.2 Mapping of Logical Database to Relations

Regular Entity Types

appendix1(FirstName,lastName,Email,ContactNo,supervisor1,supervisor2,supervisor3,project_title,problem,solution,technology)

Appendix2(Index_No,Project_Title,Examinor1,Examinor2,Supervisor)

Appendix 3 (Resaerch_Area,Supervisor1_Name , Supervisor1_Institution, Supervisor1_Area, Supervisor1_Contact_Num, Supervisor1_Email, Supervisor2_Name , Supervisor2_Institution, Supervisor2_Area, Supervisor2_Contact_Num, Supervisor2_Email, Supervisor3_Name , Supervisor3_Institution, Supervisor3_Area, Supervisor3_Contact_Num, Supervisor3_Email,Student_Name, Student_Index No,Stream Student_Contact_Num Supervisor3_Contact_Num ,E-mail)

Appendix4(Project_Title,Student_Name,Supervisors,Outline_Activities,Feedback,Date)

Appendix5(Index_NO,Title,Comments,Final_Marks,Examiner1_Name,Examiner2_Name,Supervisor_Name,Date)

Supervisors(indexNo, Name, ResearchArea)

Student(indexNo, email, password, dateTime)

Admin(indexNo, email, password, dateTime)

Staff(indexNo, email, password, dateTime)

Chat(indexNo1, indexNo2, name, message, dateTime, date, time)

6.3 Data Type Design for the Database

This table shows the tables of the database with their attributes and respective data types.

TABLE-Admin		
Attribute	Data Type	Length
username	varchar	20
password	varchar	20
User_type	varchar	15
dateTime	dateTime	10

TABLE-Supervisor		
Attribute	Data Type	Length
username	varchar	20
password	varchar	20
User_type	varchar	15
dateTime	dateTime	10

TABLE-SUPERVISOR		
Attribute	Data Type	Length
Supervisor_id	varchar	20
Supervisor_name	varchar	20
username	varchar	25
password	varchar	25
Research_Area	varchar	25
Contact_Num	varchar	25
Email	varchar	25

TABLE-EXAMINER		
Attribute	Data Type	Length
Examiner_id	varchar	20
Examiner_name	varchar	20
username	varchar	25
password	varchar	25
Contact_Num	varchar	25
Email	varchar	25

TABLE-STUDENTS		
Attribute	Data Type	Length
Student_id	varchar	20
Student_name	varchar	20
username	varchar	25
password	varchar	25
email	varchar	25
phone	varchar	10

TABLE-NORM_FORM(APPENDIX_1)		
Attribute	Data Type	Length
Student_id	varchar	25
First_Name	varchar	100
Last_Name	varchar	100
Email	varchar	25
Contact_num	varchar	25
Supervisor1	varchar	25
Supervisor2	varchar	25
Supervisor3	varchar	25
Project_Title	varchar	250
Problem	text	100
Solution	text	100
Technology	text	100

TABLE-PROJECT_EVALUTION(APPENDIX_3)		
Attribute	Data Type	Length
Student_id	varchar	25
Supervisor	varchar	10
Examinor1	varchar	25
Examinor2	varchar	25
date	text	25

7.0 References

- [1] "www.sampletemplates.com," [Online]. Available: <https://www.sampletemplates.com/business-templates/design-document.html>.
- [2] "http://ant.comm.ccu.edu.tw," [Online]. Available: http://ant.comm.ccu.edu.tw/course/97_Programming/7_SampleCode/Design%20Document%20Template%20-%20%20Chapters.pdf.
- [3] "images.sampletemplates.com," [Online]. Available: <https://images.sampletemplates.com/wp-content/uploads/2017/01/27172714/Design-Control-Document.zip>.
- [4] "www.projectmanagementdocs.com," [Online]. Available: <https://www.projectmanagementdocs.com/template/project-documents/system-design-document/#axzz6MhyeAv22>.
- [5] "writing.engr.psu.edu," [Online]. Available: <http://writing.engr.psu.edu/workbooks/designreport.html>.
- [6] "www.freecodecamp.org," [Online]. Available: <https://www.freecodecamp.org/news/how-to-write-a-good-software-design-document-66fcf019569c/>.
- [7] "www.toptal.com," [Online]. Available: <https://www.toptal.com/freelance/why-design-documents-matter>.
- [8] "www.craftofscientificwriting.com," [Online]. Available: <https://www.craftofscientificwriting.com/design-reports.html>.