prepared by ainin sofiya

BRIEF REPORT ON WORKSHOP 1 : GISC REGISTRATION SYSTEM

CHAPTER 1: INTRODUCTION

1.1 Project Background

Global I-Lead STEM Camp (GISC) is a program by Universiti Putra malaysia(UPM) to provide exposure to school students who are participants of the competition on the importance Science, Technology, Engineering and Mathematics Education or known as STEM Education.

As an IT developer, an organized system need to be designed and developed to help the registration of this program in computerised way as for reducing the errors that could occurs during manual registration system. This system named as GISC Registration System.

The ‘GISC Registration System’ is a system that allows the user which is the school’s student or thier parents to make fee and registration on the available courses provided by the University. This system need to be developed as there will be many student will participate in this GISC program.

1.2 Problem Statement

There are a lot of problem occur during the student registration which are:

a. The registeration system of university is unable to manage the high capacity of telephone calls during registeration day. Hence, the common problem may occurs such as line busy or long distance charges.

b. There much paperwork to do and piles of paper are needed for printing the registeration forms and required more time to distribute to the school students which lead to late get feedback.

c. Major errors occur while filling up the details in the registeration forms.

1.3 Objectives

The objectives of this project are:

1. To develop and design new 100% computerized system for the GISC program in terms of their registeration services management.
2. To to ensure that the data or details in the system are organized and manageable consistently.
3. To ensure user can easily use the system especially to make 100% accurate data in taking or receiving the details from both participants and administration.

1.4 Scope

i. Target User

Users/Participants: Insert their details and requirement, final view and print registered details form.

Administration: View and keep the participants details, add some details required, update any new details, search participants by courses, or by faculty.

ii. Module

The module of the target users are divides into 2 categories which are administrator and participant which are the student.

Input: User able to insert, add, update or edit, search and sort the data and details.

Output: User able to view the list of courses available, preview their registered information or print the form.

Extended data:

USER(schoolName,schoolAddress,schoolCategory,teacherName,teacherNoHp,teacherEmail,studentName1,studentICNo1,studentNoHp1,gender1,sizeTshirt1,parentsorpenjagaName1,parentsorpenjagaNoHp1,studentName2,studentICNo2,studentNoHp2,gender2,sizeTshirt2,parentsorpenjagaName2 ,parentsorpenjagaNoHp2,courseName,refNo,pymentType1,pymentDate1,pymentTime1,pymentRefno1,totalpayment1,buktipyment1,pymentType2,pymentDate2,pymentTime2,pymentRefno2,totalpayment2,buktipyment2,joinCategory)

ADMIN(username, password)

1.5 System Requirement

- Operating system of Windows 7 or latest
- Any mobile phone either Android or Apple
- Internet access to open website
- Server with Microsoft SQL Server Database File(SqlClient) version equal or more than 5.6.12 MySQL

CHAPTER 2: PROBLEM ANALYSIS

2.1 Problem Description

There are many problem occurred during the student registration. One of the most difficulties the registration staff faced is unable to manage high capacity of telephone calls during the registration day. Therefore, it can cause the problem in communication such as line busy or long distance charges. This could give the higher intensity of unsatisfying among customers or participants especially in giving information to them.

Second problem is there are piles of paper required to make paperwork for each participant details. Plus, staffs need to print bunch of the registeration forms and required more time to distribute to the schools which lead to late get feedback. Since then, they are using google form, but there are still no actual database that able to store the customer or participant's details. All the details need to be transferred into paperwork as for keep it as manual database management. Hence, it can cause wastage and unorganized and loss data if anything happen.

Last is major errors that occur while filling up the details in the registeration forms are the problem that mostly happened to all participants or customers. Some of the time, they might leave the fill blank in certain required section which is the most important for registration information. This could give difficulties in the registeration part, where if the information given by the customer is incomplete they might be terminated just like that.

2.2 Problem Solution

As for solution to the 3 problems, there are something that can be improved in order to solve the problems.

As for the first problem, the solution is to develop and design new 100% computerized system for the GISC program in terms of their registrations services management. With fully computerised system, the registration work can be faster and better. All they need is hired the clerk that knew about IT and everything is right on their fingertips. Since they will use the computerised system, the database also in computerised. So, if customers insert their details, all the data will be kept automatically to a database with high capacity.

Solution 2 is ensuring the data or details being kept in the system are organize, manageable consistently in order to solve the paper wastage, time consuming and loss data . If they have computerised database management, they don't need to piles of paper to make registration forms and distribute them to schools and students. With the system provided, customers must fill the forms just in time hence there will be no wasting time to gain feedback or response from customers. All the details will be saved into the database right after they done fill in the form. Hence, the registration have less time consuming and organised details and less wastage of papers.

Last solution is make user can easily use the system especially to make 100% accurate data in taking or receiving the details from both participants and administration. In order to provide an accurate data to organizer of the program, the customer must fill in all the blank in registration form. If there are one or more data not filled, the system will pop up message asking for fill in all the participant's details. Therefore, both user can easily interact if there is some changes due to errors exist while filling up the forms for joining the program which mean have less errors and more accurate data can being kept.

2.3 Problem Decomposition

Problem 1: Both side customer and registration staff faced problem in communication and unable to manage high of telephone calls during manual registration process.

Solution 1: Design and develop 100% computerised system with help section.

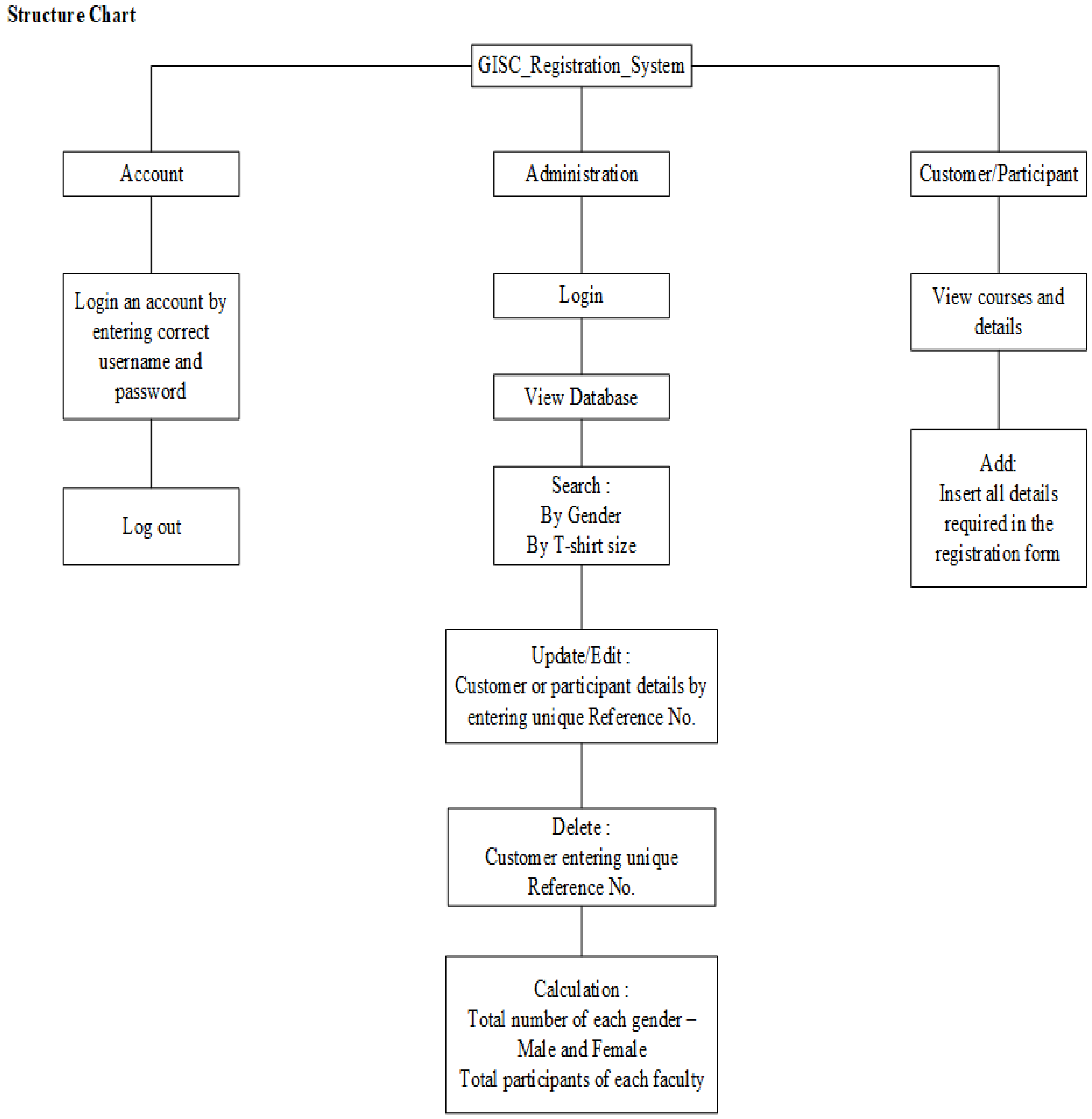
Problem 2: The registration staffs need piles of paper to make registration form and keep their registrar details in manual book keeping which might could lead to loss data if anything happen.

Solution 2: With the computerised system, the database management also can be digitalised, so all the participant's details can be kept in organised way.

Problem 3: Major error in registration form while participant or customer inserting their details which can lead to give wrong information.

Solution 3: System will provide error message if user insert wrong format in any section in e-registration form.

2.4 Structure Chart


Structure Chart
GISC_Registration_System
Account
Administration
Customer/Participant
Login an account by entering correct username and password
Log out
Login
View Database
Search :
By Gender
By T-shirt size
Update/Edit :
Customer or participant details by entering unique Reference No.
Delete :
Customer entering unique Reference No.
Calculation :
Total number of each gender – Male and Female
Total participants of each faculty
View courses and details
Add:
Insert all details required in the registration form

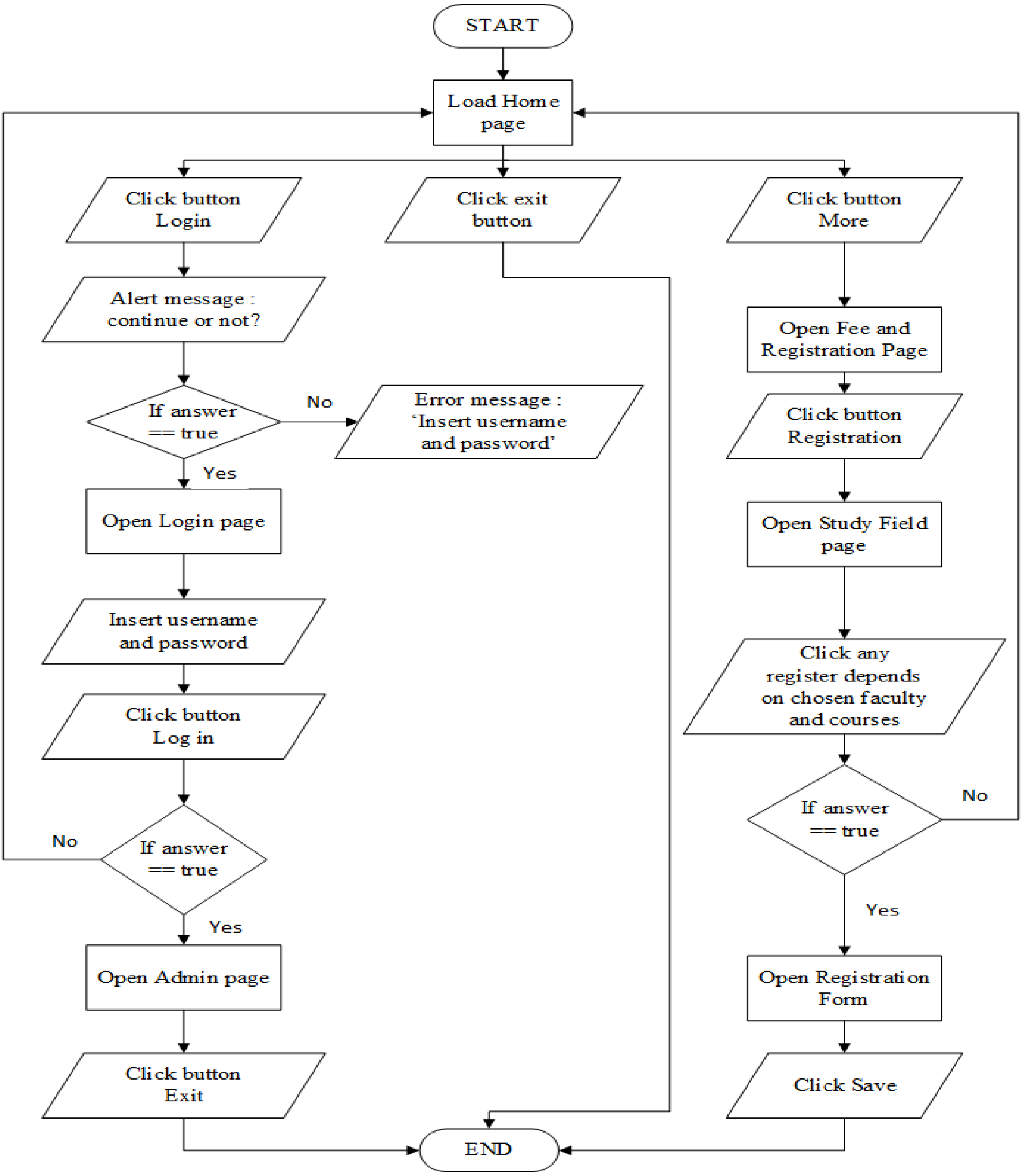
CHAPTER 3: DESIGN

3.1 Introduction

In this phase, the flow and structure of the system will be discuss. The process of the system flow chart, ERD diagram and Interface design will be state.

3.2 Flow Chart

Overall

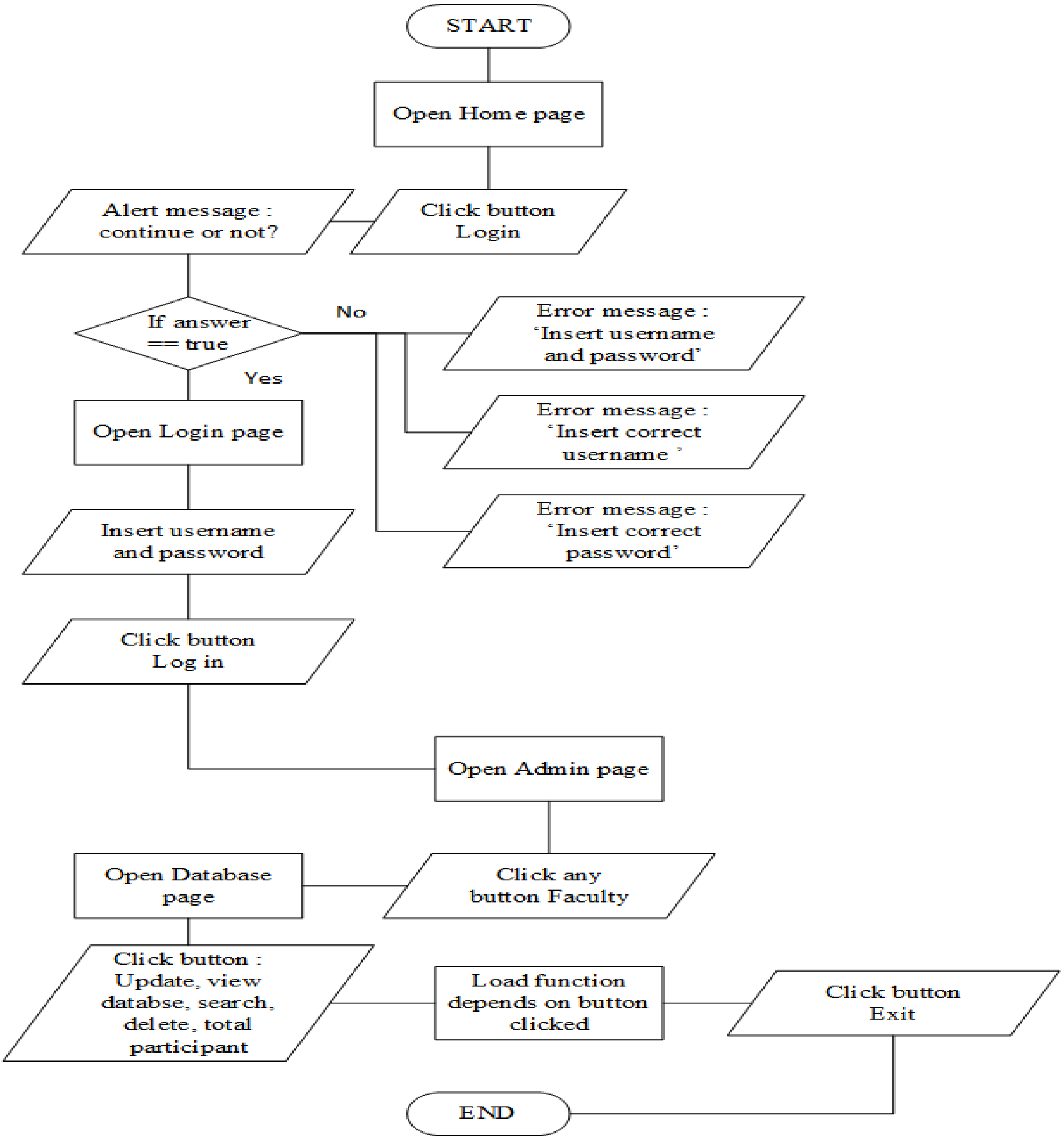


Flowchart 1: Overall

Explanation Overall flowchart:-

Firstly the Flowchart 1 Overall shows the system will run and load Homepage. There are 3 button provided which are button Exit to exit the application, button MORE INFO to load Fee and Registration page as for getting more information about the registration. While button LOGIN is for administration only. When user click MORE INFO button and load the Fee and Registration page, user then click the button Registration and the Bidang Kajian page will show. User may click any button REGISTER that related to each box which contain courses and faculties. If user click any of the button REGISTER, the registration form will display and user may enter the next data or exit application.

Login

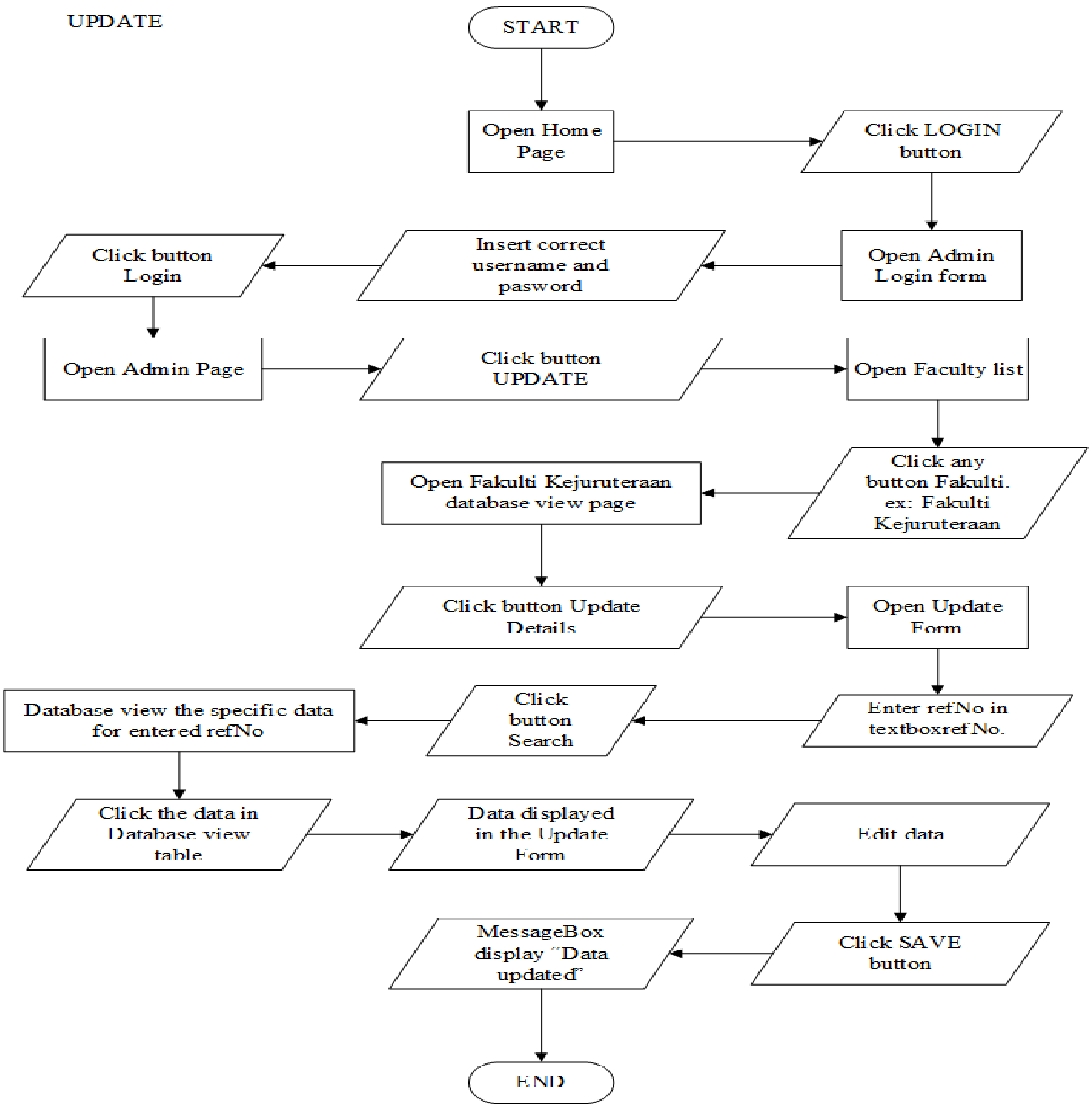

START
Open Home page
Click button Login
Alert message : continue or not?
If answer == true
No
Yes
Error message : 'Insert username and password'
Error message : 'Insert correct username '
Error message : 'Insert correct password'
Open Login page
Insert username and password
Click button Log in
Open Admin page
Click any button Faculty
Open Database page
Click button : Update, view databse, search, delete, total participant
Load function depends on button clicked
Click button Exit
END

Flowchart 2: Login

Explanation Login flowchart:-

Flowchart 2 above shows how the system allow user to login successfully if they insert correct username and password. If user entered the invalid details, the system will pop up message box to alert the user that their username and password are incorrect. As for accessing the database, system will make sure that only authorised user can go to proceed any other function after login successfully.

Update

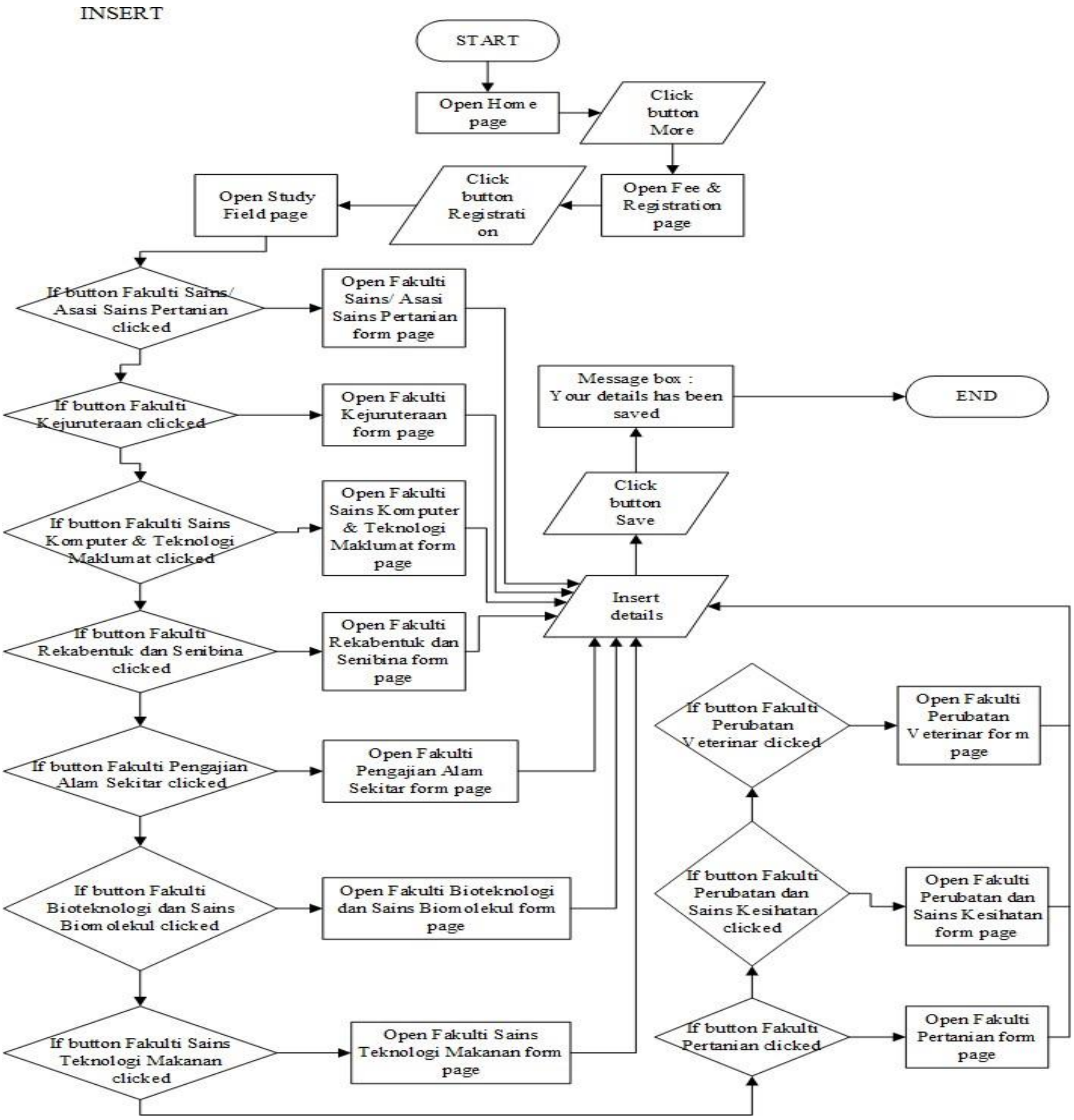

UPDATE
START
Open Home Page
Click LOGIN button
Open Admin Login form
Insert correct username and pasword
Click button Login
Open Admin Page
Click button UPDATE
Open Faculty list
Click any button Fakulti. ex: Fakulti Kejuruteraan
Open Fakulti Kejuruteraan database view page
Click button Update Details
Open Update Form
Enter refNo in textboxrefNo.
Click button Search
Database view the specific data for entered refNo
Click the data in Database view table
Data displayed in the Update Form
Edit data
Click SAVE button
MessageBox display "Data updated"
END

Flowchart 3: Update

Explanation Update flowchart:-

Flowchart 3 above shows how the user can update the participant's details. User just have to follow the flow and click the right button, insert any update requested by customer. The user may enter the unique reference number that is auto-generated while saving the details before. By entering the reference number, user may update the details that is related to the reference number only without messing the other's details.

Insert

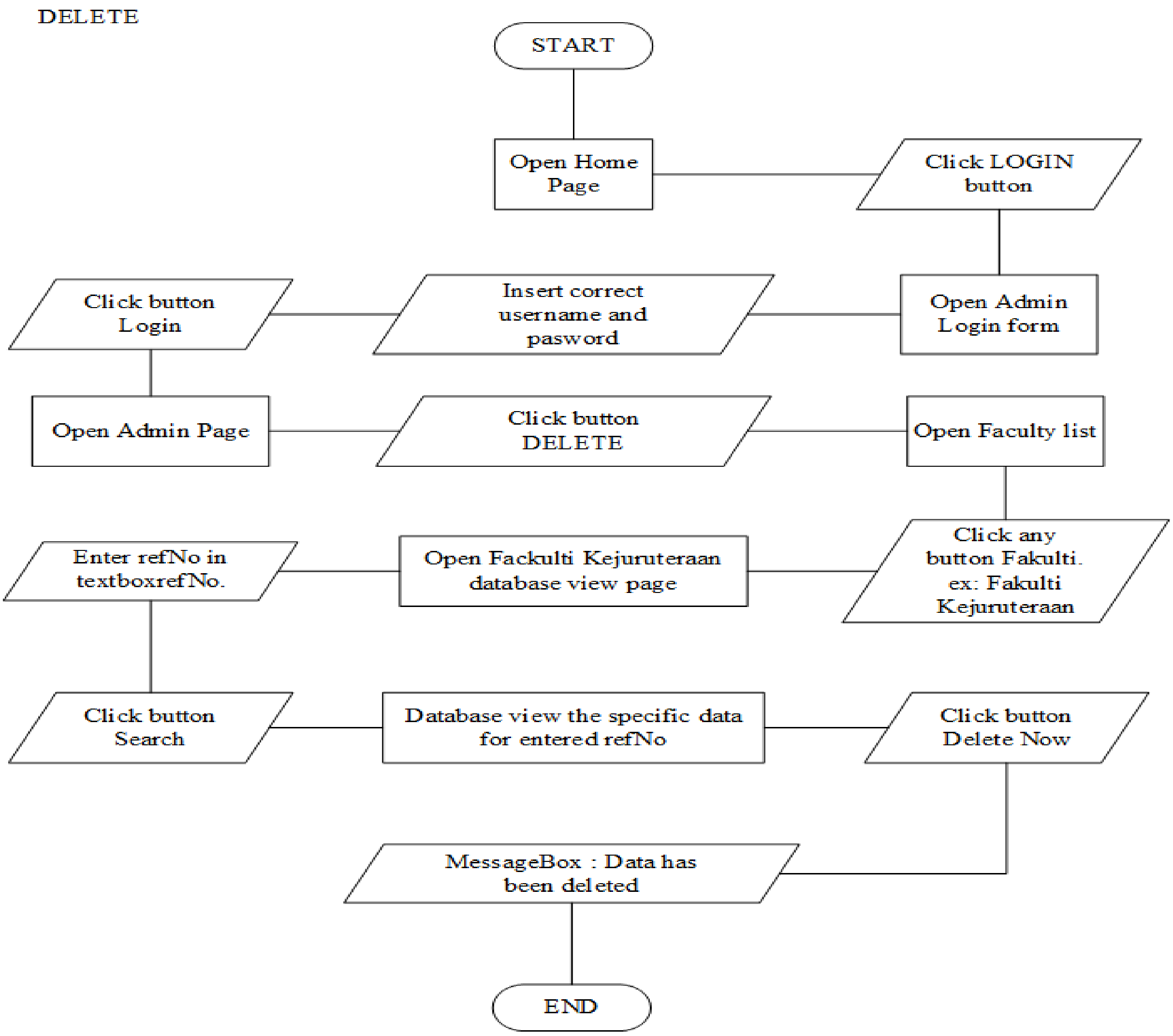

INSERT
START
Open Home page
Click button More
Open Fee & Registration page
Click button Registration
Open Study Field page
If button Fakulti Sains/ Asasi Sains Pertanian clicked
Open Fakulti Sains/ Asasi Sains Pertanian form page
If button Fakulti Kejuruteraan clicked
Open Fakulti Kejuruteraan form page
If button Fakulti Sains Komputer & Teknologi Maklumat clicked
Open Fakulti Sains Komputer & Teknologi Maklumat form page
If button Fakulti Rekabentuk dan Senibina clicked
Open Fakulti Rekabentuk dan Senibina form page
If button Fakulti Pengajian Alam Sekitar clicked
Open Fakulti Pengajian Alam Sekitar form page
If button Fakulti Bioteknologi dan Sains Biomolekul clicked
Open Fakulti Bioteknologi dan Sains Biomolekul form page
If button Fakulti Sains Teknologi Makanan clicked
Open Fakulti Sains Teknologi Makanan form page
If button Fakulti Pertanian clicked
Open Fakulti Pertanian form page
If button Fakulti Perubatan dan Sains Kesihatan clicked
Open Fakulti Perubatan dan Sains Kesihatan form page
If button Fakulti Perubatan Veterinar clicked
Open Fakulti Perubatan Veterinar form page
Insert details
Click button Save
Message box : Your details has been saved
END

Flowchart 4: Insert

Explanation Insert flowchart:-

Flowchart 4 shows that how user can insert their details within specific courses and faculty. System will run the Homepage and user need to click button MORE INFO to read more information about the fee program and registration. After that user may click button Registration and the Bidang Kajian page will display. User may choose any course suite their selves and for example user choose Fakulti Kejuruteraan. To proceed the registration, user need to click button REGISTER and system will display the registration form for Fakulti Kejuruteraan. The system will show the auto generate number in the textbox where the customer must remember and alert because each customer may have unique reference number. Then user may proceed the registration by inserting all the field and click button SAVE at the bottom of the page. System will pop up message either the details has been saved successfully or not. If not they must insert the correct details in the correct fields.

Delete

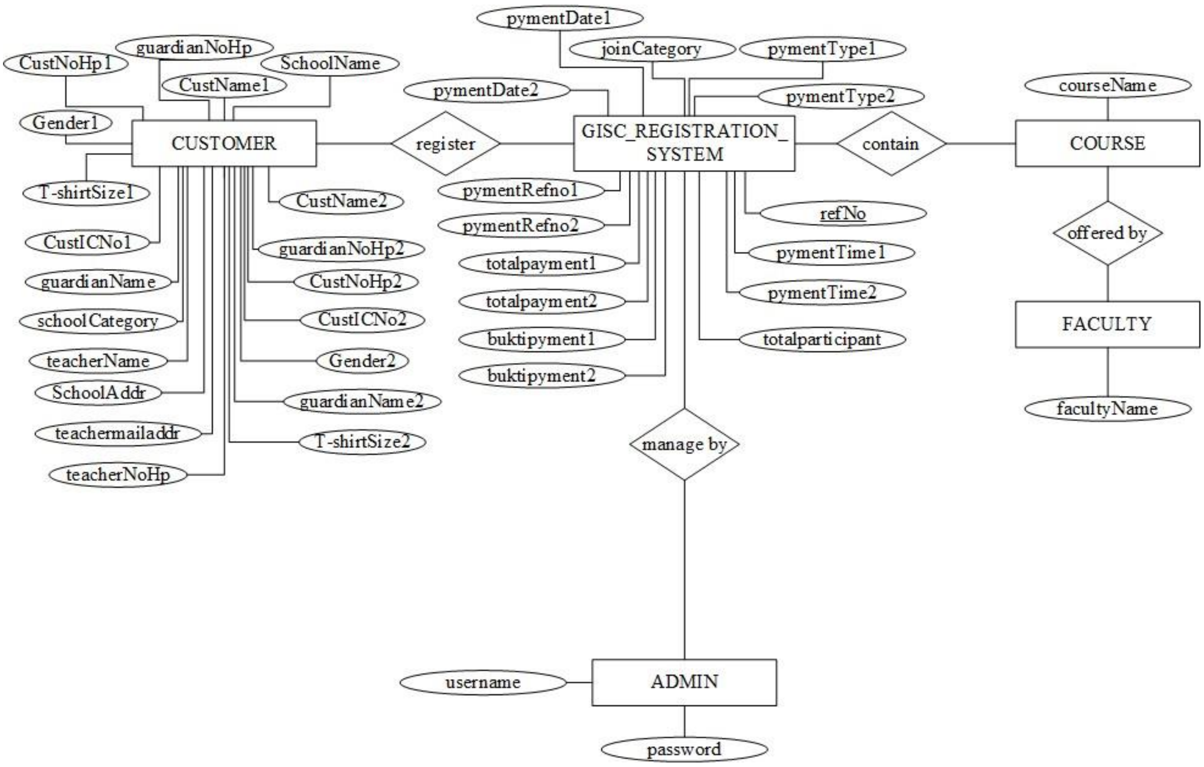

DELETE
START
Open Home Page
Click LOGIN button
Open Admin Login form
Insert correct username and pasword
Click button Login
Open Admin Page
Click button DELETE
Open Faculty list
Click any button Fakulti. ex: Fakulti Kejuruteraan
Open Fackulti Kejuruteraan database view page
Enter refNo in textboxrefNo.
Click button Search
Database view the specific data for entered refNo
Click button Delete Now
MessageBox : Data has been deleted
END

Flowchart 5: Delete

Explanation Delete flowchart:-

Flowchart 5 above shows how the data can be deleted by entering the unique reference number that provided while saving the details. The steps are system will run and load the Homepage, then user may click the LOGIN button to load CRUD page, where user can update, delete, view details and report analysis. After success login, the CRUD page will display and user need to click button DELETE to proceed the cancellation. Once clicked the Delete page form displayed, and user may view first the overall data then search the specific data to be deleted by entering reference number in the textbox provided and click Delete. The system will pop up message box that the data has been deleted. User may click view button want to look whether the data are surely deleted or not.

3.3 Entity Relationship Diagram



ER Diagram 1

3.4 Data Dictionary

Administration

Field name	Data type	Data format	Field size	Description	Example
username	nvarchar(50)		50	Admin username	admin1
password	nchar(10)	xxxx	10	Admin password	123

Customer

Field name	Data type	Data format	Field size	Description	Example
schoolName	nvarchar(50)		50	Student school's name	SMK SENAWANG
schoolAddress	nvarchar(100)		100	Student school's address	JALAN KUALA PILAH, N. SEMBILAN
schoolCategory	nvarchar(50)		50	Type of school	MRSM
teacherName	nvarchar(50)		50	Teacher's name	ZAID
teacherNoHp	int	xxxxxxxxxx		Teacher's phone number	0192345467
teacherEmail	nvarchar(50)		50	Teacher's email address	zaid@gmail.com
studentName1	nvarchar(50)		50	Participant's name	ATIQAH
studentICNo1	int	xxxxxxxxxxxx		Participant's IC number	961025065668
studentNoHp1	int	xxxxxxxxxx		Participant's phone number	0196719387

gender1	nvarchar(10)		10	Partcipant 1's gender	FEMALE
sizeTshirt1	nvarchar(4)		4	Participant 1 size tshirt	L
parentsorpenjagaName1	nvarchar(50)		50	Participant 1 parents or guidance name	HALIM
parentsorpenjagaNoHp1	int	xxxxxxxxxx		Participant 1 parents or guidance phone number	0123231232
studentName2	nvarchar(50)		50	Participant 2 name	RUBEN
studentICNo2	int	xxxxxxxxxxxx		Participant 2 IC number	911230015141
studentNoHp2	int	xxxxxxxxxx		Participant 2 phone number	0144443242
gender2	nvarchar(10)		10	Participant 2 gender	MALE
sizeTshirt2	nvarchar(4)		4	Participant 2 size tshirt	S
parentsorpenjagaName2	nvarchar(50)		50	Participant 2 parents or guider 's name	RAVINDRAN
parentsorpenjagaNoHp2	int	xxxxxxxxxx		Parents or guider's phone number	0198765432

Course

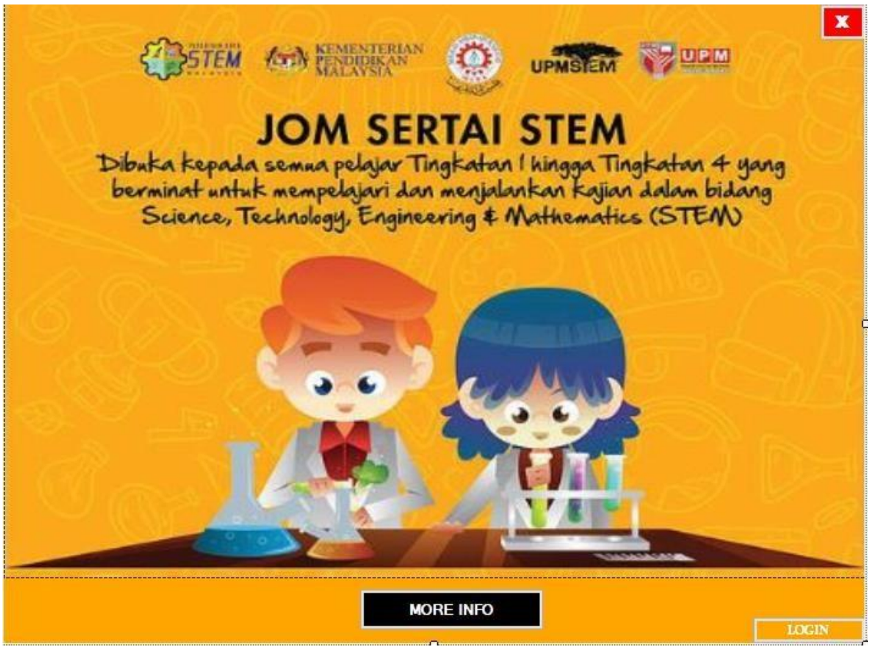
Field name	Data type	Data format	Field size	Description	Example
courseName	nvarchar(50)		50	Name of courses	Kejuruteraan Mekanikal

GISC_Registration_System

Field name	Data type	Data format	Field size	Description	Example
refNo	int				0001
pymentType1	nvarchar(20)		20	Participant 1 payment type	CDM
pymentDate1	nvarchar(50)		50	Participant 1 payment date	12/12/2019
pymentTime1	varchar(50)		50	Participant 1 payment time	12:30
pymentRefno1	int			Participant 1 payment reference no	1113
totalpayment1	int			Participant 1 total payment	300.00
buktipyment1	nvarchar(20)		20	Participant 1 payment proof	CDM
pymentType2	nvarchar(20)		20	Participant 2 payment type	Transfer
pymentDate2	nvarchar(50)		50	Participant 2 payment date	03/09/2019
pymentTime2	varchar(50)		50	Participant 2 payment time	16:30
pymentRefno2	int			Participant 2 payment reference no.	1109
totalpayment2	int			Participant 2 total payment	250

buktipyment2	nvarchar(20)		20	Participant 2 proof payment	Online transaction
joinCategory	nvarchar(20)		20	Category in joining program	Individu/Kumpulan

3.5 Interface Design

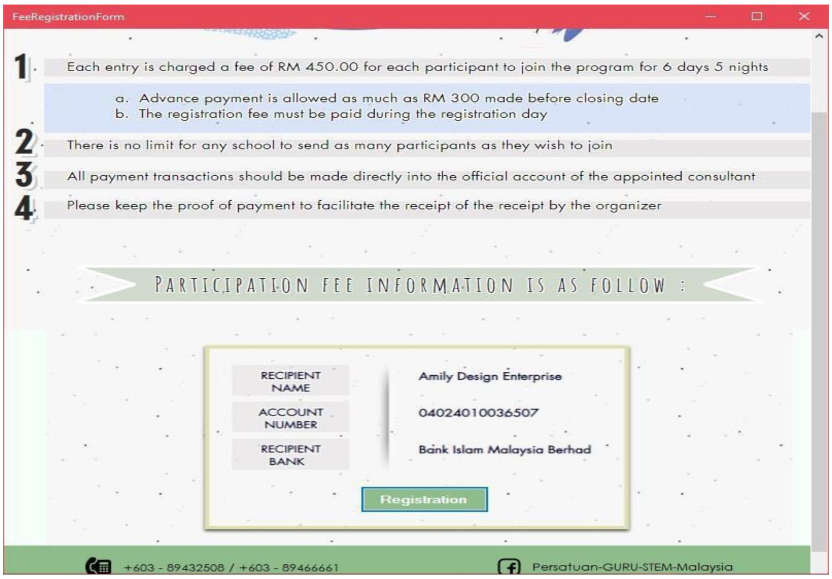


JOM SERTAI STEM

Dibuka kepada semua pelajar Tingkatan I hingga Tingkatan 4 yang berminat untuk mempelajari dan menjalankan kajian dalam bidang Science, Technology, Engineering & Mathematics (STEM)

Interface 1: Homepage

Interface 1 shown above act as main page and called Homepage. From here user which is the participants or customer may click button ‘MORE INFO’ in order to proceed the registration process. As for ‘LOGIN’ button is used for administration only for log in account.



FeeRegistrationForm

1. Each entry is charged a fee of RM 450.00 for each participant to join the program for 6 days 5 nights
   a. Advance payment is allowed as much as RM 300 made before closing date
   b. The registration fee must be paid during the registration day
2. There is no limit for any school to send as many participants as they wish to join
3. All payment transactions should be made directly into the official account of the appointed consultant
4. Please keep the proof of payment to facilitate the receipt of the receipt by the organizer

PARTICIPATION FEE INFORMATION IS AS FOLLOW :

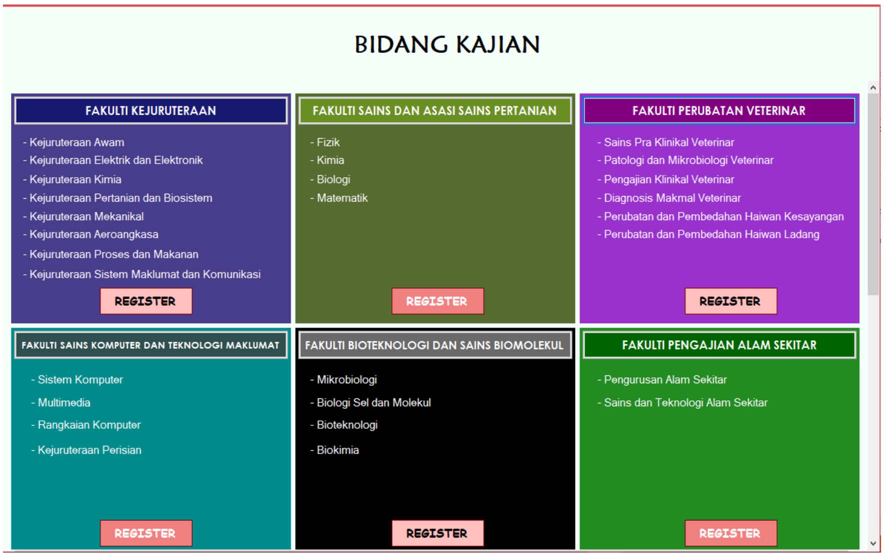
RECIPIENT NAME	Amily Design Enterprise
ACCOUNT NUMBER	04024010036507
RECIPIENT BANK	Bank Islam Malaysia Berhad

☎ +603 - 89432508 / +603 - 89466661

Persatuan-GURU-STEM-Malaysia

Interface 2: Fee and Registration Info

Interface 2 shown above is a fee and registration information page. Customer and participants may read the information about fees and how to do the registration. There is telephone number at the bottom of the page. If customers have any enquiries, they may call the hotline number. After that, they will just only to proceed the registration by clicking the ‘Registration’ button.



BIDANG KAJIAN

FAKULTI KEJURUTERAAN

- Kejuruteraan Awam
- Kejuruteraan Elektrik dan Elektronik
- Kejuruteraan Kimia
- Kejuruteraan Pertanian dan Biosistem
- Kejuruteraan Mekanikal
- Kejuruteraan Aeroangkasa
- Kejuruteraan Proses dan Makanan
- Kejuruteraan Sistem Maklumat dan Komunikasi

FAKULTI SAINS DAN ASASI SAINS PERTANIAN

- Fizik
- Kimia
- Biologi
- Matematik

FAKULTI PERUBATAN VETERINAR

- Sains Pra Klinikal Veterinar
- Patologi dan Mikrobiologi Veterinar
- Pengajian Klinikal Veterinar
- Diagnosis Makmal Veterinar
- Perubatan dan Pembedahan Haiwan Kesayangan
- Perubatan dan Pembedahan Haiwan Ladang

FAKULTI SAINS KOMPUTER DAN TEKNOLOGI MAKLUMAT

- Sistem Komputer
- Multimedia
- Rangkaian Komputer
- Kejuruteraan Perisian

FAKULTI BIOTEKNOLOGI DAN SAINS BIOMOLEKUL

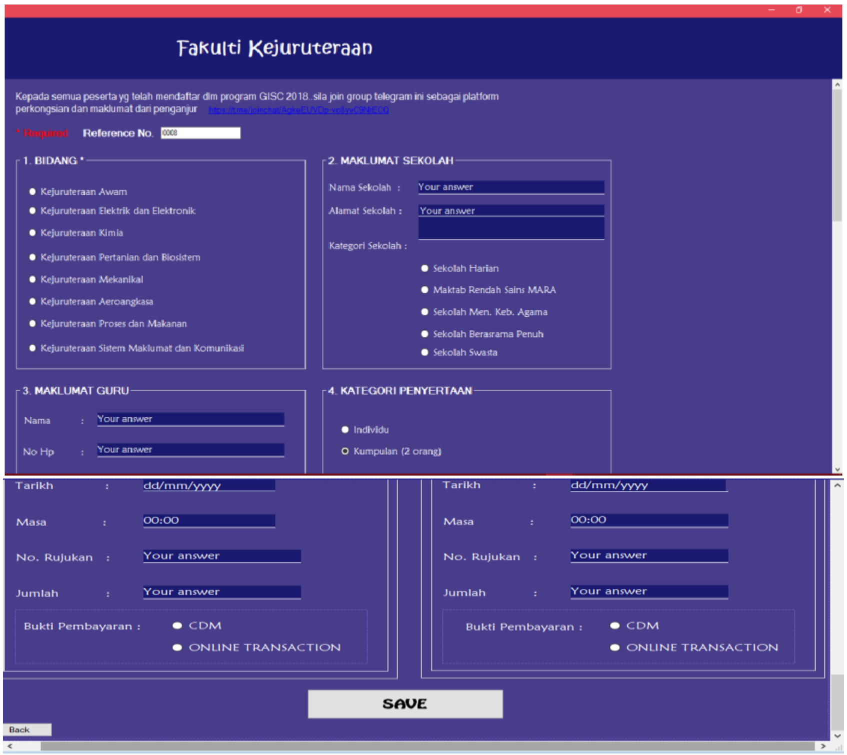
- Mikrobiologi
- Biologi Sel dan Molekul
- Bioteknologi
- Biokimia

FAKULTI PENGAJIAN ALAM SEKITAR

- Pengurusan Alam Sekitar
- Sains dan Teknologi Alam Sekitar

Interface 3: Study Field page

Interface 3 shown above is a Study Field page. Customer and participants may read the information about courses that available in each faculty. User may click any ‘REGISTER’ button suite their selves in what course that they want to register and participate. Each of ‘REGISTER’ button will lead to the registration form page.



Fakulti Kejuruteraan

Kepada semua peserta yg telah mendaftar dlm program GISC 2018..sila join group telegram ini sebagai platform perkongsian dan maklumat dari penganjur https://t.me/joinchat/AgkeEUVDp-vo8yvC9NrECQ

* Required Reference No. 0008

1. BIDANG *

● Kejuruteraan Awam
● Kejuruteraan Elektrik dan Elektronik
● Kejuruteraan Kimia
● Kejuruteraan Pertanian dan Biosistem
● Kejuruteraan Mekanikal
● Kejuruteraan Aeroangkasa
● Kejuruteraan Proses dan Makanan
● Kejuruteraan Sistem Maklumat dan Komunikasi

2. MAKLUMAT SEKOLAH

Nama Sekolah : Your answer
Alamat Sekolah : Your answer
Kategori Sekolah :
● Sekolah Harian
● Maktab Rendah Sains MARA
● Sekolah Men. Keb. Agama
● Sekolah Berasrama Penuh
● Sekolah Swasta

3. MAKLUMAT GURU

Nama : Your answer
No Hp : Your answer

4. KATEGORI PENYERTAAN

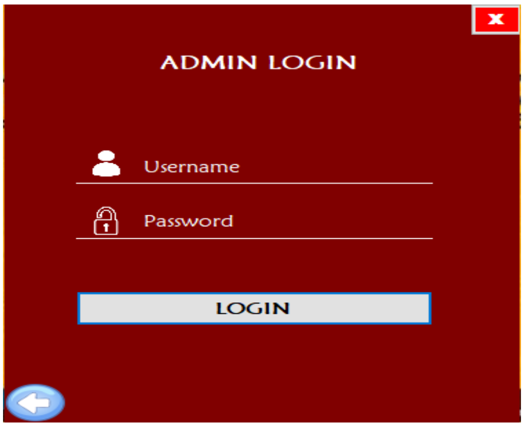
● Individu
○ Kumpulan (2 orang)

Tarikh : dd/mm/yyyy
Masa : 00:00
No. Rujukan : Your answer
Jumlah : Your answer
Bukti Pembayaran : ● CDM ● ONLINE TRANSACTION

Tarikh : dd/mm/yyyy
Masa : 00:00
No. Rujukan : Your answer
Jumlah : Your answer
Bukti Pembayaran : ● CDM ● ONLINE TRANSACTION

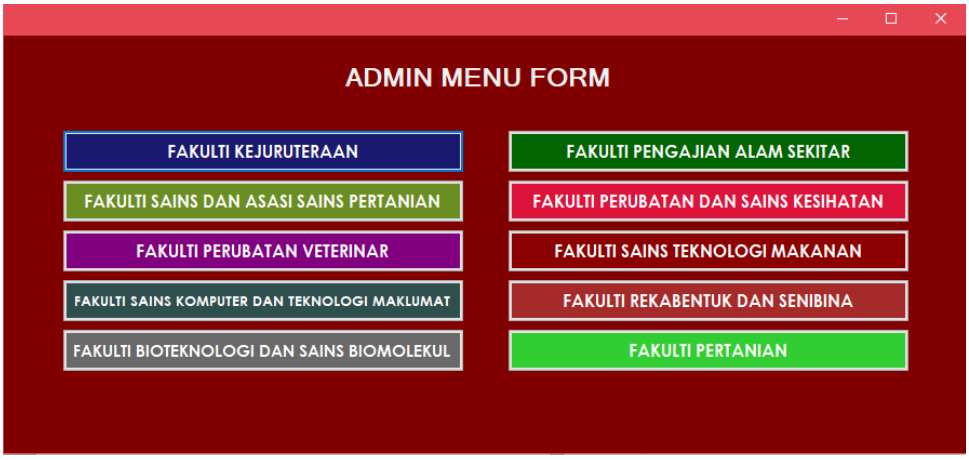
Interface 4: Registration Form page

Interface 4 shown above is a Registration Form page. Customers and participants must fill in all the required blanks and save all the details by clicking the 'SAVE' button. Then a message box will pop up asking whether to submit any other response. If not, they may click No and back to homepage and exit or else they may insert other details required.



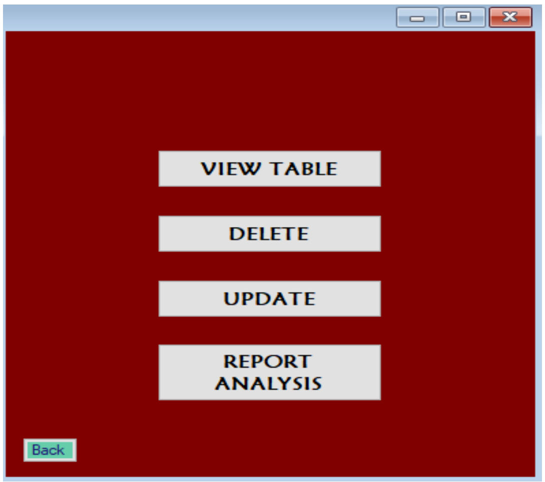
Interface 5: Admin Login Page

Interface 5 shown above is a Admin Login Page. This page is for authorization only. If user is an admin, he/she may enter correct username and password and click 'LOGIN' button. If the username or password are not correct the system will pop up error messages. Click button back arrow in order to go to Homepage.


ADMIN MENU FORM
FAKULTI KEJURUTERAAN
FAKULTI PENGAJIAN ALAM SEKITAR
FAKULTI SAINS DAN ASASI SAINS PERTANIAN
FAKULTI PERUBATAN DAN SAINS KESIHATAN
FAKULTI PERUBATAN VETERINAR
FAKULTI SAINS TEKNOLOGI MAKANAN
FAKULTI SAINS KOMPUTER DAN TEKNOLOGI MAKLUMAT
FAKULTI REKABENTUK DAN SENIBINA
FAKULTI BIOTEKNOLOGI DAN SAINS BIOMOLEKUL
FAKULTI PERTANIAN

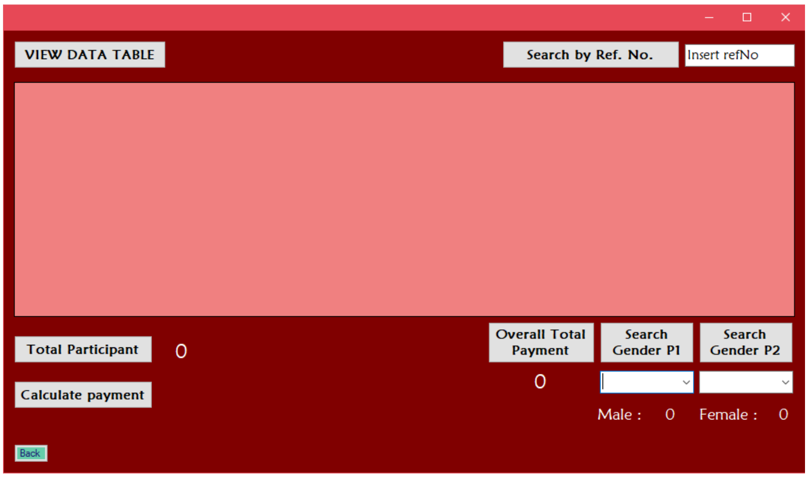
Interface 6: Admin Menu page

Interface 6 shown above is an Admin Menu Page. This page able the user to choose what faculty they want to survey or make view for the raw data, update, delete, search and view the report analysis. Each has different data saved.



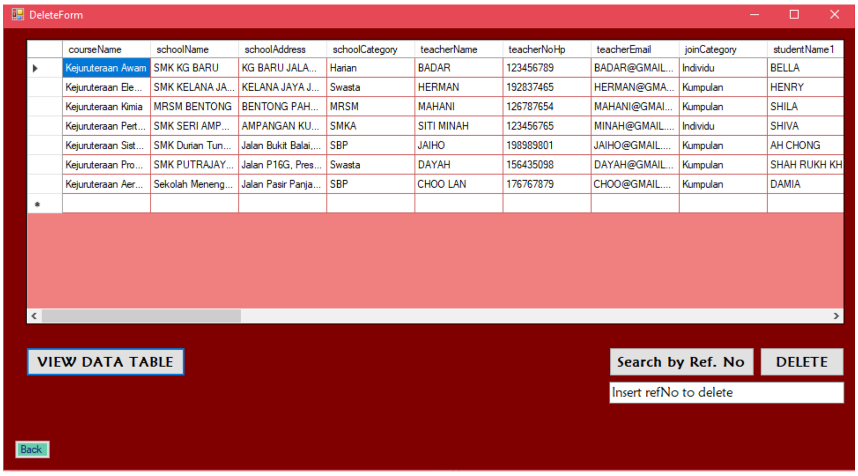
Interface 7: Admin Options page

Interface 7 shown above is an Admin Options page. This page able the admin to choose what he/she want to do with the data. User may view the raw data table if they want to view the database by clicked the VIEW TABLE button. As for deleting data, admin may click the DELETE button. While for updating the data in the database, admin may click the UPDATE button options.



Interface 8: View Data Table page

Interface 8 shown above is a View Data Table page. This page able the admin to view the data table in the database. User may click button Total Participant in order to calculate the set of data available in database. As for calculate the total number of gender participants 1, user may choose gender in the dropdown list and click the Search Gender P1 button. As for calculate the total number of gender participants 2, user may choose gender in the dropdown list and click the Search Gender P2 button and then it will display the total number of male and female gender. User may calculate the Overall Total Payment in the table by clicking the Calculate Payment button first then followed by Overall Total Payment button. And it will display the exact values.



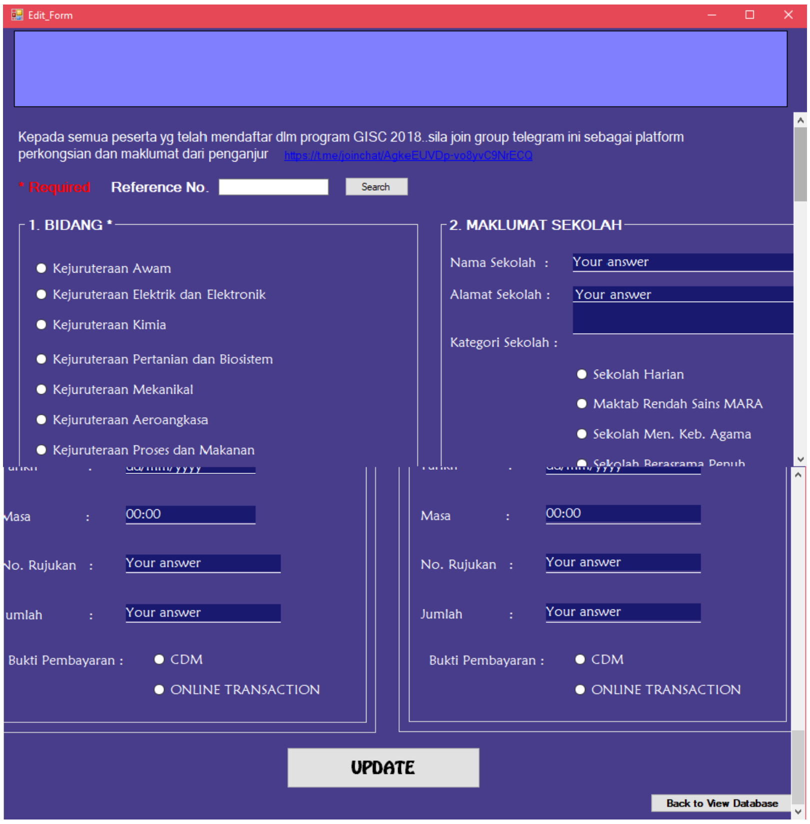
DeleteForm

	courseName	schoolName	schoolAddress	schoolCategory	teacherName	teacherNoHp	teacherEmail	joinCategory	studentName1
▶	Kejuruteraan Awam	SMK KG BARU	KG BARU JALA...	Harian	BADAR	123456789	BADAR@GMAIL...	Individu	BELLA
	Kejuruteraan Ele...	SMK KELANA JA...	KELANA JAYA J...	Swasta	HERMAN	192837465	HERMAN@GMA...	Kumpulan	HENRY
	Kejuruteraan Kimia	MRSM BENTONG	BENTONG PAH...	MRSM	MAHANI	126787654	MAHANI@GMAI...	Kumpulan	SHILA
	Kejuruteraan Pert...	SMK SERI AMP...	AMPANGAN KU...	SMKA	SITI MINAH	123456765	MINAH@GMAIL....	Individu	SHIVA
	Kejuruteraan Sist...	SMK Durian Tun...	Jalan Bukit Balai,...	SBP	JAIHO	198989801	JAIHO@GMAIL....	Kumpulan	AH CHONG
	Kejuruteraan Pro...	SMK PUTRAJAY...	Jalan P16G, Pres...	Swasta	DAYAH	156435098	DAYAH@GMAIL...	Kumpulan	SHAH RUKH KH
	Kejuruteraan Aer...	Sekolah Meneng...	Jalan Pasir Panja...	SBP	CHOO LAN	176767879	CHOO@GMAIL....	Kumpulan	DAMIA
*									

Insert refNo to delete

Interface 9: Delete page

Interface 9 shown above is a Delete page. This page able the admin to delete data in database. User may view the data table and enter the reference no in order to delete all details that related to the reference number by clicking the DELETE button.



Edit_Form

Kepada semua peserta yg telah mendaftar dlm program GISC 2018..sila join group telegram ini sebagai platform perkongsian dan maklumat dari penganjur https://t.me/joinchat/AgkeEUVDp-vo8yvC9NrECQ

* Required Reference No.

1. BIDANG *

- ● Kejuruteraan Awam
- ● Kejuruteraan Elektrik dan Elektronik
- ● Kejuruteraan Kimia
- ● Kejuruteraan Pertanian dan Biosistem
- ● Kejuruteraan Mekanikal
- ● Kejuruteraan Aeroangkasa
- ● Kejuruteraan Proses dan Makanan

2. MAKLUMAT SEKOLAH

Nama Sekolah : Your answer

Alamat Sekolah : Your answer

Kategori Sekolah :

- ● Sekolah Harian
- ● Maktab Rendah Sains MARA
- ● Sekolah Men. Keb. Agama
- ● Sekolah Berasrama Penuh

Masa : 00:00

No. Rujukan : Your answer

Jumlah : Your answer

Bukti Pembayaran : ● CDM ● ONLINE TRANSACTION

Masa : 00:00

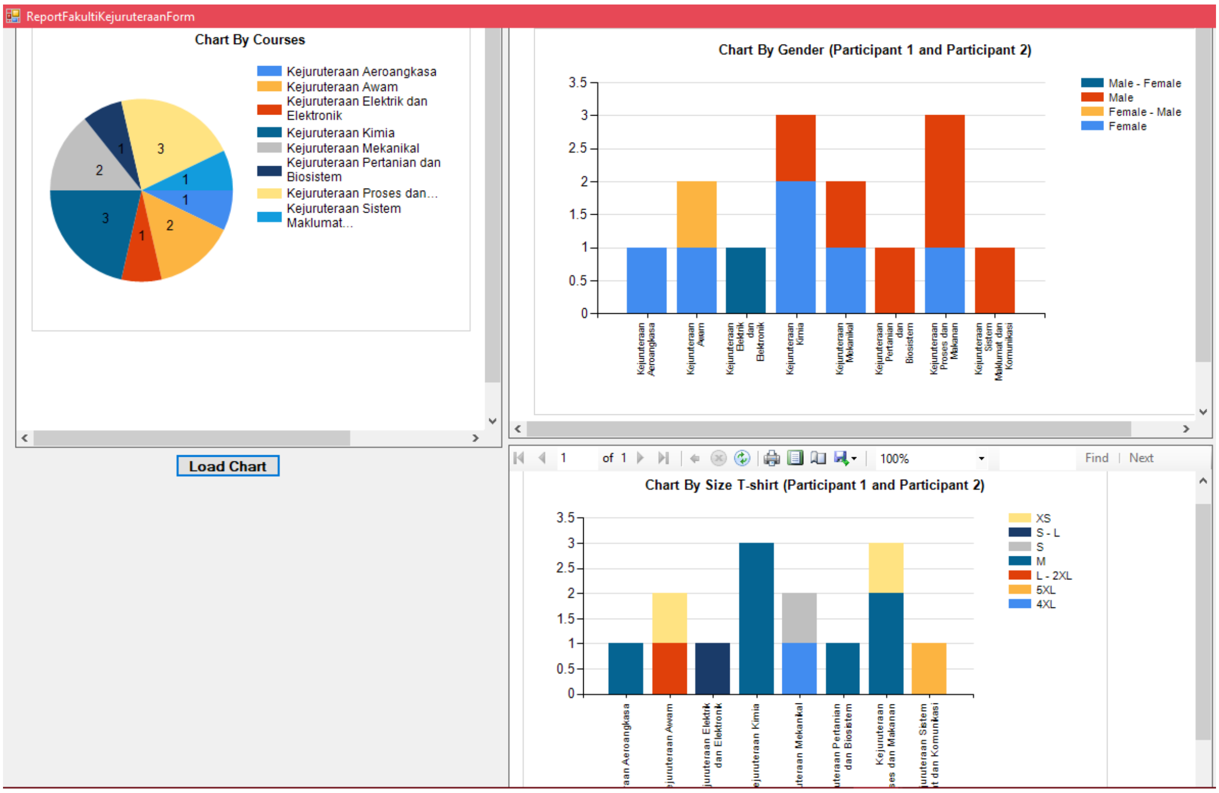
No. Rujukan : Your answer

Jumlah : Your answer

Bukti Pembayaran : ● CDM ● ONLINE TRANSACTION

Interface 10: Update page

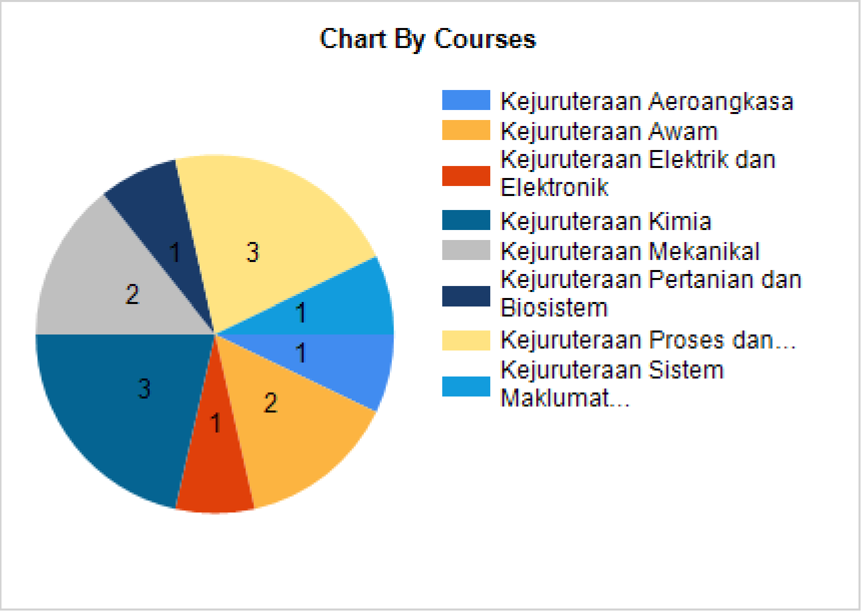
Interface 10 shown above is an Update page. This page able the admin to update data in database. User may view the specific data table by entering the reference number and click the Search button. Then update all the details by clicking the UPDATE button.


ReportFakultiKejuruteraanForm
Chart By Courses
Kejuruteraan Aeroangkasa
Kejuruteraan Awam
Kejuruteraan Elektrik dan Elektronik
Kejuruteraan Kimia
Kejuruteraan Mekanikal
Kejuruteraan Pertanian dan Biosistem
Kejuruteraan Proses dan...
Kejuruteraan Sistem Maklumat...
1
3
2
1
1
3
1
2
Load Chart
Chart By Gender (Participant 1 and Participant 2)
Male - Female
Male
Female - Male
Female
3.5
3
2.5
2
1.5
1
0.5
0
Kejuruteraan Aeroangkasa
Kejuruteraan Awam
Kejuruteraan Elektrik dan Elektronik
Kejuruteraan Kimia
Kejuruteraan Mekanikal
Kejuruteraan Pertanian dan Biosistem
Kejuruteraan Proses dan Makanan
Kejuruteraan Sistem Maklumat dan Komunikasi
1
of 1
100%
Find | Next
Chart By Size T-shirt (Participant 1 and Participant 2)
XS
S - L
S
M
L - 2XL
5XL
4XL
3.5
3
2.5
2
1.5
1
0.5
0
aan Aeroangkasa
juruteraan Awam
juruteraan Elektrik dan Elektronik
ejuruteraan Kimia
uteraan Mekanikal
uteraan Pertanian dan Biosistem
Kejuruteraan ses dan Makanan
juruteraan Sistem t dan Komunikasi

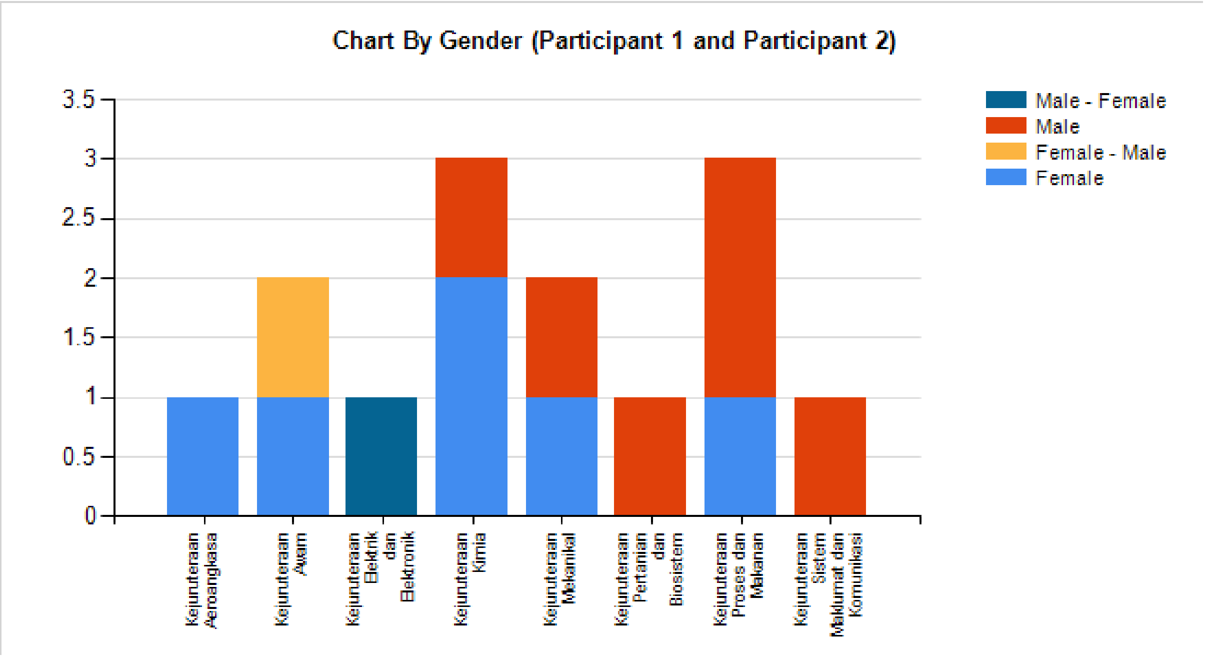
Interface 11: Report Analysis page

Interface 11 shown above is a Report Analysis page. For example: Fakulti Kejuruteraan. This page able the admin to view some report analysis based on Courses, by gender and size T-shirt.

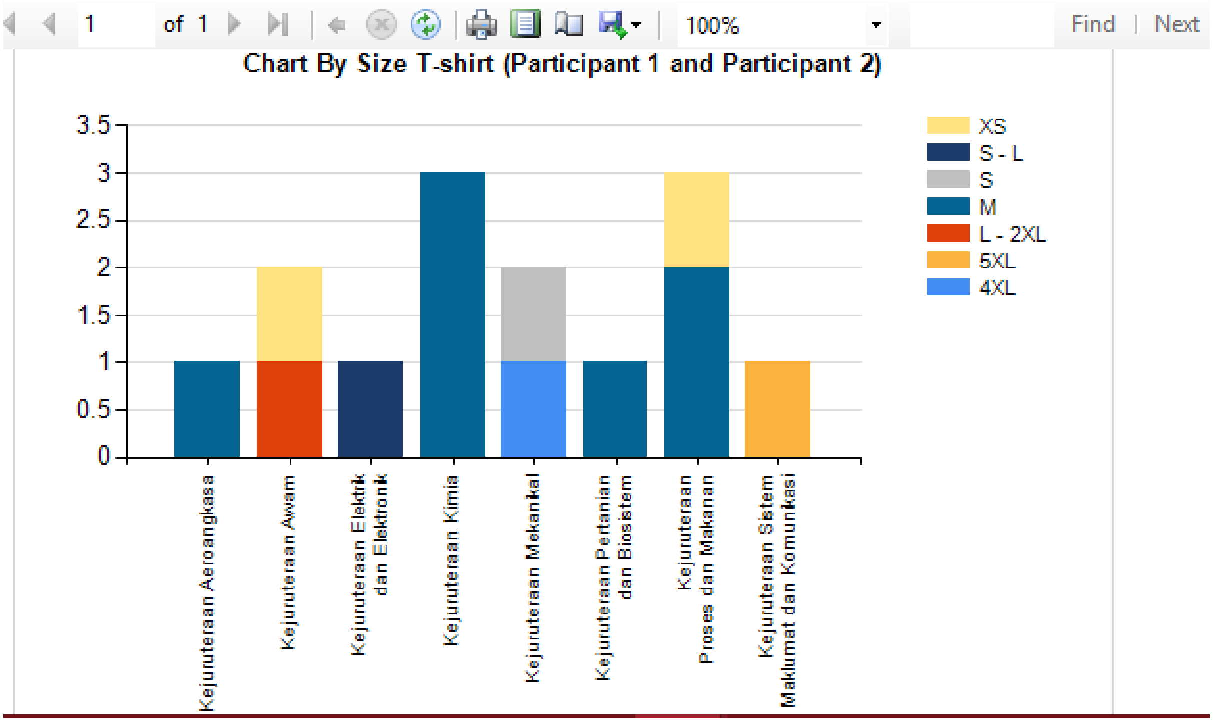
From here they will able to summaries all the data easily.



Pie Chart 1: Filtered by Courses and Number of Set Participants



Bar Chart 1: Filtered by Courses and Number of Set Participants with gender


1
of 1
100%
Find | Next
Chart By Size T-shirt (Participant 1 and Participant 2)
XS
S - L
S
M
L - 2XL
5XL
4XL
3.5
3
2.5
2
1.5
1
0.5
0
Kejuruteraan Aeroangkasa
Kejuruteraan Awam
Kejuruteraan Elektrik dan Elektronik
Kejuruteraan Kimia
Kejuruteraan Mekanikal
Kejuruteraan Pertanian dan Biosistem
Kejuruteraan Proses dan Makanan
Kejuruteraan Sistem Maklumat dan Komunikasi

Bar Chart 1: Filtered by Courses and Number of Set Participants with Size T-shirt

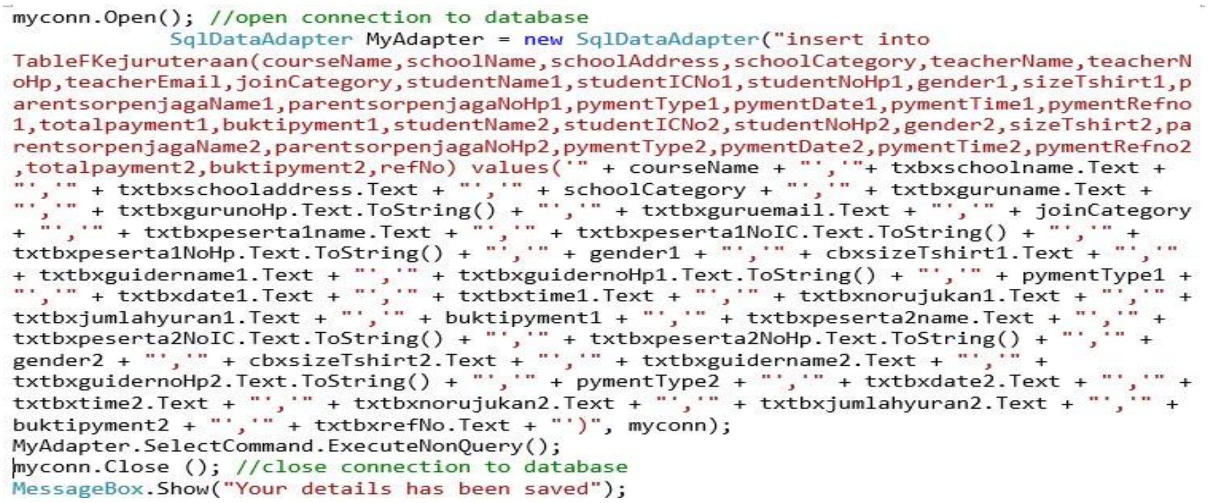
CHAPTER 4: IMPLEMENTATION

4.1 Introduction

This section is explain in details regarding the elements of programming technique such as function, array, selection, control and pointer including the name conversion. The codes given below with the explanation on how the error-handling is being implemented.

4.2 Storing, Retrieve and Manage of Data

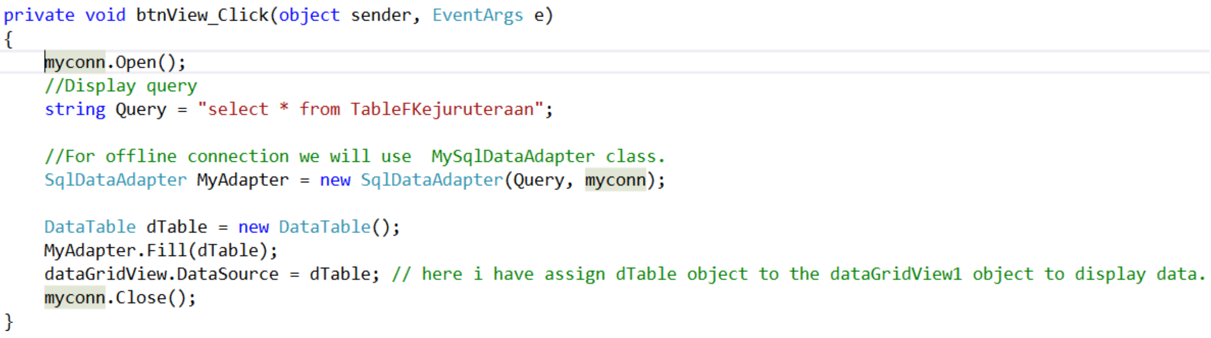
Code 1 shown below how the details been insert in the database for Fakulti Kejuruteraan courses.



```
myconn.Open(); //open connection to database
            SqlDataAdapter MyAdapter = new SqlDataAdapter("insert into
TableFKejuruteraan(courseName,schoolName,schoolAddress,schoolCategory,teacherName,teacherN
oHp,teacherEmail,joinCategory,studentName1,studentICNo1,studentNoHp1,gender1,sizeTshirt1,p
arentsorpenjagaName1,parentsorpenjagaNoHp1,pymentType1,pymentDate1,pymentTime1,pymentRefno
1,totalpayment1,buktipyment1,studentName2,studentICNo2,studentNoHp2,gender2,sizeTshirt2,pa
rentsorpenjagaName2,parentsorpenjagaNoHp2,pymentType2,pymentDate2,pymentTime2,pymentRefno2
,totalpayment2,buktipyment2,refNo) values('" + courseName + "','"+ txbxschoolname.Text +
"','" + txtbxschooladdress.Text + "','" + schoolCategory + "','" + txtbxguruname.Text +
"','" + txtbxgurunoHp.Text.ToString() + "','" + txtbxguruemail.Text + "','" + joinCategory
+ "','" + txtbxpeserta1name.Text + "','" + txtbxpeserta1NoIC.Text.ToString() + "','" +
txtbxpeserta1NoHp.Text.ToString() + "','" + gender1 + "','" + cbxsizeTshirt1.Text + "','"
+ txtbxguidername1.Text + "','" + txtbxguidernoHp1.Text.ToString() + "','" + pymentType1 +
"','" + txtbxdate1.Text + "','" + txtbxtime1.Text + "','" + txtbxnorujukan1.Text + "','" +
txtbxjumlahyuran1.Text + "','" + buktipyment1 + "','" + txtbxpeserta2name.Text + "','" +
txtbxpeserta2NoIC.Text.ToString() + "','" + txtbxpeserta2NoHp.Text.ToString() + "','" +
gender2 + "','" + cbxsizeTshirt2.Text + "','" + txtbxguidername2.Text + "','" +
txtbxguidernoHp2.Text.ToString() + "','" + pymentType2 + "','" + txtbxdate2.Text + "','" +
txtbxtime2.Text + "','" + txtbxnorujukan2.Text + "','" + txtbxjumlahyuran2.Text + "','" +
buktipyment2 + "','" + txtbxrefNo.Text + "')", myconn);
MyAdapter.SelectCommand.ExecuteNonQuery();
myconn.Close (); //close connection to database
MessageBox.Show("Your details has been saved");
```

Code 1: Insert and save data into database

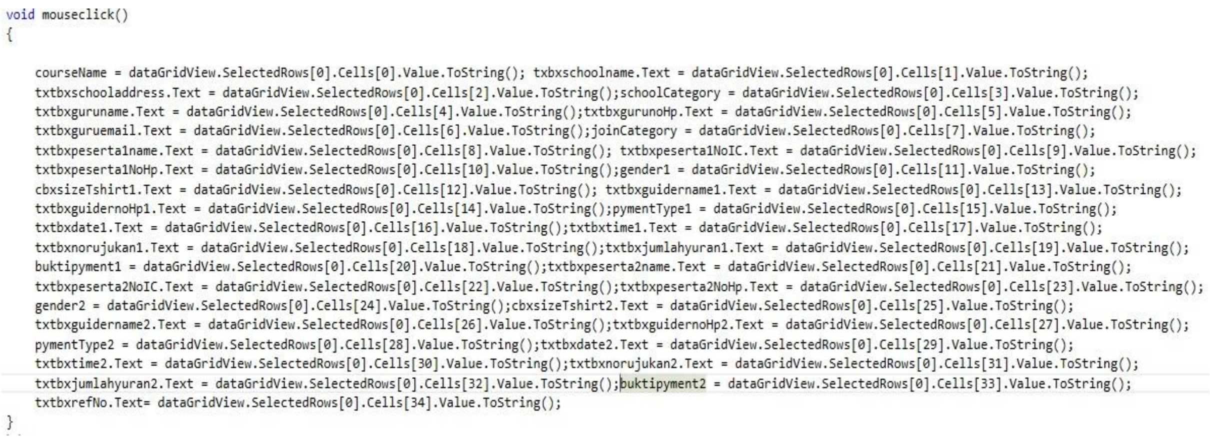
Code 2 shown below how the details can be deleted from the database for Fakulti Kejuruteraan courses.



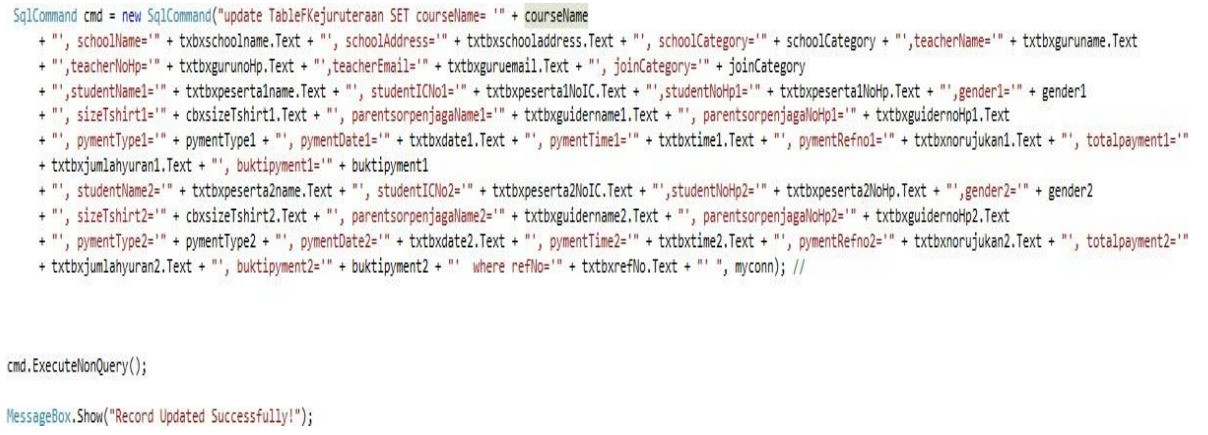
```
private void btnView_Click(object sender, EventArgs e)
{
    myconn.Open();
    //Display query
    string Query = "select * from TableFKejuruteraan";

    //For offline connection we will use  MySqlDataAdapter class.
    SqlDataAdapter MyAdapter = new SqlDataAdapter(Query, myconn);

    DataTable dTable = new DataTable();
    MyAdapter.Fill(dTable);
    dataGridView.DataSource = dTable; // here i have assign dTable object to the dataGridView1 object to display data.
    myconn.Close();
}
```

Code 2: Delete data from database

Code 3 shown below how the details can be read and updated from the database for Fakulti Kejuruteraan courses.



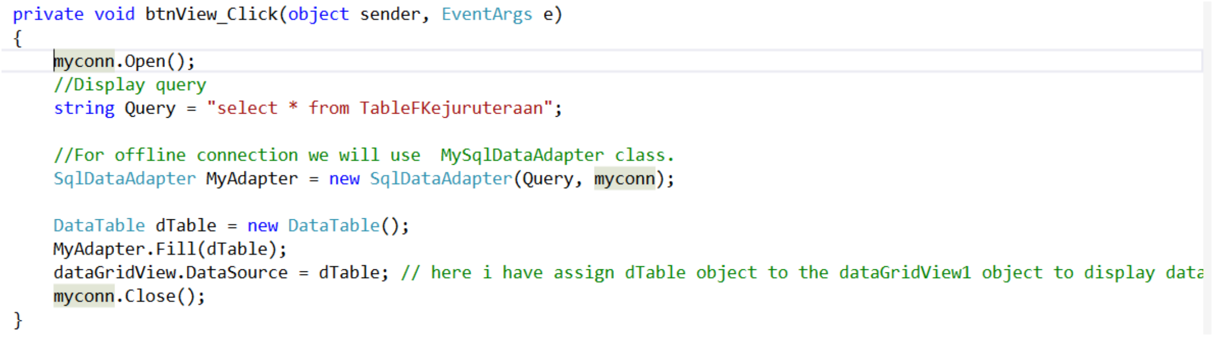
```
void mouseclick()
{

    courseName = dataGridView.SelectedRows[0].Cells[0].Value.ToString(); txbxschoolname.Text = dataGridView.SelectedRows[0].Cells[1].Value.ToString();
    txtbxschooladdress.Text = dataGridView.SelectedRows[0].Cells[2].Value.ToString();schoolCategory = dataGridView.SelectedRows[0].Cells[3].Value.ToString();
    txtbxguruname.Text = dataGridView.SelectedRows[0].Cells[4].Value.ToString();txtbxgurunoHp.Text = dataGridView.SelectedRows[0].Cells[5].Value.ToString();
    txtbxguruemail.Text = dataGridView.SelectedRows[0].Cells[6].Value.ToString();joinCategory = dataGridView.SelectedRows[0].Cells[7].Value.ToString();
    txtbxpeserta1name.Text = dataGridView.SelectedRows[0].Cells[8].Value.ToString(); txtbxpeserta1NoIC.Text = dataGridView.SelectedRows[0].Cells[9].Value.ToString();
    txtbxpeserta1NoHp.Text = dataGridView.SelectedRows[0].Cells[10].Value.ToString();gender1 = dataGridView.SelectedRows[0].Cells[11].Value.ToString();
    cbxsizeTshirt1.Text = dataGridView.SelectedRows[0].Cells[12].Value.ToString(); txtbxguidername1.Text = dataGridView.SelectedRows[0].Cells[13].Value.ToString();
    txtbxguidernoHp1.Text = dataGridView.SelectedRows[0].Cells[14].Value.ToString();pymentType1 = dataGridView.SelectedRows[0].Cells[15].Value.ToString();
    txtbxdate1.Text = dataGridView.SelectedRows[0].Cells[16].Value.ToString();txtbxtime1.Text = dataGridView.SelectedRows[0].Cells[17].Value.ToString();
    txtbxnorujukan1.Text = dataGridView.SelectedRows[0].Cells[18].Value.ToString();txtbxjumlahyuran1.Text = dataGridView.SelectedRows[0].Cells[19].Value.ToString();
    buktipyment1 = dataGridView.SelectedRows[0].Cells[20].Value.ToString();txtbxpeserta2name.Text = dataGridView.SelectedRows[0].Cells[21].Value.ToString();
    txtbxpeserta2NoIC.Text = dataGridView.SelectedRows[0].Cells[22].Value.ToString();txtbxpeserta2NoHp.Text = dataGridView.SelectedRows[0].Cells[23].Value.ToString();
    gender2 = dataGridView.SelectedRows[0].Cells[24].Value.ToString();cbxsizeTshirt2.Text = dataGridView.SelectedRows[0].Cells[25].Value.ToString();
    txtbxguidername2.Text = dataGridView.SelectedRows[0].Cells[26].Value.ToString();txtbxguidernoHp2.Text = dataGridView.SelectedRows[0].Cells[27].Value.ToString();
    pymentType2 = dataGridView.SelectedRows[0].Cells[28].Value.ToString();txtbxdate2.Text = dataGridView.SelectedRows[0].Cells[29].Value.ToString();
    txtbxtime2.Text = dataGridView.SelectedRows[0].Cells[30].Value.ToString();txtbxnorujukan2.Text = dataGridView.SelectedRows[0].Cells[31].Value.ToString();
    txtbxjumlahyuran2.Text = dataGridView.SelectedRows[0].Cells[32].Value.ToString();buktipyment2 = dataGridView.SelectedRows[0].Cells[33].Value.ToString();
    txtbxrefNo.Text= dataGridView.SelectedRows[0].Cells[34].Value.ToString();
}
```



```
SqlCommand cmd = new SqlCommand("update TableFKejuruteraan SET courseName= '" + courseName
    + "', schoolName='" + txbxschoolname.Text + "', schoolAddress='" + txtbxschooladdress.Text + "', schoolCategory='" + schoolCategory + "',teacherName='" + txtbxguruname.Text
    + "',teacherNoHp='" + txtbxgurunoHp.Text + "',teacherEmail='" + txtbxguruemail.Text + "', joinCategory='" + joinCategory
    + "',studentName1='" + txtbxpeserta1name.Text + "', studentICNo1='" + txtbxpeserta1NoIC.Text + "',studentNoHp1='" + txtbxpeserta1NoHp.Text + "',gender1='" + gender1
    + "', sizeTshirt1='" + cbxsizeTshirt1.Text + "', parentsorpenjagaName1='" + txtbxguidername1.Text + "', parentsorpenjagaNoHp1='" + txtbxguidernoHp1.Text
    + "', pymentType1='" + pymentType1 + "', pymentDate1='" + txtbxdate1.Text + "', pymentTime1='" + txtbxtime1.Text + "', pymentRefno1='" + txtbxnorujukan1.Text + "', totalpayment1='"
    + txtbxjumlahyuran1.Text + "', buktipyment1='" + buktipyment1
    + "', studentName2='" + txtbxpeserta2name.Text + "', studentICNo2='" + txtbxpeserta2NoIC.Text + "',studentNoHp2='" + txtbxpeserta2NoHp.Text + "',gender2='" + gender2
    + "', sizeTshirt2='" + cbxsizeTshirt2.Text + "', parentsorpenjagaName2='" + txtbxguidername2.Text + "', parentsorpenjagaNoHp2='" + txtbxguidernoHp2.Text
    + "', pymentType2='" + pymentType2 + "', pymentDate2='" + txtbxdate2.Text + "', pymentTime2='" + txtbxtime2.Text + "', pymentRefno2='" + txtbxnorujukan2.Text + "', totalpayment2='"
    + txtbxjumlahyuran2.Text + "', buktipyment2='" + buktipyment2 + "'  where refNo='" + txtbxrefNo.Text + "' ", myconn); //



cmd.ExecuteNonQuery();

MessageBox.Show("Record Updated Successfully!");
```

Code 3: Read and Update data into database

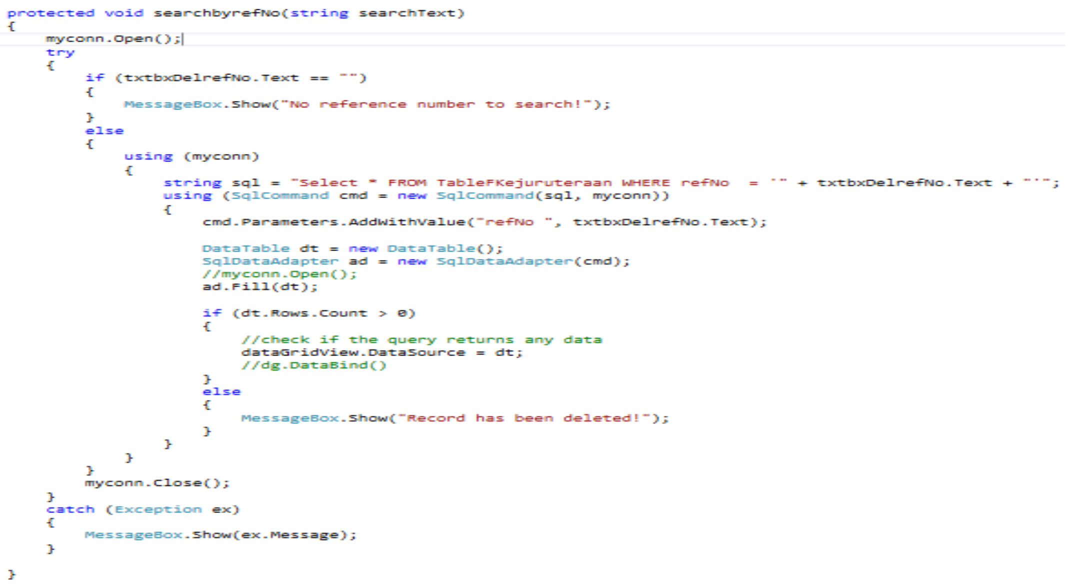
Code 4 shown below how the details can be viewed display the database for Fakulti Kejuruteraan courses in datagridview.



```
private void btnView_Click(object sender, EventArgs e)
{
    myconn.Open();
    //Display query
    string Query = "select * from TableFKejuruteraan";

    //For offline connection we will use  MySqlDataAdapter class.
    SqlDataAdapter MyAdapter = new SqlDataAdapter(Query, myconn);

    DataTable dTable = new DataTable();
    MyAdapter.Fill(dTable);
    dataGridView.DataSource = dTable; // here i have assign dTable object to the dataGridView1 object to display data
    myconn.Close();
}
```

Code 4: View data from database

Code 5 shown below how the details can be searched and display the database for ‘ex: Fakulti Kejuruteraan courses’ in datagridview.

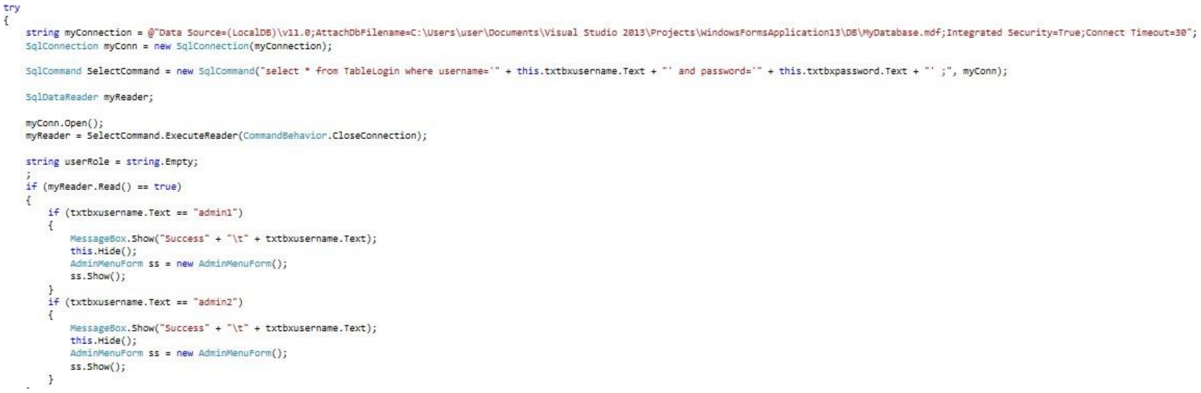


```
protected void searchbyrefNo(string searchText)
{
    myconn.Open();
    try
    {
        if (txtbxDelrefNo.Text == "")
        {
            MessageBox.Show("No reference number to search!");
        }
        else
        {
            using (myconn)
            {
                string sql = "Select * FROM TableFKejuruteraan WHERE refNo  = '" + txtbxDelrefNo.Text + "'";
                using (SqlCommand cmd = new SqlCommand(sql, myconn))
                {
                    cmd.Parameters.AddWithValue("refNo ", txtbxDelrefNo.Text);

                    DataTable dt = new DataTable();
                    SqlDataAdapter ad = new SqlDataAdapter(cmd);
                    //myconn.Open();
                    ad.Fill(dt);

                    if (dt.Rows.Count > 0)
                    {
                        //check if the query returns any data
                        dataGridView.DataSource = dt;
                        //dg.DataBind()
                    }
                    else
                    {
                        MessageBox.Show("Record has been deleted!");
                    }
                }
            }
        }
        myconn.Close();
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
```

Code 5: Search data from database

4.3 Security

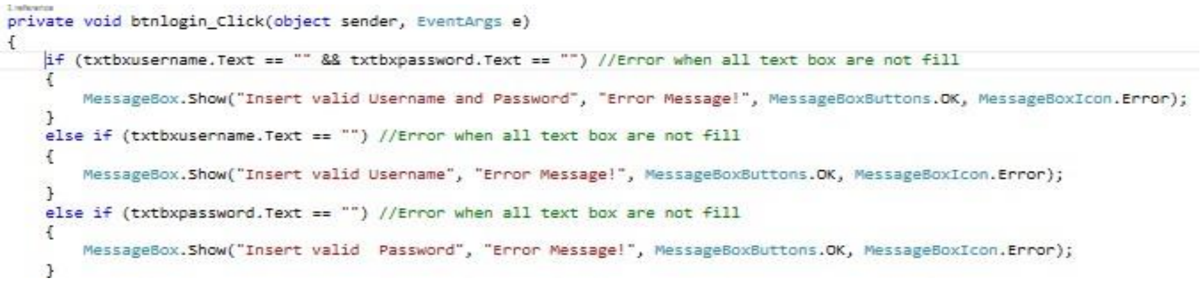
Code 6 shows that only ‘admin1’ and ‘admin2’ with correct password read from database will have successful log in otherwise MessageBox will pop up to insert again.



```
try
{
    string myConnection = @"Data Source=(LocalDB)\v11.0;AttachDbFilename=C:\Users\user\Documents\Visual Studio 2013\Projects\WindowsFormsApplication13\DB\MyDatabase.mdf;Integrated Security=True;Connect Timeout=30";
    SqlConnection myConn = new SqlConnection(myConnection);

    SqlCommand SelectCommand = new SqlCommand("select * from TableLogin where username='" + this.txtbxusername.Text + "' and password='" + this.txtbxpassword.Text + "' ;", myConn);

    SqlDataReader myReader;

    myConn.Open();
    myReader = SelectCommand.ExecuteReader(CommandBehavior.CloseConnection);

    string userRole = string.Empty;
    ;
    if (myReader.Read() == true)
    {
        if (txtbxusername.Text == "admin1")
        {
            MessageBox.Show("Success" + "\t" + txtbxusername.Text);
            this.Hide();
            AdminMenuForm ss = new AdminMenuForm();
            ss.Show();
        }
        if (txtbxusername.Text == "admin2")
        {
            MessageBox.Show("Success" + "\t" + txtbxusername.Text);
            this.Hide();
            AdminMenuForm ss = new AdminMenuForm();
            ss.Show();
        }
```

Code 6: Login Security for administrator

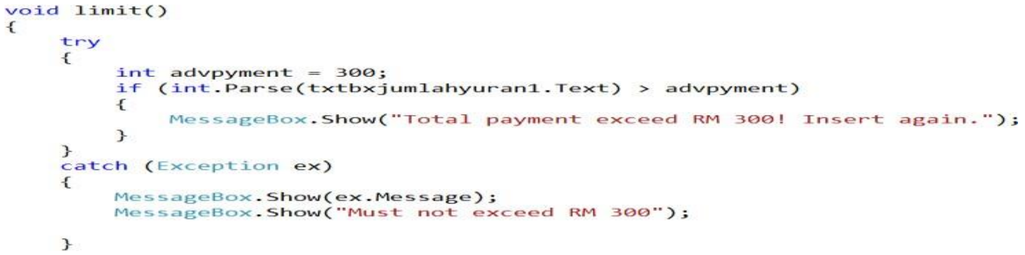
4.4 Error Handling



```
private void btnlogin_Click(object sender, EventArgs e)
{
    if (txtbxusername.Text == "" && txtbxpassword.Text == "") //Error when all text box are not fill
    {
        MessageBox.Show("Insert valid Username and Password", "Error Message!", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    else if (txtbxusername.Text == "") //Error when all text box are not fill
    {
        MessageBox.Show("Insert valid Username", "Error Message!", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
    else if (txtbxpassword.Text == "") //Error when all text box are not fill
    {
        MessageBox.Show("Insert valid  Password", "Error Message!", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
```

Code 7: Login

Code 7 is shows an administrator login. If admin input the username and password, but input data doesn’t match the database, the login page will popup message login error and specified which causes the error.



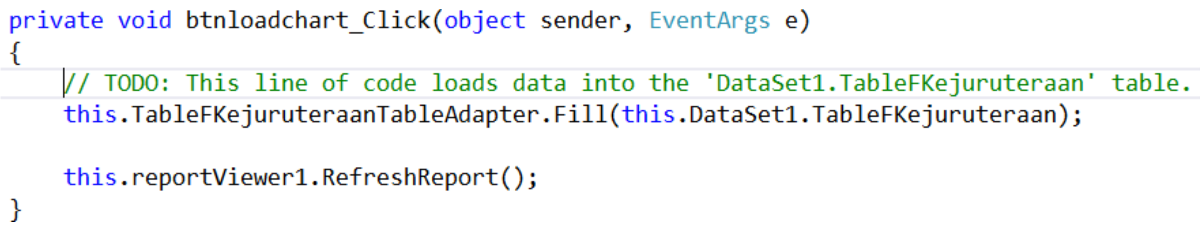
```
void limit()
{
    try
    {
        int advpyment = 300;
        if (int.Parse(txtbxjumlahyuran1.Text) > advpyment)
        {
            MessageBox.Show("Total payment exceed RM 300! Insert again.");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
        MessageBox.Show("Must not exceed RM 300");

    }
```

Code 8: Limit for inserted payment

Code 8 is shows limit for total payment made entered by customer. If customer insert more than 300 in the textbox field, the system MessageBox will pop up and ask customer to enter once again.

4.5 Report Analysis

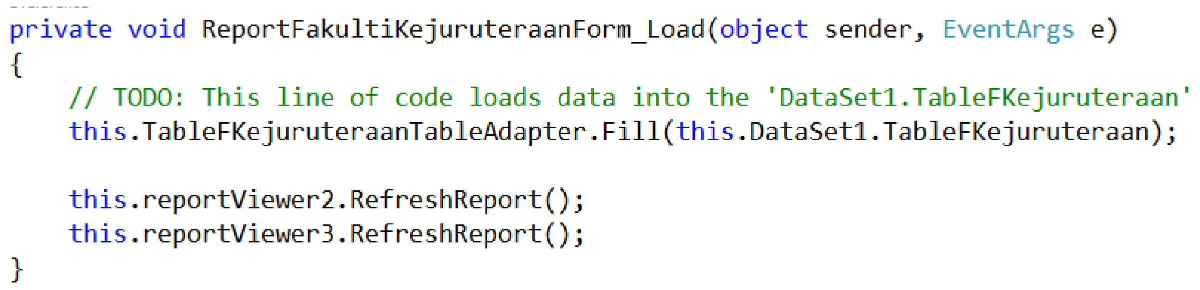
Code 9 shown below how the certain required and important details to be analysed can be display the report charts view by courses. For ex: ‘Fakulti Kejuruteraan courses’ database.



```
private void btnloadchart_Click(object sender, EventArgs e)
{
    // TODO: This line of code loads data into the 'DataSet1.TableFKejuruteraan' table.
    this.TableFKejuruteraanTableAdapter.Fill(this.DataSet1.TableFKejuruteraan);

    this.reportViewer1.RefreshReport();
}
```

Code 9: Load Report Chart View by Courses

Code 10 shown below how the certain required and important details to be analysed can be display the report charts view by gender and size of T-shirt for both participants. For ex: ‘Fakulti Kejuruteraan courses’ database.

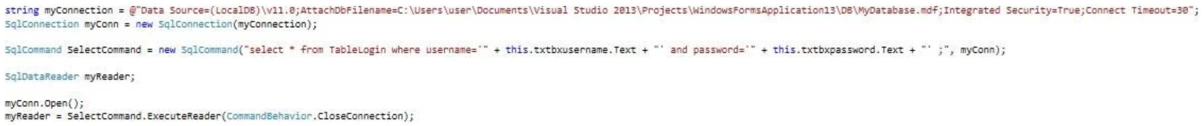


```
private void ReportFakultiKejuruteraanForm_Load(object sender, EventArgs e)
{
    // TODO: This line of code loads data into the 'DataSet1.TableFKejuruteraan'
    this.TableFKejuruteraanTableAdapter.Fill(this.DataSet1.TableFKejuruteraan);

    this.reportViewer2.RefreshReport();
    this.reportViewer3.RefreshReport();
}
```

Code 10: Load Report Chart View by Course

4.6 Selection SQL

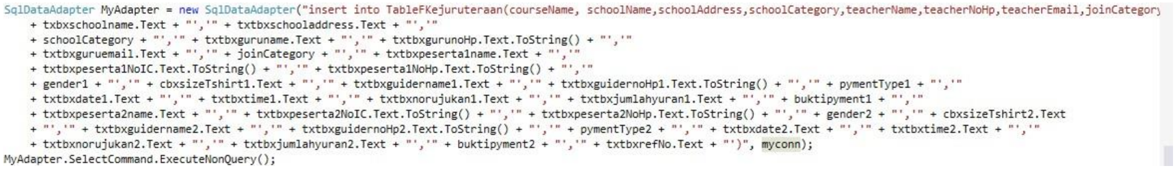
Below Code 11 shows the login code for admin



```
string myConnection = @"Data Source=(LocalDB)\v11.0;AttachDbFilename=C:\Users\user\Documents\Visual Studio 2013\Projects\WindowsFormsApplication13\DB\MyDatabase.mdf;Integrated Security=True;Connect Timeout=30";
SqlConnection myConn = new SqlConnection(myConnection);

SqlCommand SelectCommand = new SqlCommand("select * from TableLogin where username='" + this.txtbxusername.Text + "' and password='" + this.txtbxpassword.Text + "' ;", myConn);

SqlDataReader myReader;

myConn.Open();
myReader = SelectCommand.ExecuteReader(CommandBehavior.CloseConnection);
```

Code 11: Admin Login

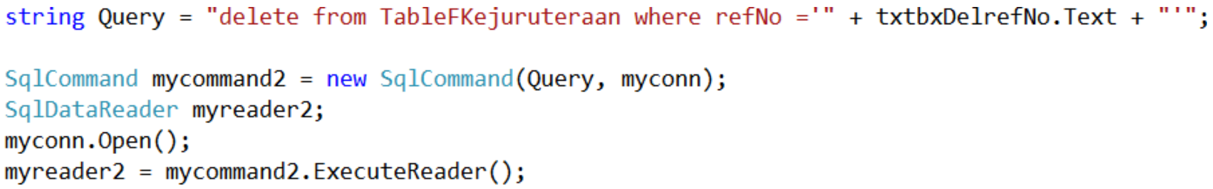
Below Code 12 shows the SQL syntax to insert data into database



```
SqlDataAdapter MyAdapter = new SqlDataAdapter("insert into TableFKejuruteraan(courseName, schoolName,schoolAddress,schoolCategory,teacherName,teacherNoHp,teacherEmail,joinCategory
    + txbxschoolname.Text + "','" + txtbxschooladdress.Text + "','"
    + schoolCategory + "','" + txtbxguruname.Text + "','" + txtbxgurunoHp.Text.ToString() + "','"
    + txtbxguruemail.Text + "','" + joinCategory + "','" + txtbxpeserta1name.Text + "','"
    + txtbxpeserta1NoIC.Text.ToString() + "','" + txtbxpeserta1NoHp.Text.ToString() + "','"
    + gender1 + "','" + cbxsizeTshirt1.Text + "','" + txtbxguidername1.Text + "','" + txtbxguidernoHp1.Text.ToString() + "','" + pymentType1 + "','"
    + txtbxdate1.Text + "','" + txtbxtime1.Text + "','" + txtbxnorujukan1.Text + "','" + txtbxjumlahyuran1.Text + "','" + buktipyment1 + "','"
    + txtbxpeserta2name.Text + "','" + txtbxpeserta2NoIC.Text.ToString() + "','" + txtbxpeserta2NoHp.Text.ToString() + "','" + gender2 + "','" + cbxsizeTshirt2.Text
    + "','" + txtbxguidername2.Text + "','" + txtbxguidernoHp2.Text.ToString() + "','" + pymentType2 + "','" + txtbxdate2.Text + "','" + txtbxtime2.Text + "','"
    + txtbxnorujukan2.Text + "','" + txtbxjumlahyuran2.Text + "','" + buktipyment2 + "','" + txtbxrefNo.Text + "')", myconn);
MyAdapter.SelectCommand.ExecuteNonQuery();
```

Code 12: Insert data

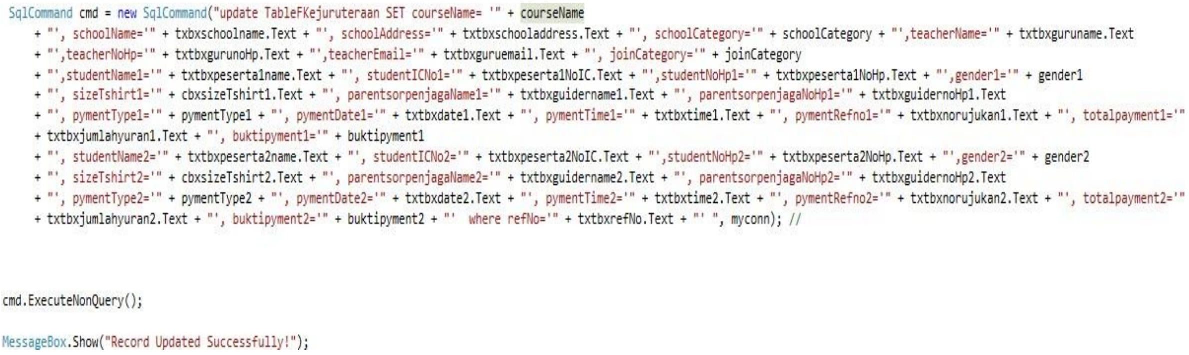
Below Code 13shows the SQL syntax to delete data from database



```
string Query = "delete from TableFKejuruteraan where refNo ='" + txtbxDelrefNo.Text + "'";

SqlCommand mycommand2 = new SqlCommand(Query, myconn);
SqlDataReader myreader2;
myconn.Open();
myreader2 = mycommand2.ExecuteReader();
```

Code 13: Delete data

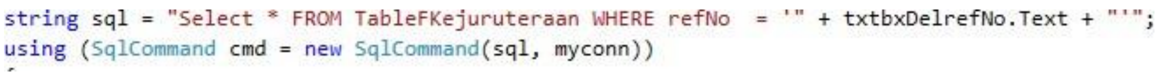
Below Code 14 shows the SQL syntax to update data from database



```
SqlCommand cmd = new SqlCommand("update TableFKejuruteraan SET courseName= '" + courseName
    + "', schoolName='" + txbxschoolname.Text + "', schoolAddress='" + txtbxschooladdress.Text + "', schoolCategory='" + schoolCategory + "',teacherName='" + txtbxguruname.Text
    + "',teacherNoHp='" + txtbxgurunoHp.Text + "',teacherEmail='" + txtbxguruemail.Text + "', joinCategory='" + joinCategory
    + "',studentName1='" + txtbxpeserta1name.Text + "', studentICNo1='" + txtbxpeserta1NoIC.Text + "',studentNoHp1='" + txtbxpeserta1NoHp.Text + "',gender1='" + gender1
    + "', sizeTshirt1='" + cbxsizeTshirt1.Text + "', parentsorpenjagaName1='" + txtbxguidername1.Text + "', parentsorpenjagaNoHp1='" + txtbxguidernoHp1.Text
    + "', pymentType1='" + pymentType1 + "', pymentDate1='" + txtbxdate1.Text + "', pymentTime1='" + txtbxtime1.Text + "', pymentRefno1='" + txtbxnorujukan1.Text + "', totalpayment1='"
    + txtbxjumlahyuran1.Text + "', buktipyment1='" + buktipyment1
    + "', studentName2='" + txtbxpeserta2name.Text + "', studentICNo2='" + txtbxpeserta2NoIC.Text + "',studentNoHp2='" + txtbxpeserta2NoHp.Text + "',gender2='" + gender2
    + "', sizeTshirt2='" + cbxsizeTshirt2.Text + "', parentsorpenjagaName2='" + txtbxguidername2.Text + "', parentsorpenjagaNoHp2='" + txtbxguidernoHp2.Text
    + "', pymentType2='" + pymentType2 + "', pymentDate2='" + txtbxdate2.Text + "', pymentTime2='" + txtbxtime2.Text + "', pymentRefno2='" + txtbxnorujukan2.Text + "', totalpayment2='"
    + txtbxjumlahyuran2.Text + "', buktipyment2='" + buktipyment2 + "'  where refNo='" + txtbxrefNo.Text + "' ", myconn); //


cmd.ExecuteNonQuery();

MessageBox.Show("Record Updated Successfully!");
```

Code 14: Update data

Below Code 15 shows the SQL syntax to update data from database

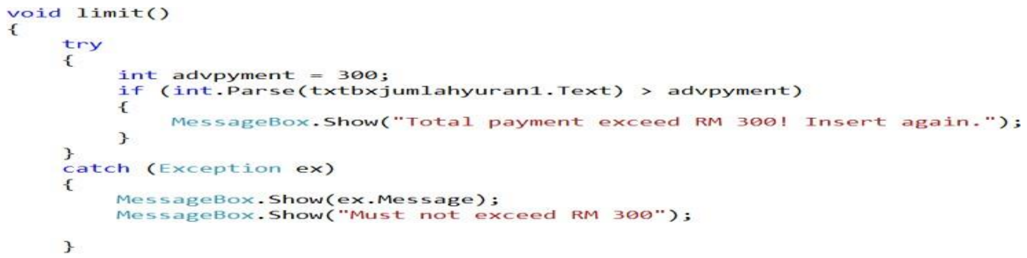


```
string sql = "Select * FROM TableFKejuruteraan WHERE refNo  = '" + txtbxDelrefNo.Text + "'";
using (SqlCommand cmd = new SqlCommand(sql, myconn))
```

Code 15: Search data by reference no

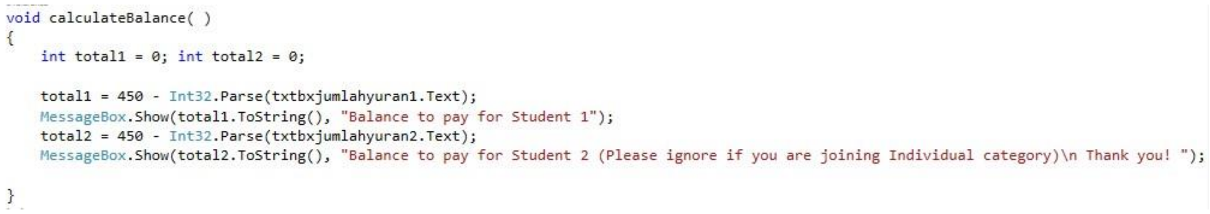
4.7 Calculation

Limit



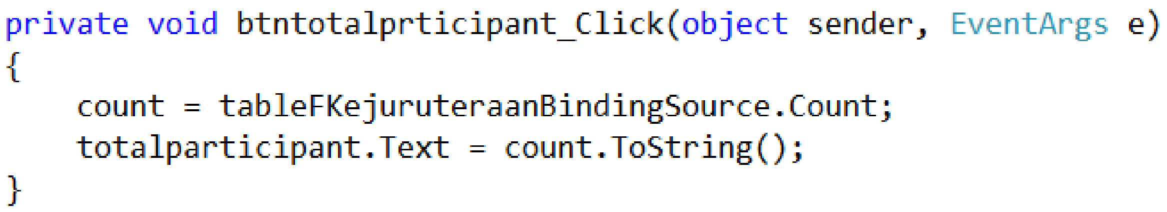
```
void limit()
{
    try
    {
        int advpyment = 300;
        if (int.Parse(txtbxjumlahyuran1.Text) > advpyment)
        {
            MessageBox.Show("Total payment exceed RM 300! Insert again.");
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show(ex.Message);
        MessageBox.Show("Must not exceed RM 300");

    }
```

Balance



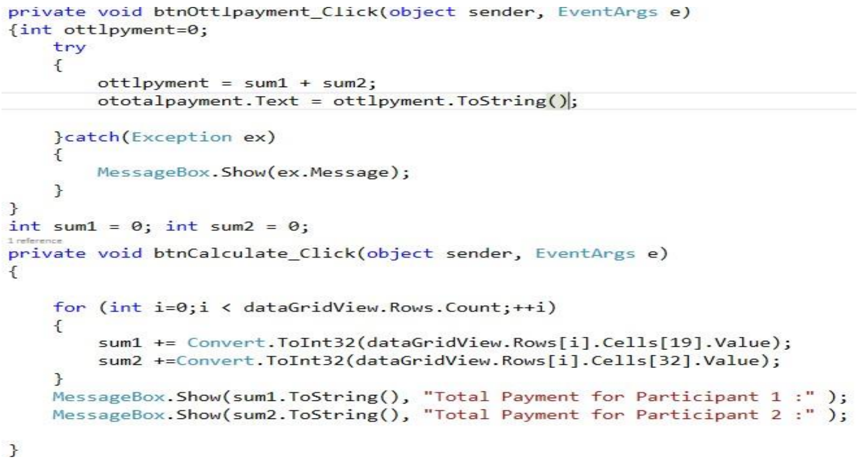
```
void calculateBalance( )
{
    int total1 = 0; int total2 = 0;

    total1 = 450 - Int32.Parse(txtbxjumlahyuran1.Text);
    MessageBox.Show(total1.ToString(), "Balance to pay for Student 1");
    total2 = 450 - Int32.Parse(txtbxjumlahyuran2.Text);
    MessageBox.Show(total2.ToString(), "Balance to pay for Student 2 (Please ignore if you are joining Individual category)\n Thank you! ");

}
```

Total set of participants



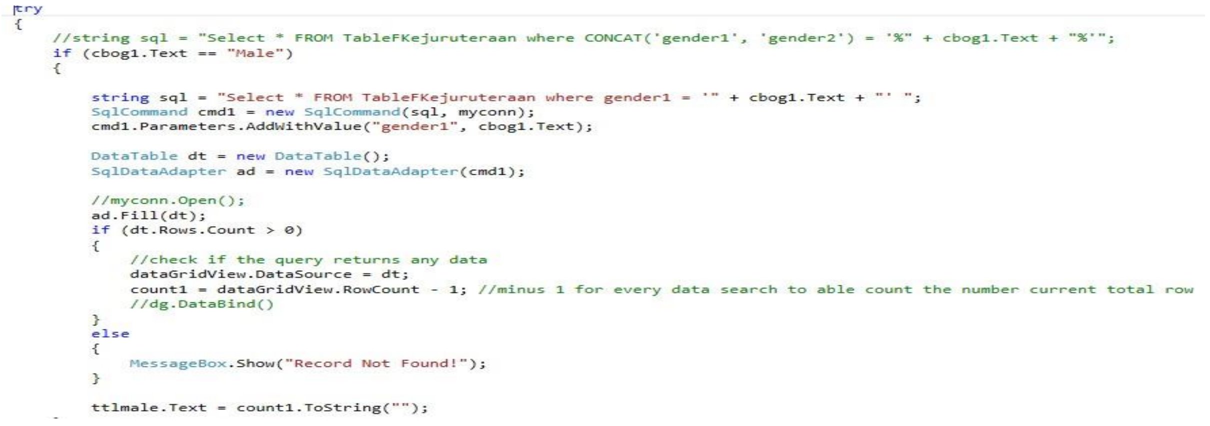
```
private void btntotalprticipant_Click(object sender, EventArgs e)
{
    count = tableFKejuruteraanBindingSource.Count;
    totalparticipant.Text = count.ToString();
}
```

Overall total payment

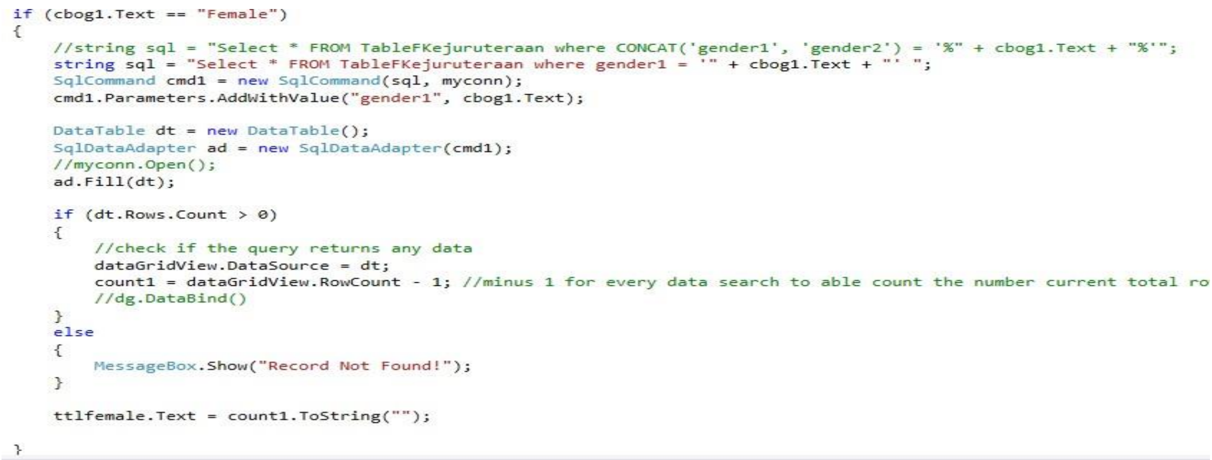


```
private void btnOttlpayment_Click(object sender, EventArgs e)
{int ottlpyment=0;
    try
    {
        ottlpyment = sum1 + sum2;
        ototalpayment.Text = ottlpyment.ToString();

    }catch(Exception ex)
    {
        MessageBox.Show(ex.Message);
    }
}
int sum1 = 0; int sum2 = 0;
1 reference
private void btnCalculate_Click(object sender, EventArgs e)
{

    for (int i=0;i < dataGridView.Rows.Count;++i)
    {
        sum1 += Convert.ToInt32(dataGridView.Rows[i].Cells[19].Value);
        sum2 +=Convert.ToInt32(dataGridView.Rows[i].Cells[32].Value);
    }
    MessageBox.Show(sum1.ToString(), "Total Payment for Participant 1 :" );
    MessageBox.Show(sum2.ToString(), "Total Payment for Participant 2 :" );

}
```

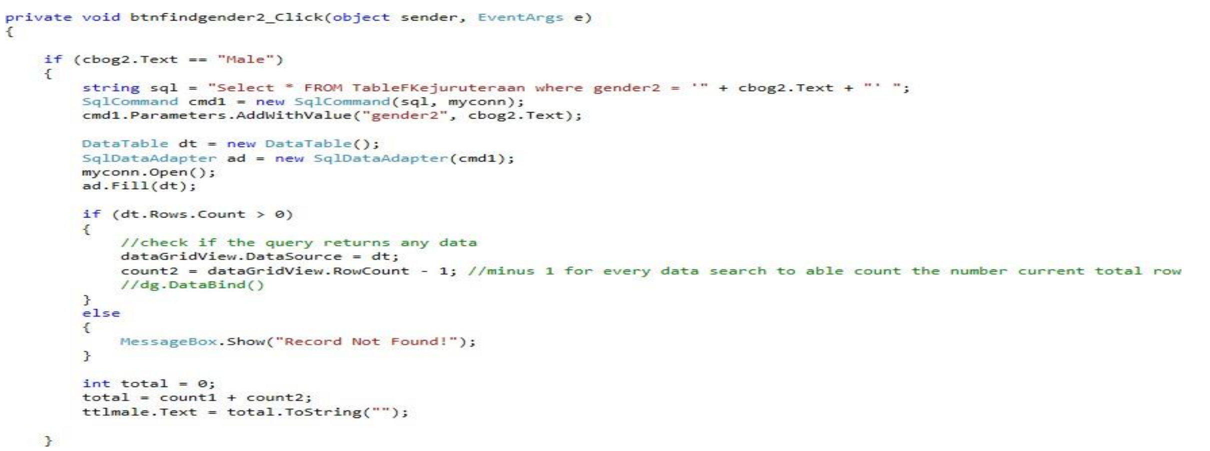
Total gender male and female



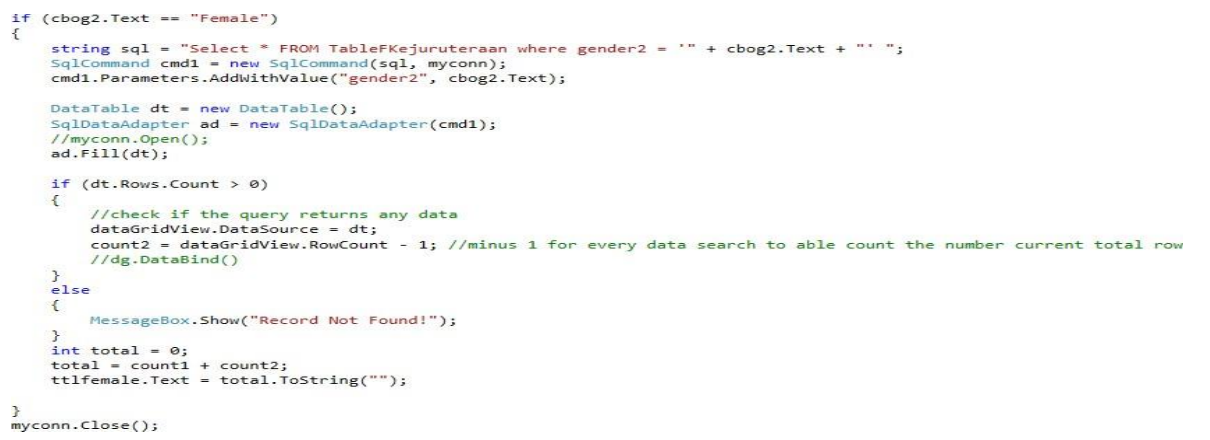
```
try
{
    //string sql = "Select * FROM TableFKejuruteraan where CONCAT('gender1', 'gender2') = '%" + cbog1.Text + "%'";
    if (cbog1.Text == "Male")
    {

        string sql = "Select * FROM TableFKejuruteraan where gender1 = '" + cbog1.Text + "' ";
        SqlCommand cmd1 = new SqlCommand(sql, myconn);
        cmd1.Parameters.AddWithValue("gender1", cbog1.Text);

        DataTable dt = new DataTable();
        SqlDataAdapter ad = new SqlDataAdapter(cmd1);

        //myconn.Open();
        ad.Fill(dt);
        if (dt.Rows.Count > 0)
        {
            //check if the query returns any data
            dataGridView.DataSource = dt;
            count1 = dataGridView.RowCount - 1; //minus 1 for every data search to able count the number current total row
            //dg.DataBind()
        }
        else
        {
            MessageBox.Show("Record Not Found!");
        }

        ttlmale.Text = count1.ToString("");
```



```
if (cbog1.Text == "Female")
{
    //string sql = "Select * FROM TableFKejuruteraan where CONCAT('gender1', 'gender2') = '%" + cbog1.Text + "%'";
    string sql = "Select * FROM TableFKejuruteraan where gender1 = '" + cbog1.Text + "' ";
    SqlCommand cmd1 = new SqlCommand(sql, myconn);
    cmd1.Parameters.AddWithValue("gender1", cbog1.Text);

    DataTable dt = new DataTable();
    SqlDataAdapter ad = new SqlDataAdapter(cmd1);
    //myconn.Open();
    ad.Fill(dt);

    if (dt.Rows.Count > 0)
    {
        //check if the query returns any data
        dataGridView.DataSource = dt;
        count1 = dataGridView.RowCount - 1; //minus 1 for every data search to able count the number current total ro
        //dg.DataBind()
    }
    else
    {
        MessageBox.Show("Record Not Found!");
    }

    ttlfemale.Text = count1.ToString("");

}
```



```
private void btnfindgender2_Click(object sender, EventArgs e)
{

    if (cbog2.Text == "Male")
    {
        string sql = "Select * FROM TableFKejuruteraan where gender2 = '" + cbog2.Text + "' ";
        SqlCommand cmd1 = new SqlCommand(sql, myconn);
        cmd1.Parameters.AddWithValue("gender2", cbog2.Text);

        DataTable dt = new DataTable();
        SqlDataAdapter ad = new SqlDataAdapter(cmd1);
        myconn.Open();
        ad.Fill(dt);

        if (dt.Rows.Count > 0)
        {
            //check if the query returns any data
            dataGridView.DataSource = dt;
            count2 = dataGridView.RowCount - 1; //minus 1 for every data search to able count the number current total row
            //dg.DataBind()
        }
        else
        {
            MessageBox.Show("Record Not Found!");
        }

        int total = 0;
        total = count1 + count2;
        ttlmale.Text = total.ToString("");

    }
```



```
if (cbog2.Text == "Female")
{
    string sql = "Select * FROM TableFKejuruteraan where gender2 = '" + cbog2.Text + "' ";
    SqlCommand cmd1 = new SqlCommand(sql, myconn);
    cmd1.Parameters.AddWithValue("gender2", cbog2.Text);

    DataTable dt = new DataTable();
    SqlDataAdapter ad = new SqlDataAdapter(cmd1);
    //myconn.Open();
    ad.Fill(dt);

    if (dt.Rows.Count > 0)
    {
        //check if the query returns any data
        dataGridView.DataSource = dt;
        count2 = dataGridView.RowCount - 1; //minus 1 for every data search to able count the number current total row
        //dg.DataBind()
    }
    else
    {
        MessageBox.Show("Record Not Found!");
    }
    int total = 0;
    total = count1 + count2;
    ttlfemale.Text = total.ToString("");

}
myconn.Close();
```

CHAPTER 5: CONCLUSION

5.1 Summary

As for conclusion, this system is about an alternative way to help UPM GISC STEM program keep their participator’s personal details in an organized and manageable. With this system, the objective of this system development can be achieved and meet the client requirements. Where the time is less consumed and saving much cost rather than using the piles of paper then disappear without backup. With this system, user can eliminate the errors that might be occurred while fill in the form. Plus this system is user friendly, where it can be handled by non IT person. Lastly, the system can ease the administration and registration staff to handle numbers of participants and they can filter the data in the best way and understandable.

5.2 Limitation

This system has some limitation where it produced complex of analysis of overall courses for each faculty such as by gender because there two participants in one rows which mean they have to count as 1 even there are two participants registered. And then there might be some confusion and error since many field need to be filled in. Lastly, this system might be need more space for storage due to many interfaces developed due to some consequences.

5.3 Future Improvement

In the future improvement, developer can make more specific form for participants 1 and participants 2 so that all the analysis can be made separate and more accurate. Then, put many error handling for each of the field and some help to ease the customer filling up the registration form. By reducing the interfaces might reduce some of the storage used.

REFERENCES

Data Dictionary (Database)

1. https://www.youtube.com/watch?v=kH0bcw9P2Lc

MODERN DESIGN Visual Studio C# | TUTORIAL Slide Menu Login UI

2. https://www.youtube.com/watch?v=kcrOAF473LM&list=PLTMPEtTibDPmz9OC_RQpLcPG9GS8dxM_m&index=13&t=790s

Show records from database to gridview using c# and SQL

3. https://www.youtube.com/watch?v=0Sc6gXF1rU0&list=PLTMPEtTibDPmz9OC_RQpLcPG9GS8dxM_m&index=22

Creating Reports in C# - Part 2 of 2

4. https://www.youtube.com/watch?v=nQvWNAldA1E&list=PLTMPEtTibDPmz9OC_RQpLcPG9GS8dxM_m&index=24

APPOINTMENT & PROGRESS LOG

NAME: AININ SOFIYA HISHAM **MATRIC NUMBER:** B031810081

SUPERVISOR: PM DR. BURHANUDDIN MOHD ABOO BAIDER. **YEAR/COURSE:** 2/1 BITI

APPOINTMENT & PROGRESS LOG
WORKSHOP I (BITU 2913) SEMESTER 2 2018/2019

Progress Activity	Start Date	End Date	Progress Outcome	Supervisor Signature & Date
1. Discussion / Verification of title and synopsis. Proposal preparation	**Week 1** 18/02/2019	**Week 2** 01/03/2019	Project Title and Synopsis	
2. Student submits proposal to Supervisor & Committee. **(Proposal)**	**Week 2** 25/02/2019	**Week 2** 01/03/2019	Submission of Students' Project Proposal • Submit hardcopy to **Supervisor** for signature. • Submit to **Committee**: Softcopy in pdf format → Upload to U-Learn	
3. Discussion with supervisor on analysis of problem. **(Progress 1)**	**Week 3** 04/03/2019	**Week 4** 15/03/2019	Details of problem description. Problem decomposition and structure chart.	
4. Discussion with supervisor on design of solution. **(Progress 2)**	**Week 5** 18/03/2019	**Week 6** 29/03/2019	Flow chart, pseudo code, data model, data dictionary and input/output design.	
5. Project Implementation **(Progress 2)**	**Week 7** 01/04/2019	**Week 8** 12/04/2019	Dh connect database. *Project Implementation, Demo, 30% of project completed*	
6. Project Implementation **(Progress 3)**	**Week 9** 22/04/2019	**Week 11** 10/05/2019	*Project Implementation, Demo, 70% of project completed*	
7. Project Implementation **(Progress 3)**	**Week 12** 13/05/2019	**Week 13** 24/05/2019	Project Implementation, Demo, 80%-100% of project completed	
8. Final Presentation & Submission of Final Report	**Week 14** 27/05/2019	**Week 14** 31/05/2019	100% of project completed, Final presentation & Final Report • Submit hardcopy to Evaluator & Supervisor. • Submit to **Committee**: CD and Report.pdf format (Upload to U-Learn)	

* Week calculation is based on Lecture Week **EXCLUDES** Mid Semester Break.