

## Course Enrolment System - Project Document

Name :- Nishadi Nuwanmini Wickramanayaka

Reg No :- 2016/CS/156

Index No :- 16001567

All the implementation details are divided into activities in the order I've followed as below.

### **Activity 01:**

Read the problem definition and understood the system requirements.

### **Activity 02:**

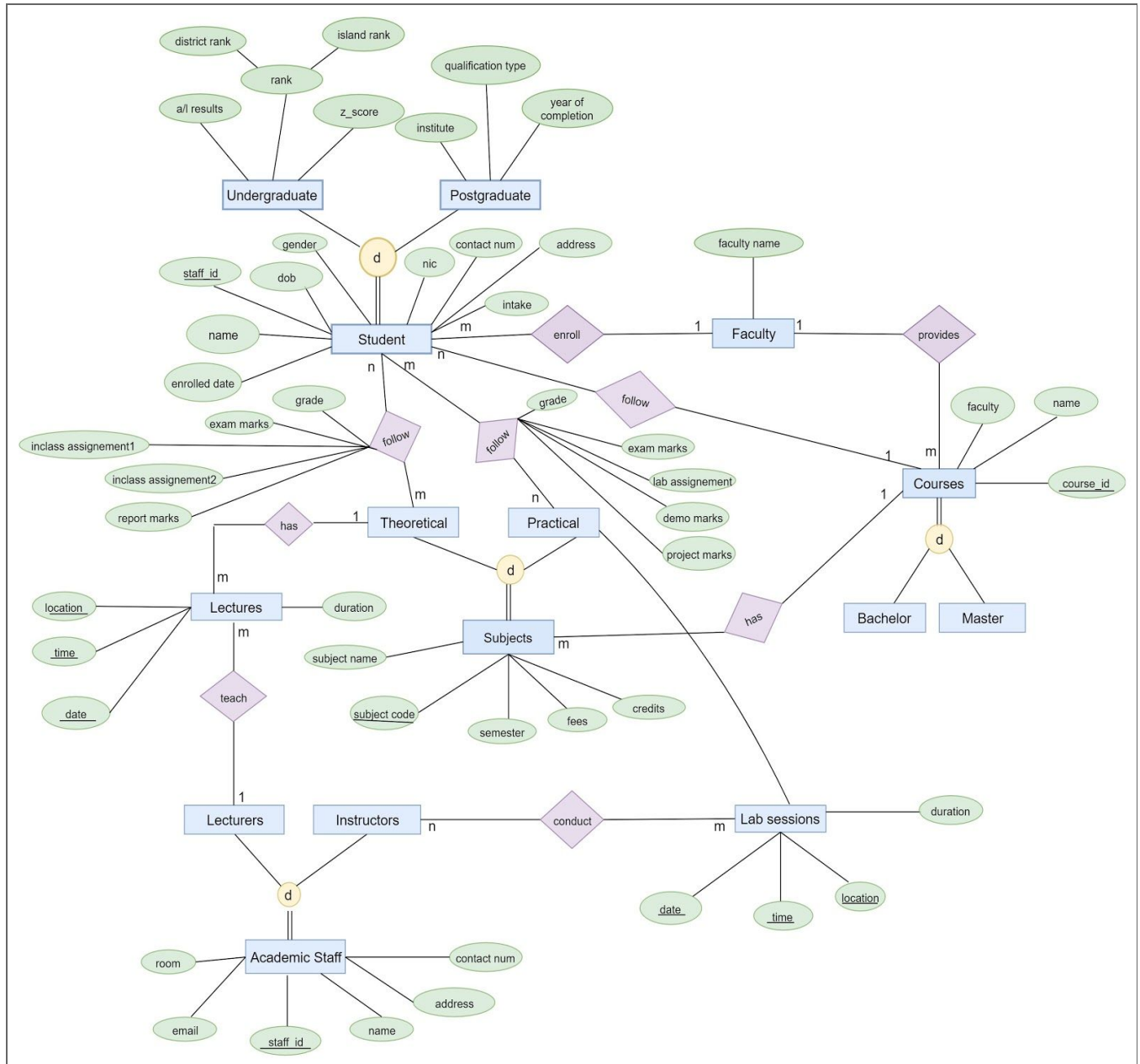
Made some assumptions in order to get a clear idea of the system and to make the system boundary.(had to assume some things since they were not given in the problem definition)

Assumptions :-

1. A student can enroll in a one course at a time.(after completing the course student can enroll in another course if he/she wants)
2. Every course,provided by the university, has enrolled students.
3. A student can not follow both bachelor and master courses at the same time.
4. Every lecturer and instructor is assigned to one or more subjects.Lecturers to conduct the lectures.Instructors to conduct the lab sessions.
5. One lecture is taken by a one lecturer and one lab is taken by two instructors.
6. A lecturer can take one or more subjects.
7. There can be subjects which don't need the lab sessions(theoretical subjects).
8. Every subject has three assignments.
9. There're no courses with only practical subjects or theoretical subjects.
10. Assigning lectures and lab sessions to relevant lecturers and instructors is upto responsible parties.It's not upto the system.After assigning them, moderators should enter those details to the system.
11. Final grade for a subject is calculated on 70% of exam marks and 30% of assignment marks.
12. Student can't obtain a final grade depending only on the assignment marks(if the student is absent for the final exam of a subject then the status of that subject for the particular student is "not completed".So he/she has to take the exam in next year)

### Activity 03:

Got a clear and complete idea of the system. Identified the attributes and entities. Finally drew the ER diagram.



(if this is not clear, pdf version is available in the project folder)

**Activity 04:**

Mapped the ER diagram as below.

1. Academic\_staff (staff\_id, name, address, con\_num, email, room, lecturer\_or\_instructor)
2. Courses (course\_id, course\_name, faculty, type)
3. Login (username, password)
4. Postgraduate (std\_id, qualification type, institute, year\_completed, course\_id)
5. Undergraduate (std\_id, district\_rank, island\_rank, z\_score, course\_id)
6. Student (std\_id, first\_name, middle\_name, last\_name, nic, dob, connum, address, gender, intake, enrolled date, course\_id)
7. Sessions (date, time, location, duration, lecture\_or\_lab, staff\_id, subject\_code)
8. Subjects (subject\_code, subject\_name, fees, credits, year\_sem, prac\_or\_theo, course\_id)
9. Std\_practical\_subjects (std\_id, subject\_code, lab\_ass\_marks, practical\_demo\_marks, project\_marks, exam\_marks, grade, semester)
10. Std\_theoretical\_subjects (std\_id, subject\_code, inclass\_ass1\_marks, inclass\_ass2\_marks, report\_marks, exam\_marks, grade, semester)
11. Staff\_subjects (staff\_id, subject\_code)

**Activity 05:**

Created the database depending on the ER diagram.

11 tables :-

1. academic\_staff
2. courses
3. login
4. postgraduate
5. undergraduate
6. sessions

7. staff\_subjects
8. std\_practical\_subjects
9. std\_theoretical\_subjects
10. students
11. subjects

The screenshot shows the phpMyAdmin web interface in a browser window. The address bar displays the URL: `localhost/phpmyadmin/db_structure.php?server=1&db=course_enrolment_system&token=ec811ad45df550bbf2c35790fda78c18`. The interface is for the 'course\_enrolment\_system' database on a MySQL server at 127.0.0.1.

On the left sidebar, the database structure is shown with a tree view. The 'course\_enrolment\_system' database is expanded, showing tables: academic\_staff, courses, login, postgraduate, sessions, staff\_subjects, std\_practical\_subjects, std\_theoretical\_subjects, student, subjects, and undergraduate.

The main panel displays a table structure view for the 'course\_enrolment\_system' database. It lists 11 tables with their respective actions (Browse, Structure, Search, Insert, Empty, Drop) and details (Rows, Type, Collation, Size, Overhead).

Table	Action	Rows	Type	Collation	Size	Overhead
academic_staff	Browse Structure Search Insert Empty Drop	22	InnoDB	latin1_swedish_ci	16 Kib	-
courses	Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 Kib	-
login	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 Kib	-
postgraduate	Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	32 Kib	-
sessions	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	48 Kib	-
staff_subjects	Browse Structure Search Insert Empty Drop	6	InnoDB	latin1_swedish_ci	32 Kib	-
std_practical_subjects	Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	32 Kib	-
std_theoretical_subjects	Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	32 Kib	-
student	Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	32 Kib	-
subjects	Browse Structure Search Insert Empty Drop	11	InnoDB	latin1_swedish_ci	32 Kib	-
undergraduate	Browse Structure Search Insert Empty Drop	3	InnoDB	latin1_swedish_ci	32 Kib	-
<b>11 tables</b>	<b>Sum</b>	<b>77</b>	<b>InnoDB</b>	<b>latin1_swedish_ci</b>	<b>320 Kib</b>	<b>0 B</b>

Below the table list, there is a 'Check all' checkbox and a 'With selected:' dropdown menu. There are also links for 'Print' and 'Data dictionary'. A 'Create table' button is visible, which opens a form with fields for 'Name:' and 'Number of columns: 4', and a 'Go' button.

The Windows taskbar at the bottom shows the system clock as 6:20 PM on 8/15/2018.

**Activity 06:**

Initiated the java project using NetBeans IDE 8.2. Then created a class called "DBconnect" to get the connection to the database. (following piece of code is used to get the connection)

```
String url = "jdbc:mysql://localhost:3306/course_enrolment_system" ;  
String username = "root" ;  
String password = "" ;  
Connection con = null;  
con = DriverManager.getConnection(url, username, password);
```

Since the DB is installed in the same computer, IP address is localhost.  
mysql service is running on port 3306.  
Url is to find the location of the Database.

**Activity 07:**

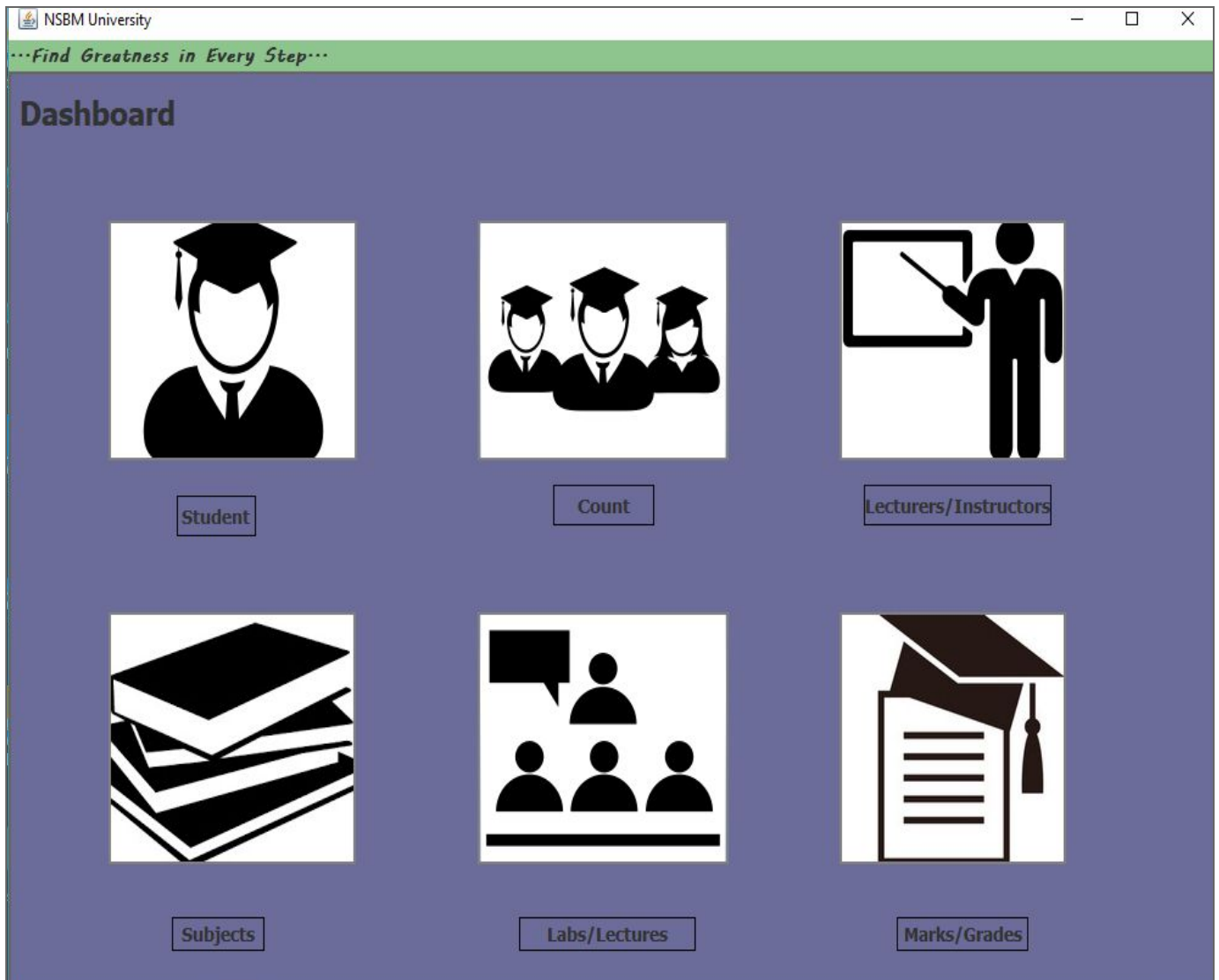
Designed the user interfaces.

- *Main Page :*



Moderator should enter his/her username and the password to login to the system.

- *Dashboard :*



6 buttons. (Student,Count,Lecturers/Instructors,Subjects,Sessions,Report)  
Each button will direct the user to the relevant operations.



- *Student Operations :*



3 buttons. Each will direct the user to the relevant operations.

- *New Student :*

NSBM University

Enter Student Details

First name :

Middle name :

Last name :

NIC :

Date Of Birth :

Address :

Contact number :

Gender :  ▼

Intake :  ▼

**Undergraduate** **Postgraduate**

Course Name :  ▼

A/L results : Maths ▼ A ▼ Chemistry ▼ A ▼ Physics ▼ A ▼

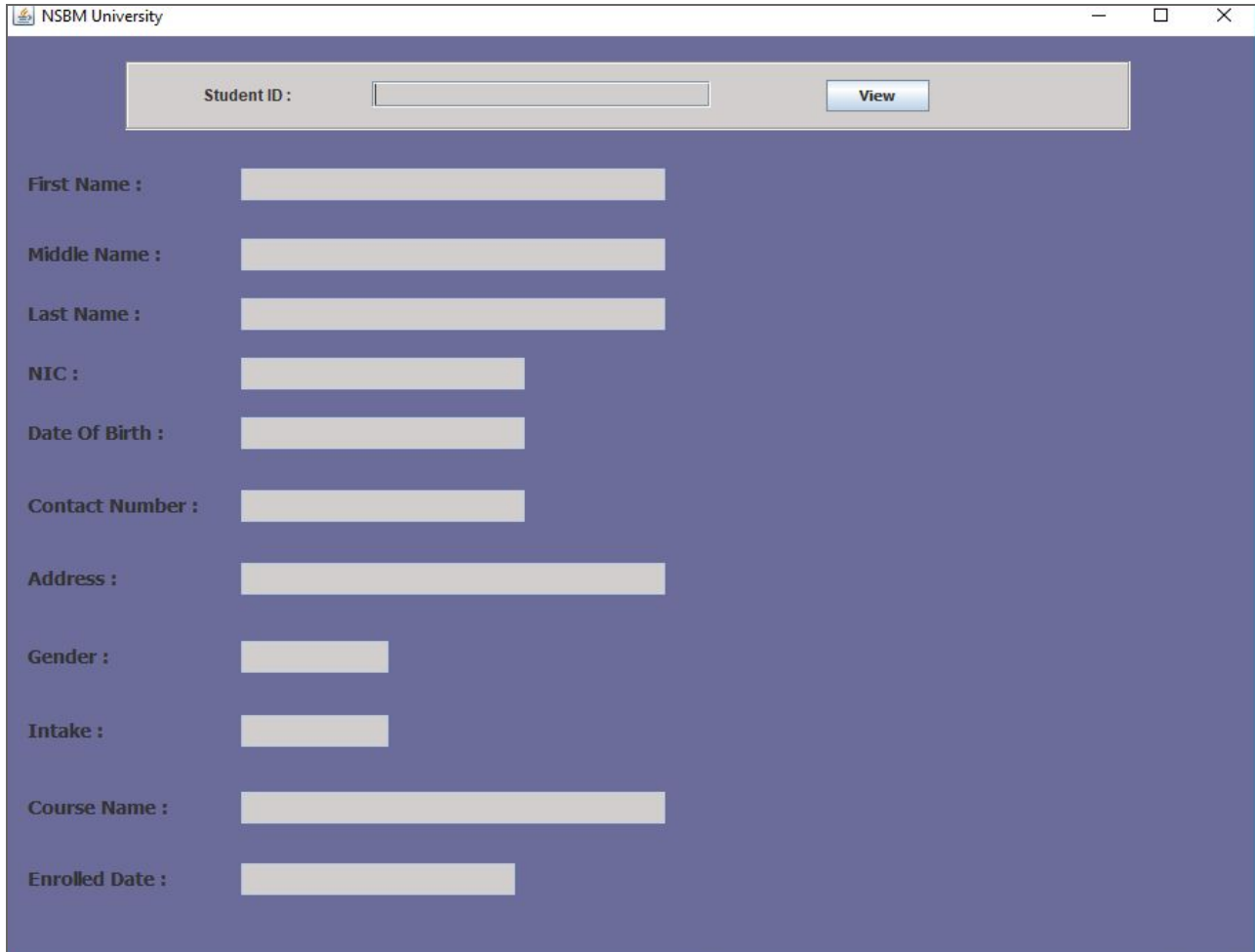
District Rank :  Island Rank :

Z Score :

**Submit**

Moderator should enter the required details of the students. Tabbed pane is used to divide specific requirements for the undergraduate and the postgraduate students.

- *View Student :*



NSBM University

Student ID :

First Name :

Middle Name :

Last Name :

NIC :

Date Of Birth :

Contact Number :

Address :

Gender :

Intake :

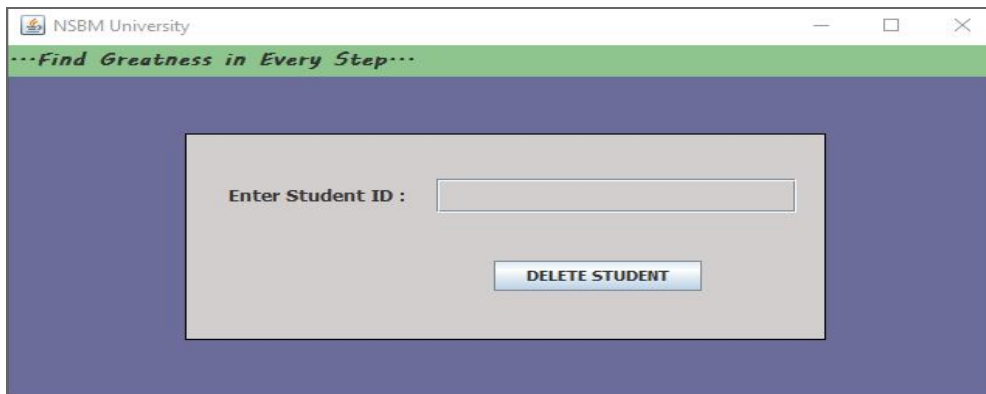
Course Name :

Enrolled Date :

System will show the above details which is related to the entered student ID.  
Text fields are uneditable since this window doesn't provide any updating function.

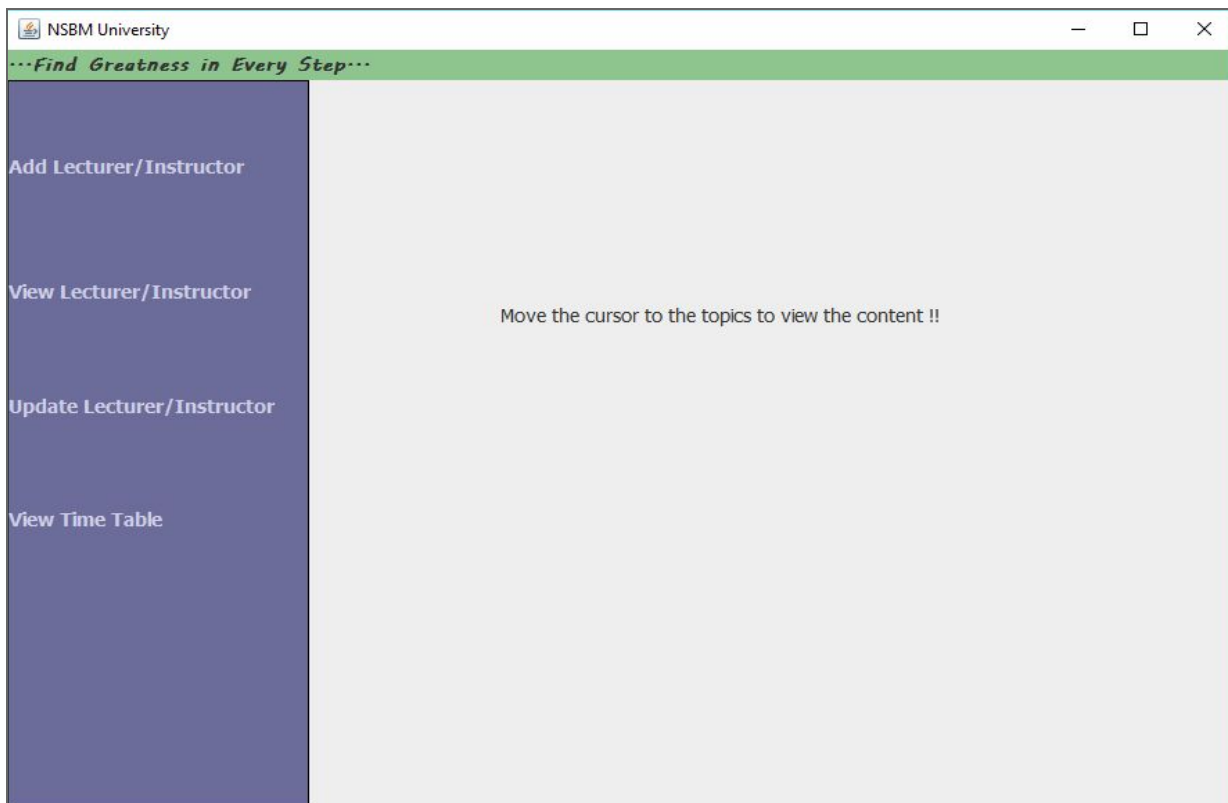


- *Delete Student :*

A screenshot of a web application window titled "NSBM University" with a green header bar containing the text "...Find Greatness in Every Step...". The main content area has a dark blue background. In the center, there is a light gray rectangular box. Inside this box, the text "Enter Student ID :" is followed by a text input field. Below the input field is a blue button with the text "DELETE STUDENT" in white capital letters.

Delete all the information related to the entered student ID.

- *Lecturers/Instructors Operations :*

A screenshot of a web application window titled "NSBM University" with a green header bar containing the text "...Find Greatness in Every Step...". The window is divided into two main sections. On the left, there is a dark blue vertical sidebar containing four white text links: "Add Lecturer/Instructor", "View Lecturer/Instructor", "Update Lecturer/Instructor", and "View Time Table". The right section has a light gray background and contains the text "Move the cursor to the topics to view the content !!".

This window is implemented using mouse motion(mouse moved) event. So to view the content cursor should be moved to the topic. Then it will show the related jpanel in the blank space.

- *Add Lecturer/Instructor :*

NSBM University

...Find Greatness in Every Step...

Add Lecturer/Instructor

View Lecturer/Instructor

Update Lecturer/Instructor

View Time Table

Enter details

Name :

Address :

Email :

Contact Number :

Room number :

Subjects :

☐ Lecturer ☐ Instructors

Enter the details of the lecturers/instructors and submit. Subjects combo box is to select the subjects that they are going to teach. Select a subject from the drop down and add it to the list.

- *View Lecturer/Instructor :*

NSBM University

...Find Greatness in Every Step...

Add Lecturer/Instructor

View Lecturer/Instructor

Update Lecturer/Instructor

View Time Table

Staff Details

Staff ID :

Name :

Address :

Email :

Contact Number :

Room :

Enter the staff id of the lecturer or instructor to view his/her details. Text fields are uneditable to ensure that viewing the details won't make any changes of the data.

- *Update Lecturer/Instructor :*

The screenshot shows a web application window titled "NSBM University" with a green header bar containing the slogan "...Find Greatness in Every Step...". On the left is a dark blue sidebar with four menu items: "Add Lecturer/Instructor", "View Lecturer/Instructor", "Update Lecturer/Instructor", and "View Time Table". The main content area is titled "Update Staff Member" and contains a form with the following fields and buttons:

- Staff ID :** A text input field followed by a "View" button.
- Name :** A text input field.
- Address :** A text input field.
- Email :** A text input field.
- Contact Number :** A text input field.
- Room :** A text input field.
- An **UPDATE** button located at the bottom right of the form area.

First you have to view the details of the particular staff member that you need to update the information. Then you can do the changes (text fields are editable) and click on the update button. It'll update the database layer too.

- *View Timetable :*

NSBM University  
...Find Greatness in Every Step...

Add Lecturer/Instructor

View Lecturer/Instructor

Update Lecturer/Instructor

View Time Table

View Time Table

Staff ID :

View

Title 1	Title 2	Title 3	Title 4

Before viewing the timetable.

NSBM University  
...Find Greatness in Every Step...

Add Lecturer/Instructor

View Lecturer/Instructor

Update Lecturer/Instructor

View Time Table

View Time Table

Staff ID :

View

Date	Time	Subject	Lab num/Hall num	Duration
2018-08-17	08:00:00	Fundamentals of C...	W002	2
2018-08-17	08:00:00	English for Professi...	W001	2

After viewing the timetable of a staff member. This table shows the details of the lab sessions / lectures of the particular lecturer or instructor.

- *Enter new subject :*

To enter a new subject to a course. There is a radio button group to determine whether it is a theoretical subject or a practical subject.

- *Select/Update subjects :*

To select the subjects or to change the selected subjects.

Subjects available shows the subjects that the student can select. Selected subjects shows the subjects that the student have selected. Maximum number of subjects that a student can select is four. Add button is to add new subject to the list and remove button is to remove a selected subject from the subject list.

- *Subject List :*

Student Index : 18000003

Semester : 1st yr\_1st sem

Subject Name	Subject Code	Credits	Fees
English for Professional Purpos...	EP002	3	2500
International Business	IB001	3	4000
ICT for Communication	IC001	2	2500
Intercultural & Interpersonal Co...	II001	2	3000
TOTAL			12000.0

PRINT

Shows the selected subjects for the semester.

Total = semester fees for the subjects.



- *Lectures and lab sessions :*

NSBM University

...Find Greatness in Every Step...

Add Lecture

Add Lab Session

Find ID

Move the cursor to the topics to view the content !!

- *Add lecture :*

NSBM University

...Find Greatness in Every Step...

Add Lecture

Add Lecture

Lecturer ID:

Subject Code:

Date: 8/15/18

Time: 23:01:41

Location:

Duration(hours):

ADD

To add a lecture. Location is the hall number.

- *Add lab sessions :*

NSBM University

...Find Greatness in Every Step...

**Add Lab Session**

Instructor ID 1:

Instructor ID 2:

Subject Code:

Date:

Time:

Location:

Duration(hours):

**ADD**

To add a lab session. Location is the lab number and the duration is set to 2 hours. (It is given in the problem definition)

- *Find ID :*

NSBM University

...Find Greatness in Every Step...

**Find ID**

**Lecturers IDs** **Instructors IDs**

**Staff IDs :**

Title 1	Title 2	Title 3	Title 4

Initial look of the table (before clicking on buttons)

Find ID

Lecturers IDs      Instructors IDs

Staff IDs :

Lecturer Name	Staff_ID
Sunil Wickramanayaka	1
Jeewani Rathnayaka	25
Manju Shri Gamage	26
Kasun Kalpana	27

Lecturers IDs will give you all the names and staff IDs of the lecturers. Instructors IDs will give you all the names and staff IDs of the instructors.

- *Marks / Grades :*

Enter Student Index Num : 18000015

Enter Subject Code : IND001

Show Subject Progress

lab assignment marks : AB

practical demonstrations : 85

Project marks : 56

Examination marks : 80

Grade C

Submit

Generate Report

'Show subject progress' shows the current state of the assignments. Grade button will calculate the grade. If examination marks field is "AB" or "not completed" grade will not be calculated.

- *Report :*

Subject Name	Subject Code	Grade
hucoiwj	BB002	B
nnnn	NNN001	B
Interior Design I	IND001	C

Final report of the semester. This will email to the student.

### Activity 08:

Identified the classes to implement.

1. Course\_enrolment\_system = initiating the program.
2. DBconnect = methods related to connect the database
3. StudentDetails = getters and setters related to the variables.
4. PostgradDetails = getters and setters related to the variables.
5. UndergradDetails = getters and setters related to the variables.
6. StudentsDBOperations = methods communicating with the database(querying).
7. CountDBOperations = methods communicating with the database(querying).

8. SubjectDetails = getters and setters related to the variables.
9. SelectedSubjects = table model to get selected subject list.
10. SubjectsDBOperations = methods communicating with the database(querying).
11. LecturerInstructorDetails = getters and setters related to the variables.
12. LecInsDBOperations = methods communicating with the database(querying).
13. SMemberTimeTable = table model to get the time table of a staff member.
14. FindIDTableModel = table model to get the staff IDs.
15. ReportDetails = getters and setters related to the variables.
16. ReportTableModel = table model to get the report of a student.
17. ReportsDBOperations = methods communicating with the database(querying).
18. SessionsDetails = getters and setters related to the variables.
19. SessionsDBOperations = methods communicating with the database(querying).

### **Activity 09:**

Implementing the system.

Important implementation details :-

- If the password or username is incorrect, system will show an error message.If not user(moderator) can login to the system successfully.

Ex:- username - nsbm001  
Password - pwnsbm001

- Course name combo box in new student form and subject combo box in add lecturer/instructor form will be loaded from the database(from the course table)
- When viewing student and deleting student, if the student id is not found in the database table system will give a message indicating that the student id is not found.

- When deleting a student, system will ask from the user again, whether to delete the particular student or not.
- Some operations are implemented using the mouse motion (mouse move) event (not mouse click event). Those operations will be initiated when the mouse is moved to the topic.
- Viewing text boxes are set as uneditable while updating text boxes are set as editable.
- Tables are loaded using separate classes (table models).
- Since a student can select maximum four subjects, if he/she selects more than 4 subjects, system will pop up a message saying that constraint.
- Find id is to find staff ids of lecturers and instructors.
- Subject grade is available only if the student take the final exam. Otherwise he/she has to take that exam in next year in order to complete the subject.

### **Activity 10:**

Entering data to the database.(ex:- course names, course IDs, subject names, subject IDs, hall numbers, lab numbers)

Entering dummy data to the database(ex:- student details, lecturer / instructor details)

Testing the system.

END..