STUDENT PERFORMANCE MONITORING SYSTEM

CSE303: DATABASE MANAGEMENT SYSTEM

GROUP 3

SHANAZ RAZIA FIDDA: 1731389
MD. TAREK AZIZ: 1730050
MD. ZAHIDUL ISLAM: 1721883
TOUFIQ AHMED NILOY: 1631281
BM FAHIM ABRAR: 1630263
SHAHNEWAZ MUHAMMAD RAJIT: 1630736
MAHFUZUR RAHMAN: 1811077

CONTENT

Chapter 1: Introduction

BACKGROUND OF THE PROJECT

Our goal is to deliver a project that will design and build to help universities to promote a productive way for student performance monitoring system. We intend to provide a wholesome experience for students, faculties, head of departments and all the higher authorities. This application is a one-stop place for students to track their progress, for faculties to track course curriculum and all the higher authorities to monitor quality of education provided. We have added features to track students CGPA trend and sleeker way of workflow. This application gives the power to generate new student accounts much faster

OBJECTIVE OF THE PROJECT

This project will be helpful for all user such as student vc dean head of department UGC spm admin and so on. This project will help student in future life and we can also learn how to implement a technology into our education

SCOPE OF THE PROJECT

- Track school-wise, department-wise and program-wise student enrolment comparison
- School-wise, department-wise and program-wise student performance trends
- Course-wise student performance trends
- Instructor-wise student performance trends
- Track PLO achievements
- Problem analysis
- Data-collection and data sharing
- Monitor project
- Team working

CHAPTER 2

REQUIREMENT ANALYSIS

- RICH PICTURE (AS-IS)
- SIX ELEMENTS (AS-IS)
- PROCESS DIAGRAM (AS-IS)
- PROBLEM ANALYSIS
- RICH PICTURE (TO-BE)
- SIX ELEMENTS (TO-BE)
- PROCESS DIAGRAM (TO-BE)

RICH PICTURE (AS-IS)

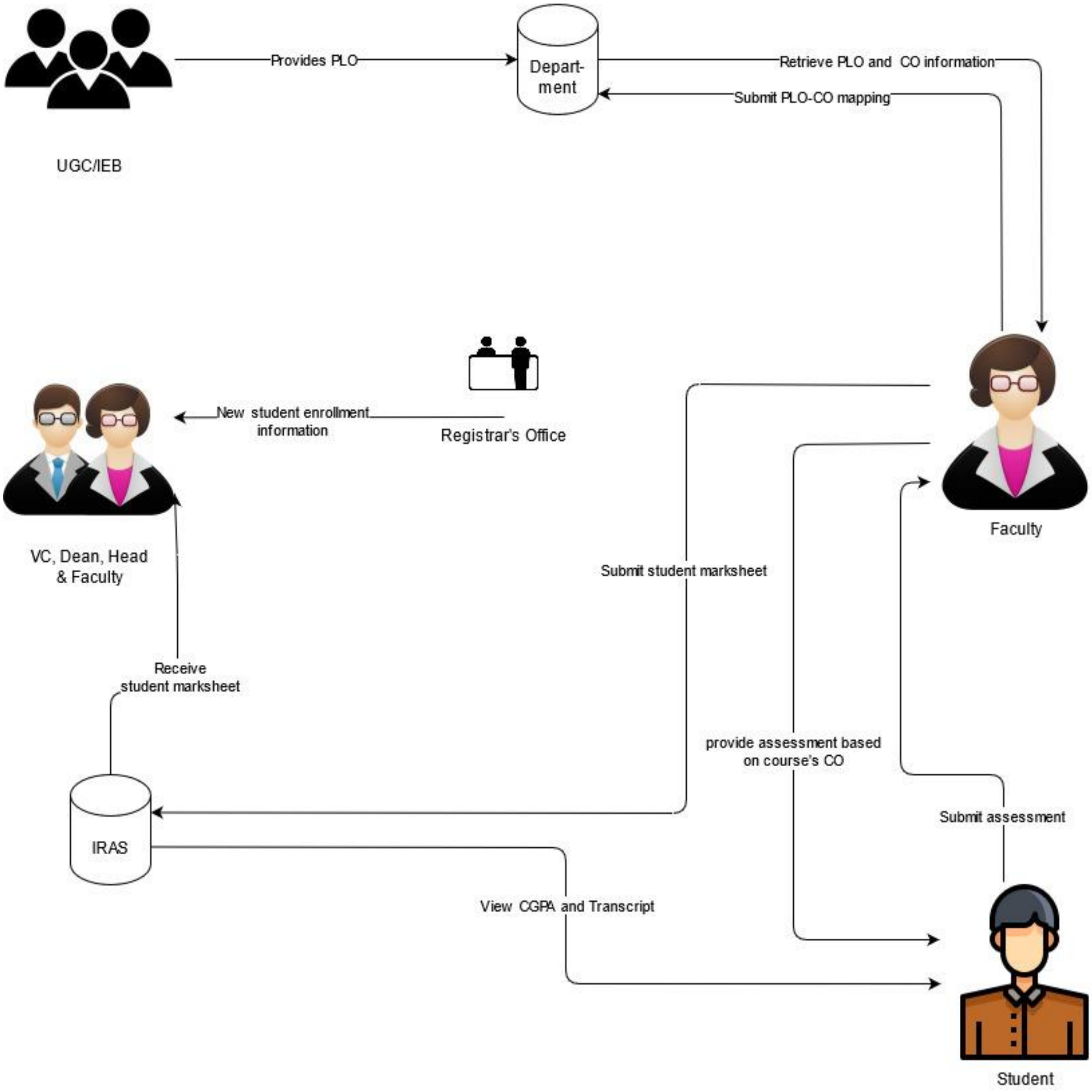


Figure: Rich Picture As-Is

SIX ELEMENTS (AS-IS)

Process	System Roles					
	Human	Non-Comp Hardware	Computing Hardware	Software	Database	Network & Communication
RECEIVE NEW STUDENT ENROLLMENT INFORMATION	**Registrar office:** 1)log in to computer 2) Receive request for enrolment information from VC, head of department, faculty, and dean. 3) Retrieve enrollment information from registrar office’s DB 4) Provide requested enrolment information to faculty, dean, head of department, VC as soft copy or printed copy. **Faculty, dean, head of department, VC:** 1) log in to computer 2) Sends request for student enrollment information	**Paper and Stationary :** Send student enrollment information as printed copy to VC, head of department, faculty, and dean. **Folders** Store student enrolment information as printed copies.	**Computer:** 1) Used by registrar office employees to send and/or receive new enrolment information to VC, head of department, faculty, and dean. 2) Used by VC, head of department, faculty, and dean to request and receive new student enrollment information. **Registrar’s Office DB server :** Used registrar’s office store student enrollment information **Printer:** To print new student enrollment information.	**Operating System:** Used in Registrar’s office, VC, head of department, faculty and dean’s computer. **Email client:** Used by Registrar’s office, VC, head of department, faculty and dean to send and receive new student information. **Office suite:** Used by VC, head of department, faculty and dean to generate enrollment trend.	**RDBMS:** Used by Registrar’s office database to store new student enrollment information. **Excel Files:** Used to store new student enrollment information in local computer.	**Internet:** Used to send or receive student enrollment information between Registrar’s office and VC, head of department, faculty, dean **Telephone:** Used for verbal communication between Registrar’s office and VC, head of department, faculty and dean.

	to registrar’s office. 3) Receive enrollment student information from registrar’s office.		**Networking Device:** Used for internet access, internal database access or communicational use.			
Receive student mark-sheet	**VC, head of department, faculty, dean :** 1) Log in to their corresponding IRAS account. 2) Search for course wise, semester wise department wise student mark-sheet. 3) Download the mark-sheet.	**Paper and Stationary :** Used to print the downloaded mark-sheet. **Folder :** Store the printed mark-sheet.	**Computer:** Used by VC, head of department, faculty, and dean to retrieve and download student mark-sheet from their IRAS account. **printer:** Used by VC, head of department, faculty, and dean to print downloaded mark-sheet. **IRAS DB server:** Used by IRAS to store student mark-sheet. **Networking device:** Used for internet access, internal	**Operating system:** Used in VC, head of department, faculty and dean’s computer. **IRAS:** To retrieve mark-sheet. **Office suite:** Used by VC, head of department, faculty and dean to generate student performance trend.	**RDBMS:** Used by IRAS to store student mark sheet. **Excel Files:** Used to store student mark-sheet local computer.	**Internet:** Used by VC, head of department, faculty, dean to retrieve student mark sheet from their corresponding IRAS account.

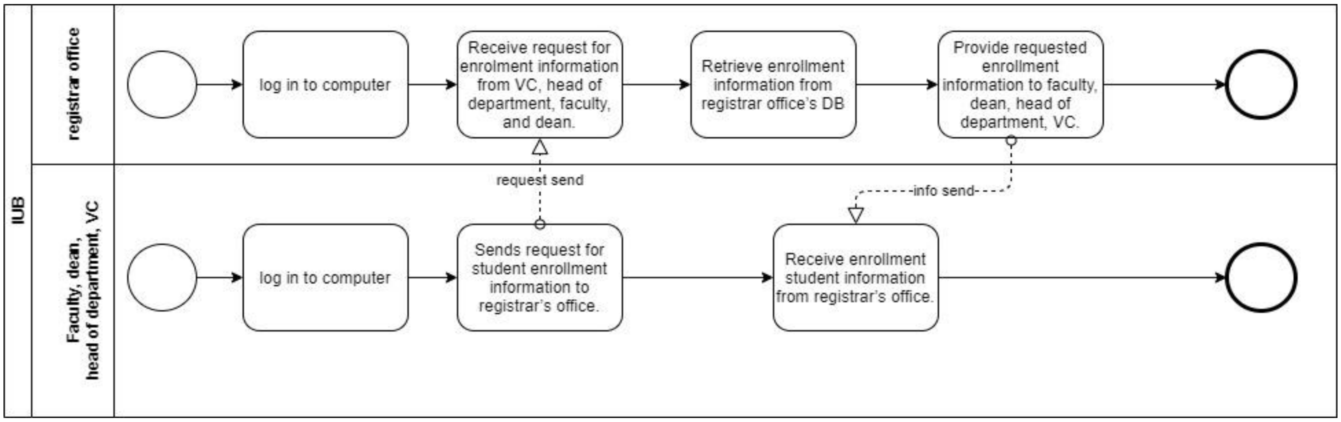
			database access			
View CGPA and Transcript	**Student:** 1) Students have to login to their IRAS account at first 2) They can view their CGPA from their IRAS dashboard 3)In order to view their transcript, they have enter year and semester from their dashboard and click on “Transcript” button to download the transcript of that particular semester	**Paper:** May be used by students to print their transcript **Folder:** To store the printed transcript	**Computer:** Used by students to visit the IRAS website and view CGPA and transcript **Printer:** Used by students to print the transcripts **IRAS DB server:** Used by IRAS to store student CGPA and transcripts **Networking device:** Used for internet access, internal database access	**Operating system:** Used in students’ computer. **IRAS:** To retrieve student CGPA and mark-sheet. **PDF Viewer:** Used to students view the downloaded transcripts	**RDBMS:** Used by IRAS to store student CGPA and transcripts	**Internet:** Used by students to login to their IRAS account and access their CGPA and transcripts.
Record student assessments and submit mark-sheet	**Faculty:** 1)Take classes, record student attendance and student class participation 2) Request department for PLO and CO	**Pen and paper:** 1) Used by department to send PLO and CO as printed form to faculty. 2) Used by teacher to make	**Computer:** 1) Used by faculty to create assignment and exam paper. 2) Used by department to store PLO	**Operating system:** Used in faculty, department and student’s computer. **Office suite:**	**RDBMS:** Used by faculty to store student mark sheets in IRAS DB.	**Internet:** 1) Used by faculty and department to receive and send PLO and CO information. 2) Used by student, faculty and department

	information of a particular course 3) Receive and download PLO and CO from department 4) Set assignment and exam paper based the CO. 5) Organize schedule and room for exam and notify students. 6) Collect exam paper and assignment from students. 7) Checks the assignment and exam paper and records marks on mark-sheet 8) Log in to IRAS and Submits mark-sheet to IRAS. **Department:** 1) Receive request for PLO and CO for a particular course from faculties	exam paper and assignment. 3) Used by student to take exam and assignment. **Folder:** Used by department student and faculty to store paper.	and CO information. 3) Used by student to view exam paper or assignment. **IRAS DB server:** Used by faculty to store student mark sheets. **Printer:** 1) Used by department to print PLO and CO information. 2) used by faculty to print assignment and exam paper **Networking device:** Used for internet access, internal database access	Used by faculty to create assignment and exam paper.		to communicate. **Telephone:** Used for verbal communication between faculty and student.

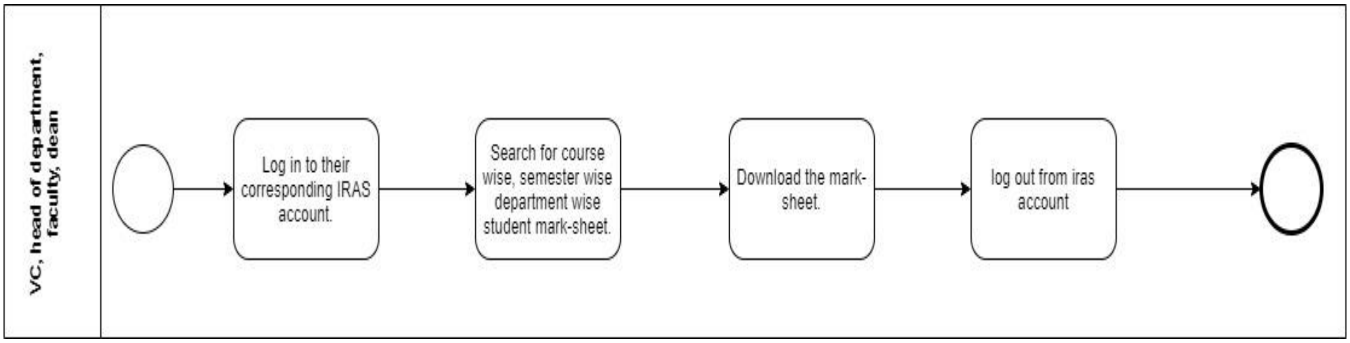
	2) Send PLO and CO to faculties **Student:** 1) Attend classes and participates in class discussion 2) Receives assignment and exam notifications from faculties 3) Attempts assignment problems and submits them to faculties 4) Takes exam on designated schedule and classroom and submits exam paper to faculties					
PLO CO mapping	**Faculty:** 1) Request department for PLO and CO information. 2)receive PLO and CO information from department 3) Download the PLO co information. 4) Discuss with otherfaculty	**Pen and paper:** Used by faculty to view PLO and CO information as printed form.	**Computer:** Used by faculty and department to receive and send PLO and CO information. **Printer:** Used by faculty to print the PLO and co information.	**Operating system:** Used in department and faculty's computer. **PDF viewer:** To see the PLO and co information.		**Internet:** Used by faculty and department to receive and send PLO and CO information and communicate with each other. **Telephone:** Used for verbal communication between

	member to create PLO and CO map. 5)sends PLO CO map to department **Department:** 1) Receive request from faculty for PLO and CO information. 2) Send PLO and CO information to faculty. 3) Receive PLO and CO mapping from faculty. 4) Store PLO co map. 5) Download the PLO co map.					faculty and department.

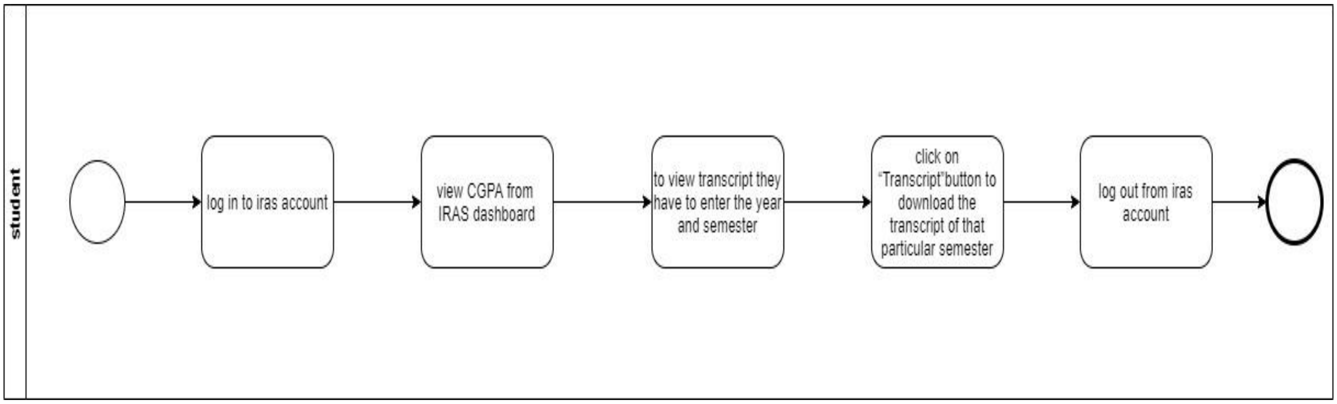
BUSINESS PROCESS DIAGRAM (AS-IS)


IUB
registrar office
log in to computer
Receive request for enrolment information from VC, head of department, faculty, and dean.
Retrieve enrollment information from registrar office's DB
Provide requested enrollment information to faculty, dean, head of department, VC.
request send
info send
Faculty, dean, head of department, VC
log in to computer
Sends request for student enrollment information to registrar's office.
Receive enrollment student information from registrar's office.

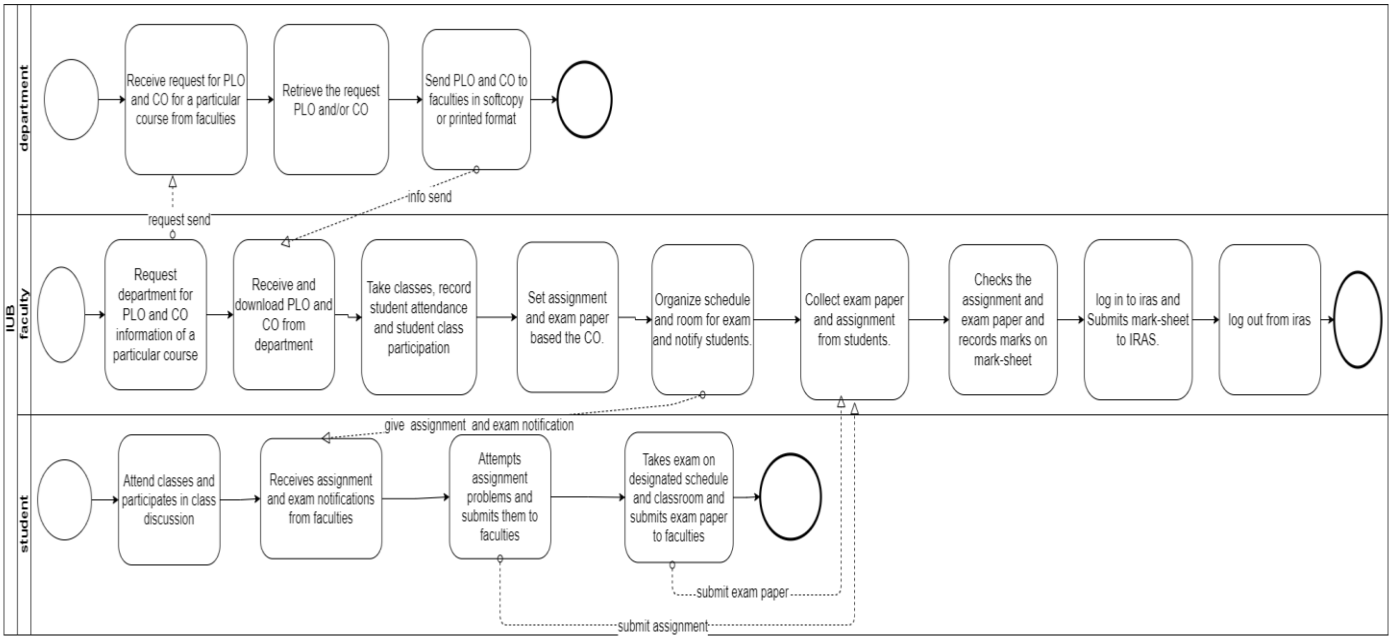
BPMN (AS-IS) FIGURE 1: View new enrollment data


VC, head of department, faculty, dean
Log in to their corresponding IRAS account.
Search for course wise, semester wise department wise student mark-sheet.
Download the mark-sheet.
log out from iras account

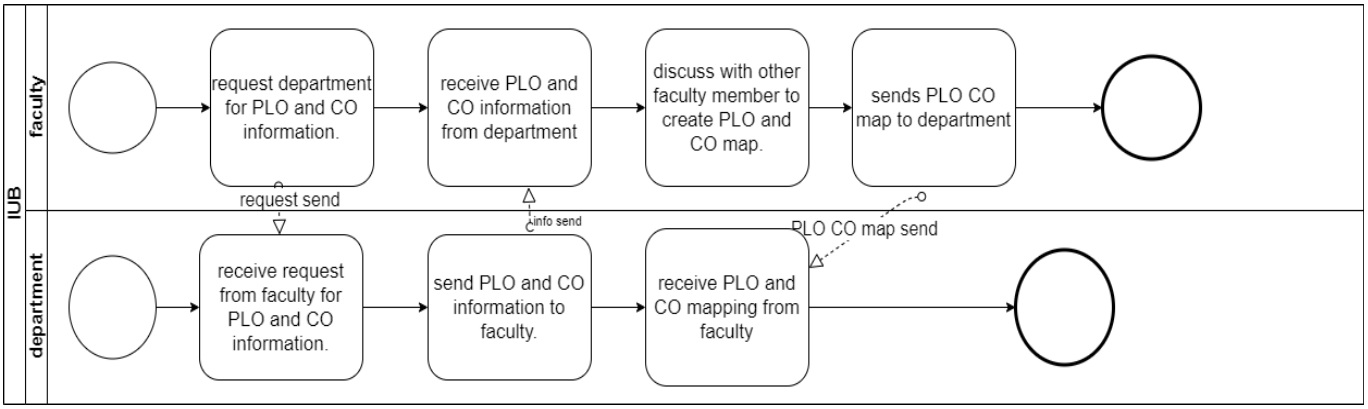
BPMN (AS-IS) FIGURE 2: View Student Mark-sheet by VC, Dean, Head, and Faculty


student
log in to iras account
view CGPA from IRAS dashboard
to view transcript they have to enter the year and semester
click on "Transcript"button to download the transcript of that particular semester
log out from iras account

BPMN (AS-IS) FIGURE 3: View Transcript by Students


IUB
department
faculty
student
Receive request for PLO and CO for a particular course from faculties
Retrieve the request PLO and/or CO
Send PLO and CO to faculties in softcopy or printed format
info send
request send
Request department for PLO and CO information of a particular course
Receive and download PLO and CO from department
Take classes, record student attendance and student class participation
Set assignment and exam paper based the CO.
Organize schedule and room for exam and notify students.
Collect exam paper and assignment from students.
Checks the assignment and exam paper and records marks on mark-sheet
log in to iras and Submits mark-sheet to IRAS.
log out from iras
give assignment and exam notification
Attend classes and participates in class discussion
Receives assignment and exam notifications from faculties
Attempts assignment problems and submits them to faculties
Takes exam on designated schedule and classroom and submits exam paper to faculties
submit exam paper
submit assignment

BPMN (AS-IS) FIGURE 4: Record Student Assessment


IUB
faculty
department
request department for PLO and CO information.
receive PLO and CO information from department
discuss with other faculty member to create PLO and CO map.
sends PLO CO map to department
request send
info send
PLO CO map send
receive request from faculty for PLO and CO information.
send PLO and CO information to faculty.
receive PLO and CO mapping from faculty

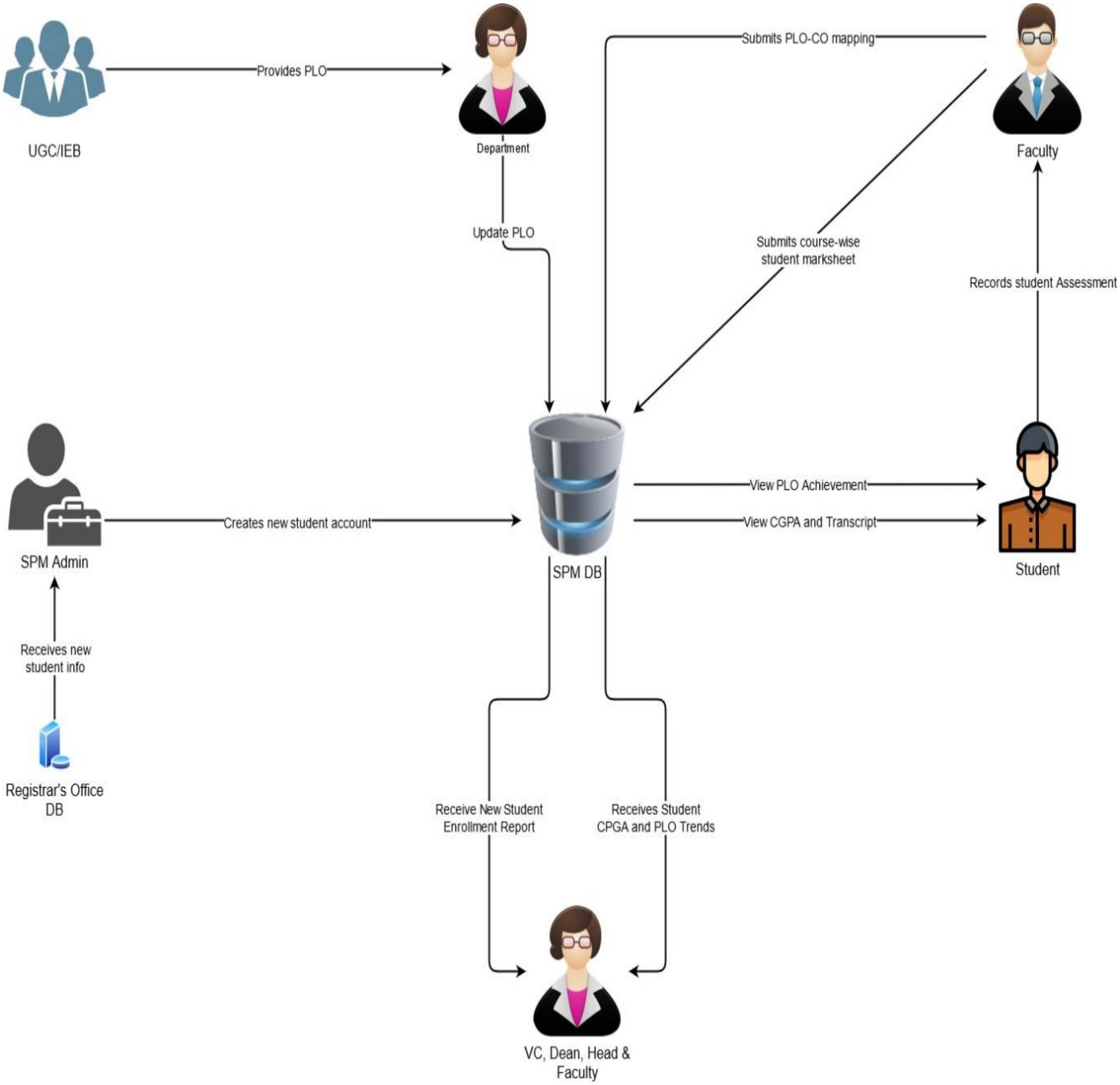
BPMN (AS-IS) FIGURE 5: Map PLO to CO

PROBLEM ANALYSIS

Process Name(s)	Stakeholders	Concerns	Analysis	Proposed Solution
“Record Student Assessment and submit marksheet”, and “Map PLO CO”	Faculty Department	Faculties have request the department to send PLO and CO details and the Department has to respond to the request	This process is time consuming as the request from the faculties has to receive by the department and retrieve necessary documents to be sent. Also, this process is resource consuming as well, as the faculty may have to send request using paper form or use any third party software for the task.	Rather than keeping the PLO and CO documents to themselves they will upload the documents to the SPM DB and faculties can easily access the files without needing to request the department
View new enrollment information	VC, Dean, Head, Faculty Registrar’s Office SPM Admin	In order to view the enrollment data, VC, Dean, Head and Faculty have send request to registrar’s office and in response the registrar’s office will send the data	This process can be time consuming and hard to manage over time, as the registrar’s office may have find manually for the request data. Also, the data sent may be just raw data and no overall trends will be shown. Users may have to use third party software to achieve that and if the data is hard copy version then generating trends will be even more difficult	Instead of requesting the registrar’s office, VC, Dean, Head and/or Faculties can view enrollment data from the enroll record on the SPM DB. Also, SPM software can show them a nice graphical analysis of enrollment
View Student CGPA Trend	VC, Dean, Head, Faculty	In order to see PLO achievement and CGPA trends of students, VC, Dean, Head and Faculty has to get the raw mark-	The mark-sheet that they download from IRAS contains raw course-wise marks for each student. It is hard	In order to solve this problem, the marksheets can stored to SPM DB and generate the desired trends

		sheet data from IRAS	the produce overall trend from the mark-sheet as they contains raw marks for specific courses. In order to generate the trends, they might require scripts to calculate the CGPA from the mark-sheet and keep track of all the entities. It becomes even complicated when they want to see trends for range of semester or for a particular department or program. Most of the users might not be skilled enough to write scripts themselves.	whenever the users need them.

RICH PICTURE (TO-BE)



UGC/IEB
Provides PLO
Department
Submits PLO-CO mapping
Faculty
Update PLO
Submits course-wise student marksheet
Records student Assessment
SPM Admin
Creates new student account
SPM DB
View PLO Achievement
View CGPA and Transcript
Student
Receives new student info
Registrar's Office DB
Receive New Student Enrollment Report
Receives Student CPGA and PLO Trends
VC, Dean, Head & Faculty

Figure: Rich Picture TO-BE

SIX ELEMENT ANALYSIS (TO-BE)

Process	System Report					
	Human	Non-computing Hardware	Computing Hardware	Software	Database	Internet & Communication
Create new students account	**SPM Admin:** 1)Receives new Students info from Registrar's office database 2) Log in to SPM DB. 3) Generate new student accounts from provided information. 4) Log out from SPM DB.	**Pen, Paper and Stationeries :** 1) May be used by SPM admin note-down any corrupted information sent from registrar's office	**Computer:** 1)Used by SPM Admin to receives data from registrar's office database & generate new Students accounts **Database Server:** 1)Registrar's office DB from which new students information will be sent 2)SPM DB in which new students account will be stored	**Operating System:** Used by SPM Admin to operate their computer **Office Suite:** May be used to store information locally in Excel format	**RDBMS:** 1) Used by both SPM DB & Registrar's office DB to stored information **Excel Software:** May be used to store information locally in Excel format	**Internet:** 1) Used to access and modify SPM database 2)Communicate between SPM Admin & Registrar's office **Telephone:** Used for verbal communicate between SPM Admin & Registrar's office

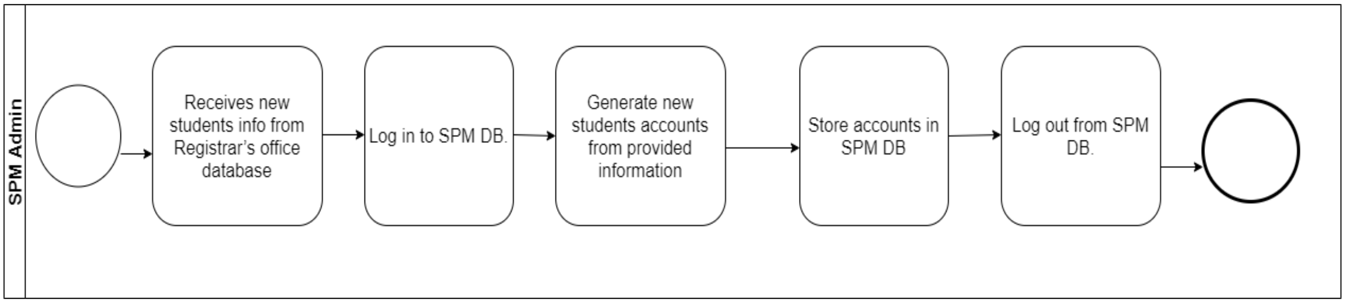
Update PLO on SPM DB	**Department:** 1) Gets PLO from UGC/IEB 2) Log in to SPM DB. 3) Stores PLO in SPM DB 4) Log out from SPM DB. **UGC/IEB:** Send PLO to department	**Pen, Paper and Stationeries:** 1) May be UGC/IEB Send printed version of PLO.	**Computer:** 1) Used by UGC/IEB to send PLO 2) Used by department to store PLO **Database Server:** Store PLO information for SPM	**Operating System:** Used by both department and UGC/IEB to operate their computer **Office Suite:** Used by UGC/IEB to create or modify PLO	**RDBMS:** Used for SPM DB to stored PLO information	**Internet:** 1) Used by UGC/IEB to provide PLO to department 2) Used by department to stored PLO in SPM DB 3) Used to communicate between department and UGC/IEB **Telephone:** Used for verbal communication between department and UGC/IEB
Record student assessment and submit mark-sheet	**Faculty:** 1)Take classes, record student attendance and student class participation 2)Log in to SPM DB. 3) Fetch PLO and CO information of a particular course from SPM 4) Set assignment and exam paper based the CO. 5) Organize schedule and room for exam and	**Pen, Paper and Stationeries:** 1) May be used by students to take lecture, write assignments , reports or take exam. 2) May be used by faculty to write lecture outline or print exam questions **Folder:** May be used by teacher to store students' assignments , exam	**Computer:** 1) May be used by student to send assignments , reports or take online exams 2) May be used by faculties to view and mark the given reports, assignments or exams **Database Server:** Used by SPM DB store student mark sheets	**Operating System:** 1) Used by both faculties and students to operate their computers **Office suite:** 1) Used by students to write assignments and reports 2) Used by faculty to write lecture outline or print exam questions	**RDBMS:** Used to store students' course wise mark sheet to SPM DB **Excel Sheet:** Used by faculties to store student mark sheets locally	**Internet:** 1) Used by students to submit the reports, assignments or take online exams 2) Used by faculty to receive student assessment 3) Used by teacher to store students course wise mark sheet to SPM DB 4) Communicate between student and teacher

	notify students. 6) Collect exam paper and assignment from students. 7) Checks the assignment and exam paper and records marks on mark-sheet 8) Log in to SPM and Submits mark-sheet. 9) Log out. **Student:** 1) Attend classes and participates in class discussion 2) Receives assignment and exam notifications from faculties 3) Attempts assignment problems and submits them to faculties 4) Takes exam on designated schedule and classroom and submits exam paper to faculties	papers or mark sheet		**Google Classroom and Google Form:** May be used for online classes and online examination		

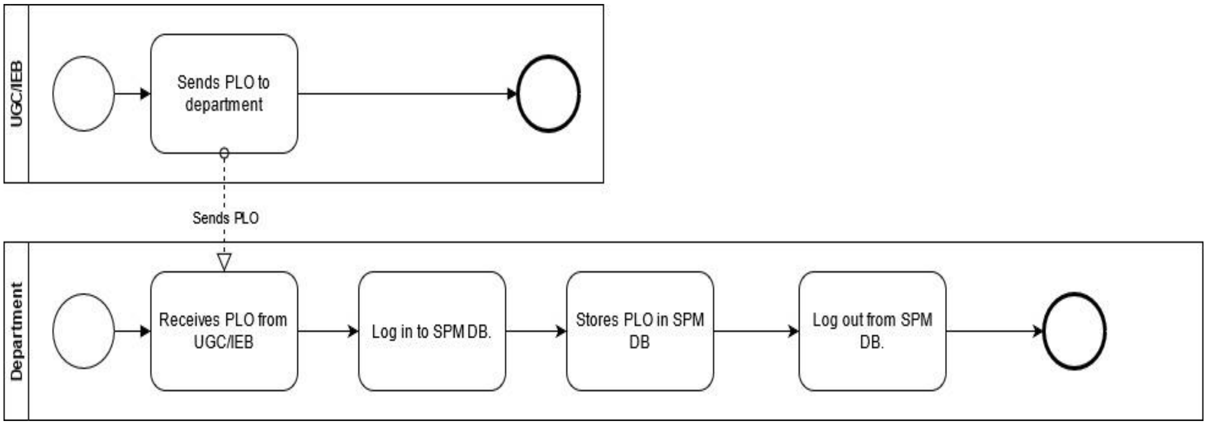
Update PLO-CO mapping to SPM DB	**Faculty:** 1) Login to SPM and retrieve PLO and CO information for a certain course from SPM 2) Discuss PLO-CO among several faculties 3) Map PLO-CO 2) Submit PLO-CO mapping in SPM DB	**Pen, Paper and Stationeries :** 1) May be used by faculty to print the PLO-CO mapping	**Computer:** 1) Used by faculty to view PLO chart and submit PLO-CO mappings **Printer:** May be used by faculty to print the PLO-CO mapping **Database Server:** Used by SPM to store PLO info and PLO-CO mappings	**Operating System:** Used by faculty to operate their computer **Printing software:** May be used by faculty to print the PLO-CO mapping	**RDBMS:** Used by SPM DB to store PLO-CO mapping	**Internet:** 1) Used by faculty to store the PLO-CO mapping 2) Used to communicate with faculties and higher authorities **Telephone:** Used for verbal communication between faculties and higher authorities
View student CGPA, transcript and PLO achievements	**Student:** 1) Login to SPM and move the “achievements”. 2) CGPA will be displayed by default 3) To view transcript, they have to enter semester and year. 4) PLO achievements will be displayed on the dashboard	**Paper:** May be used to print transcript **Folder:** May be used by store printed transcripts	**Computer:** Used to view or download, PLO achievements, CGPA and transcripts **Database Server:** Used by SPM to store student mark sheets	**Operating System:** Used by students to operate their computer **Printing software:** May be used to print transcript **PDF Viewer:** Used to view transcripts in printable format	**RDBMS:** Used by SPM DB to store student mark sheets **Excel Sheet:** May be used by students to store CGPA locally	**Internet:** Used by students to access SPM and view their CGPA and transcripts

Receive student CGPA and PLO trends	**VC, Dean of School, Head of Department and Faculties:** 1) Login to their SPM account 2) They have to enter semester range to view the CGPA and PLO achievements trends. 3) They can optionally enter particular school, department or program to view overall CGPA and PLO achievements 4) Leadership team can also view faculty-wise student CGPA and PLO achievements 5) Faculties can view overall CGPA and PLO achievements for students instructed by them.	**Pen and paper:** Used by the users to note down any particular trends in CGPA and PLO **Folder:** Used to store the papers	**Computer:** Used to view CGPA and PLO trends **Database Server:** Used by SPM to store student CGPA and PLO trends	**Operating System:** Used by the users to operate their computer	**RDBMS:** Used by SPM DB to store student CGPA and PLO trends	**Internet:** Used by the users to access SPM and view their CGPA and PLO trends

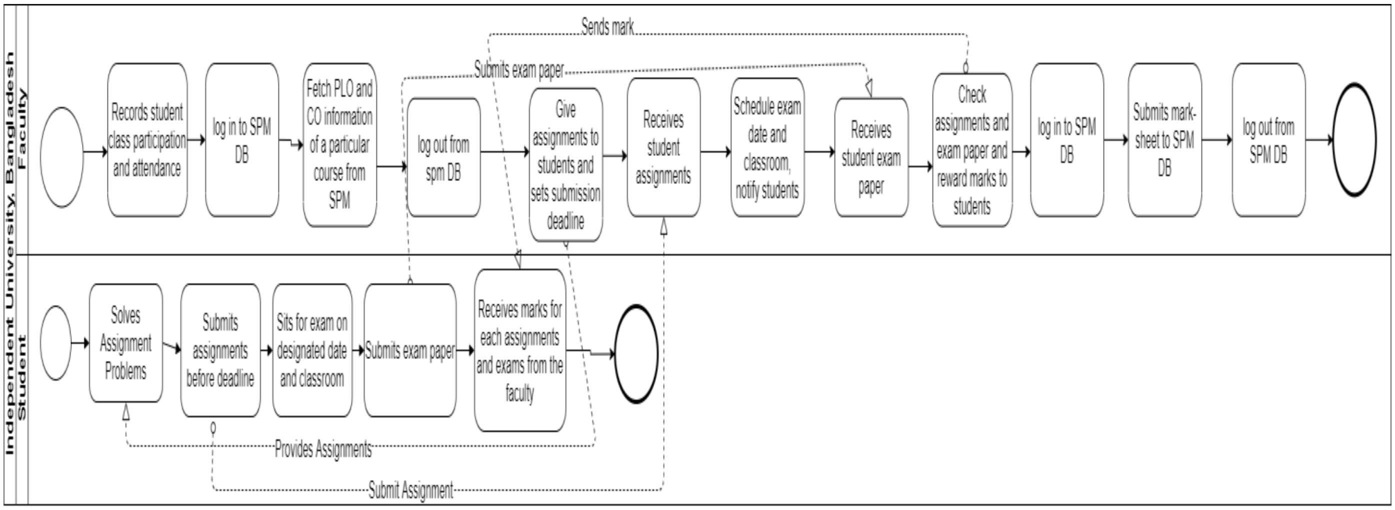
BUSINESS PROCESS DIAGRAM (TO-BE)


SPM Admin
Receives new students info from Registrar's office database
Log in to SPM DB.
Generate new students accounts from provided information
Store accounts in SPM DB
Log out from SPM DB.

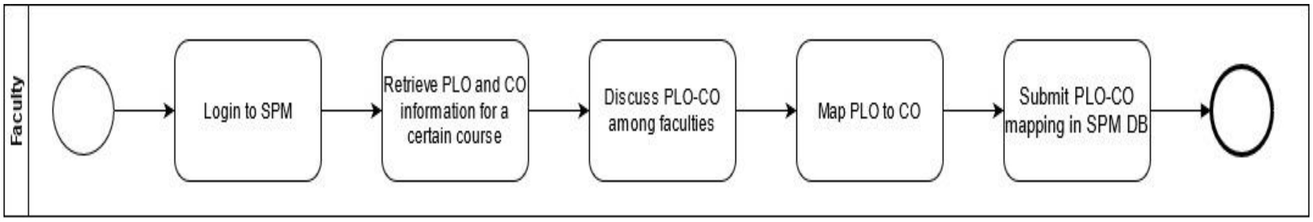
BPMN (TO-BE) FIGURE1: Create new student account



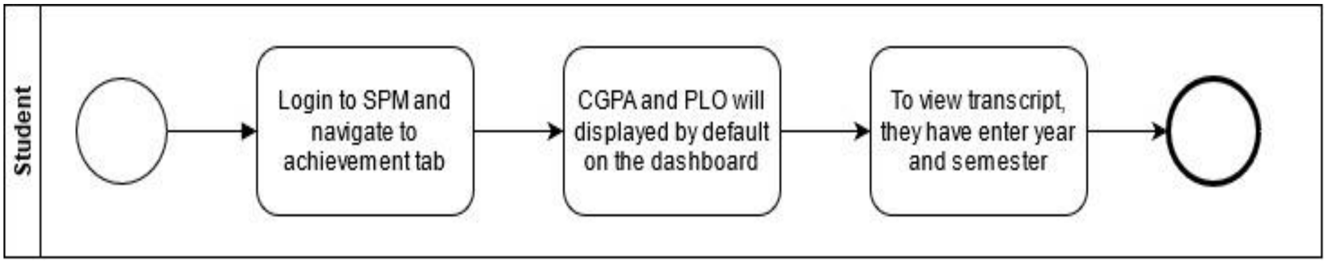
BPMN (TO-BE) FIGURE 2: Update PLO on SPM DB


Independent University, Bangladesh
Faculty
Records student class participation and attendance
log in to SPM DB
Fetch PLO and CO information of a particular course from SPM
log out from spm DB
Give assignments to students and sets submission deadline
Receives student assignments
Schedule exam date and classroom, notify students
Receives student exam paper
Check assignments and exam paper and reward marks to students
log in to SPM DB
Submits mark-sheet to SPM DB
log out from SPM DB
Sends mark
Submits exam paper
Student
Solves Assignment Problems
Submits assignments before deadline
Sits for exam on designated date and classroom
Submits exam paper
Receives marks for each assignments and exams from the faculty
Provides Assignments
Submit Assignment

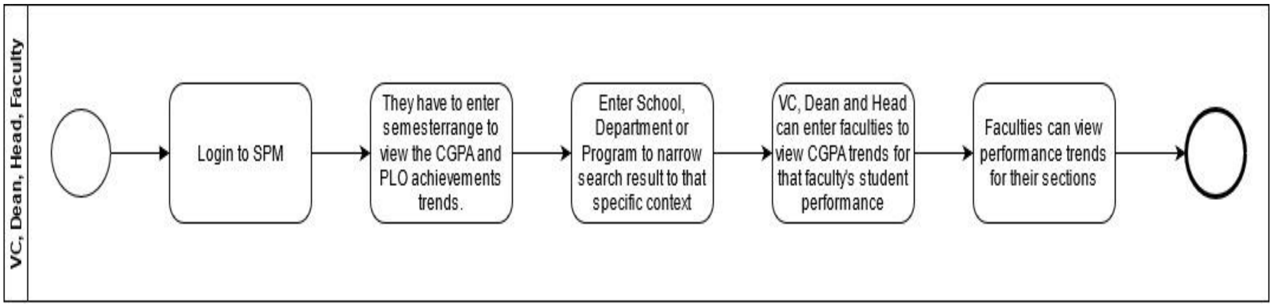
BPMN (TO-BE) FIGURE 3: Record student assessment and submit mark-sheet


Faculty
Login to SPM
Retrieve PLO and CO information for a certain course
Discuss PLO-CO among faculties
Map PLO to CO
Submit PLO-CO mapping in SPM DB

BPMN (TO-BE) FIGURE 4: Update PLO-CO mapping to SPM DB


Student
Login to SPM and navigate to achievement tab
CGPA and PLO will displayed by default on the dashboard
To view transcript, they have enter year and semester

BPMN (TO-BE) FIGURE 5: View student CGPA, transcript and PLO achievements


VC, Dean, Head, Faculty
Login to SPM
They have to enter semesterrange to view the CGPA and PLO achievements trends.
Enter School, Department or Program to narrow search result to that specific context
VC, Dean and Head can enter faculties to view CGPA trends for that faculty's student performance
Faculties can view performance trends for their sections

BPMN (TO-BE) FIGURE 6: Receive student CGPA and PLO trends

<u>CHAPTER 3</u>

LOGICAL SYSTEM DESIGN

- BUSINESS RULE
- ENTITY RELATIONSHIP DIAGRAM
- ENTITY RELATIONSHIP DIAGRAM TO RELATIONAL SCHEMA
- NORMALIZATION
- DATA DICTIONARY

Business Rule

A university must assign many employees. Each employee is assigned by exactly one university. A university must consist of many schools. Each school is belongs to exactly one university.

An employee can be faculty or VC. And a faculty can be dean of school or department head.

A university must assign exactly one VC. Each VC is assigned by exactly one university. A faculty must assign to a section. Each section must have a faculty. A department must employee many faculties. Each faculty is employed by exactly one department. a semester must consist of many faculties. Each faculty is assigned to exactly one semester. A dean of school is belongs to exactly one school. Each school must have exactly one dean of school. A department head is belongs to exactly one department. each department must have exactly one department head.

A school is consisting of many departments. each department must belong to exactly one school.

A department must enrolls many student. Each student is enrolled by exactly one department. A department is consist of many program. Each program must belong to exactly one department.

A program must enroll many students. Each student is enrolled by exactly one program. A program must provide many courses. Each course is provided by exactly one program. A program is consist of many plo. Each plo is contain by exactly one program.

A semester must contain many students. Each student enrolled to exactly one semester. A semester must contain many courses. Each course is assign to exactly one semester. A semester must consist of many sections. Each section is assign to exactly one semester.

A course is consist of many cos. Each co belongs to exactly one course. A course is assign to many sections. Each section assigned by exactly one course.

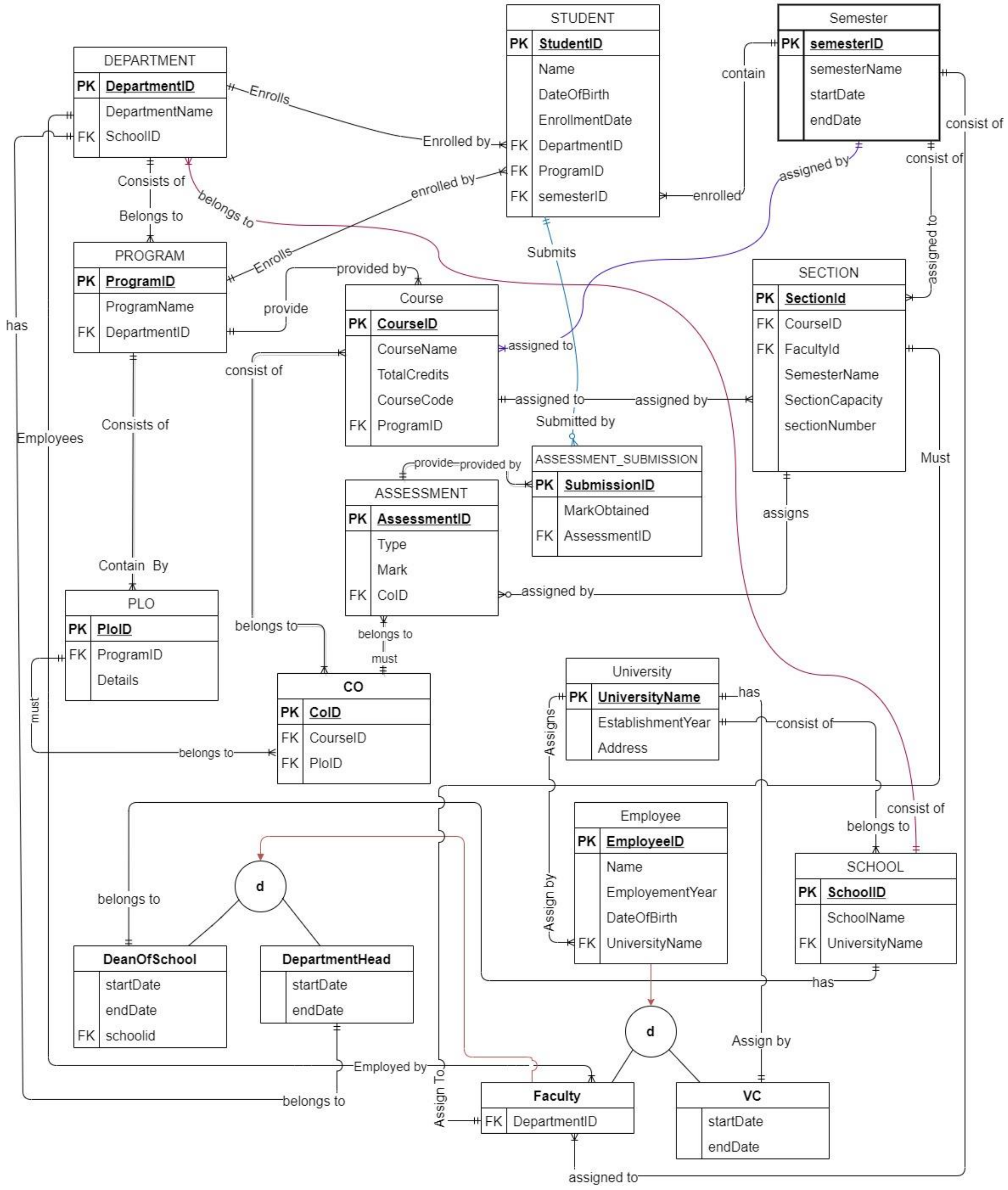
A assessment must provide many assessment submission. A assessment submission is provided by exactly one assessment. A section may assign many assessments. Each assessment is assigned by exactly one section.

A student may submit many assessment submissions. Each assessment submission is submitted by exactly one student.

A plo must have many cos. Each co is belongs to exactly one plo.

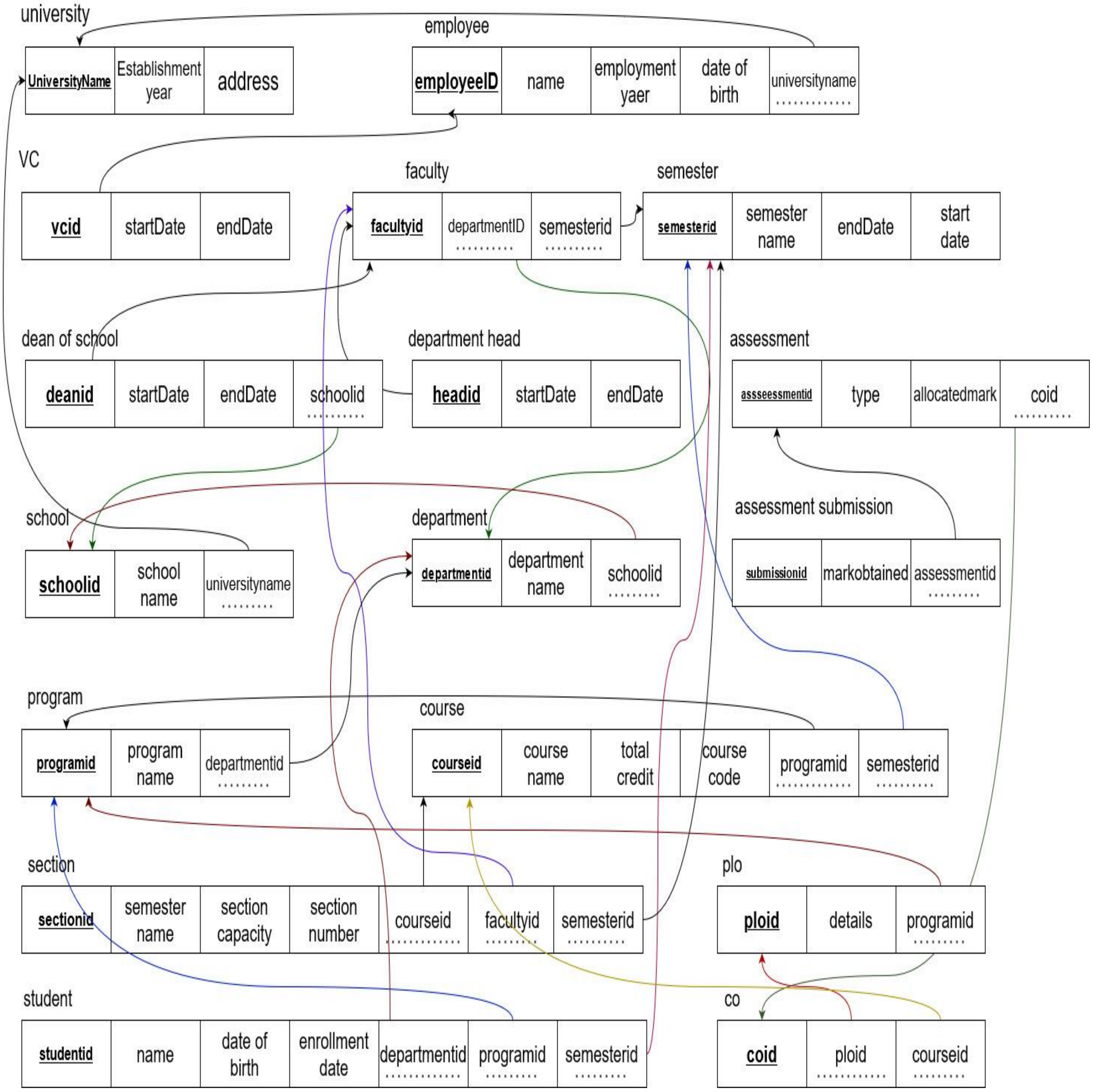
A co must have many assessments. Each assessment is belongs to exactly one co.

ERD



STUDENT
PK StudentID
Name
DateOfBirth
EnrollmentDate
FK DepartmentID
FK ProgramID
FK semesterID
Semester
PK semesterID
semesterName
startDate
endDate
DEPARTMENT
PK DepartmentID
DepartmentName
FK SchoolID
PROGRAM
PK ProgramID
ProgramName
FK DepartmentID
Course
PK CourseID
CourseName
TotalCredits
CourseCode
FK ProgramID
SECTION
PK SectionId
FK CourseID
FK FacultyId
SemesterName
SectionCapacity
sectionNumber
ASSESSMENT_SUBMISSION
PK SubmissionID
MarkObtained
FK AssessmentID
ASSESSMENT
PK AssessmentID
Type
Mark
FK CoID
PLO
PK PloID
FK ProgramID
Details
CO
PK CoID
FK CourseID
FK PloID
University
PK UniversityName
EstablishmentYear
Address
Employee
PK EmployeeID
Name
EmployementYear
DateOfBirth
FK UniversityName
SCHOOL
PK SchoolID
SchoolName
FK UniversityName
d
DeanOfSchool
startDate
endDate
FK schoolid
DepartmentHead
startDate
endDate
d
Faculty
FK DepartmentID
VC
startDate
endDate
Enrolls
Enrolled by
enrolled by
Enrolls
contain
enrolled
consist of
consist of
assigned by
assigned to
Consists of
Belongs to
belongs to
Submits
provided by
provide
assigned to
assigned to
assigned by
Submitted by
consist of
Consists of
Contain By
provide
provided by
Must
assigns
assigned by
belongs to
belongs to
must
has
Employees
must
belongs to
has
consist of
Assigns
consist of
belongs to
belongs to
Assign by
has
Assign by
Employed by
Assign To
belongs to
assigned to

RELATIONAL SCHEMA



NORMALIZATION

t1

UniversityName	Establishment year	address

t1 relation is already in 1Nf.
t1 relation is already in 2NF
t1 relation is already in 3NF
t1 relation is already in BCNF

t2

employeeID	name	employment yaer	date of birth	universityname

t2 relation is already in 1Nf.
t2 relation is already in 2NF
t2 relation is already in 3NF
t2 relation is already in BCNF

t3

programid	program name	departmentid

t3 relation is already in 1Nf.
t3 relation is already in 2NF
t3 relation is already in 3NF
t3 relation is already in BCNF

t4

departmentid	department name	schoolid

t4 relation is already in 1Nf.
t4 relation is already in 2NF
t4 relation is already in 3NF
t4 relation is already in BCNF

t5

submissionid	markobtained	assessmentid

t5 relation is already in 1Nf.
t5 relation is already in 2NF
t5 relation is already in 3NF
t5 relation is already in BCNF

t6

assseessmentid	type	allocatedmark	coid

t6 relation is already in 1Nf.
t6 relation is already in 2NF
t6 relation is already in 3NF
t6 relation is already in BCNF

t7

ploid	details	programid

t7 relation is already in 1Nf.
t7 relation is already in 2NF
t7 relation is already in 3NF
t7 relation is already in BCNF

t8

coid	ploid	courseid

t8 relation is already in 1Nf.
t8 relation is already in 2NF
t8 relation is already in 3NF
t8 relation is already in BCNF

t9

vcid	startDate	endDate

t9 relation is already in 1Nf.
t9 relation is already in 2NF
t9 relation is already in 3NF
t9 relation is already in BCNF

t10

sectionid	semester name	section capacity	section number	courseid	facultyid	semesterid

t10 relation is already in 1Nf.
t10 relation is already in 2NF
t10 relation is already in 3NF
t10 relation is already in BCNF

t11

deanid	startDate	endDate	schoolid

t11 relation is already in 1Nf.
t11 relation is already in 2NF
t11 relation is already in 3NF
t11 relation is already in BCNF

t12

courseid	course name	total credit	semesterid	programid

t12 relation is already in 1Nf.
t12 relation is already in 2NF
t12 relation is already in 3NF
t12 relation is already in BCNF

t13

studentid	name	date of birth	enrollment date	departmentid	programid	semesterid

t13 relation is already in 1Nf.
t13 relation is already in 2NF
t13 relation is already in 3NF
t13 relation is already in BCNF

t14

semesterid	semester name	endDate	start date

t14 relation is already in 1Nf.
t14 relation is already in 2NF
t14 relation is already in 3NF
t14 relation is already in BCNF

t15

headid	startDate	endDate

t15 relation is already in 1Nf.
t15 relation is already in 2NF
t15 relation is already in 3NF
t15 relation is already in BCNF

t16

schoolid	school name	universityname

t16 relation is already in 1Nf.
t16 relation is already in 2NF
t16 relation is already in 3NF
t16 relation is already in BCNF

t17

facultyid	departmentID	semesterid

t17 relation is already in 1Nf.
t17 relation is already in 2NF
t17 relation is already in 3NF
t17 relation is already in BCNF

DATA DICTONARY:

tblcourse

Name	Data type	Size	remark
courseID	Text		This is the primary key of the course. Example:cse303
ccourseName	Text		This is the name of the course. example: database management
ntotalCredit	Number		This is the credit of the course. Example:4

tbluniversity

Name	Data type	Size	Remark
cuniversityName	Text		This is the primary key and name of the university. Example: Independent University, Bangladesh
dEstablishmentYear	Datetime	yyyy	This is the year of Establish the university. Example:1993
caddress	Text		This is the address of the university. Example: Plot 16 Block B, Aftabuddin Ahmed Road Bashundhara R/A, Dhaka, Bangladesh
cVCName	Text		This is the name of the vice chancellor of the university. Example: Tanweer Hasan

tblschool

Name	Data type	Size	Remark
cschoolID	Text		This is the primary key of the school. Example:SETS
cschoolName	Text		This is the name of the school. Example: School of Engineering, Technology & Sciences
cuniversityName	Text		This is the foreign key from the university table. Example: Independent University, Bangladesh

tbldepartment

Name	Data type	Size	remark
cdepartmentID	Text		This s the primary key of the department. example: CSE
cdepartmentName	Text		This is the name of the department. example: computer science and engineering.
cschoolID	Text		This is the foreign key from the school table. Example:SETS

tblprogram

Name	Data type	Size	Remark
cprogramID	Text		This is the primary key of the program.
cprogramName	Text		This is the name of the program.
cdepartmentID	Text		This is the foreign key from the department table. Example:CSE

tblstudent

Name	Data type	Size	Remark
cstudentID	Text	7	This is the primary key of the student. Example: 1234567
cstudentName	Text		This is the name of the student. Example: MR.Abdul Korim
cemailID	Text		This is the email of student. Example: abdul@gmail.com
ncontractNo	Number	11	This is the contract number of the student. Example:01911111111
caddress	Text		This is the address of the student. Example: sector6, house 6, road 6 uttara,Dhaka
cgender	Text		This is the gender of the student. Example:male
ddateOfBirth	Date time	dd/mm/yy	This is the birth date of the student. Example:06/06/96
denrollmentYear	Date time	yyyy	This is the enrollment year when the student got admitted in the university. Example:2016
cdepartmentID	Text		This is the foreign key from department table. Example:CSE
cprogramID	Text		This is the foreign key from the program table.

tblfaculty

Name	Data type	Size	Remark
cfacultyID	Text	5	This is the primary key of the faculty. Example:11111
cname	Text		This is the name of the faculty. Example: MR.gousul azom
cemailId	Text		This is the email address of the faculty. Example:gousul@gmail.com
ncontractNo	Number	11	This is the contract number of the faculty. Example:7654321
caddress	Text		This is the address of the faculty. example:sector5,read5, house 5 uttara, dhaka
cgender	Text		This is the gender of the faculty. Example:male
ddateOfBirth	Date time	dd/mm/yy	This is birth time of the faculty. Example: 11/12/66
demploymentYear	Date time	Yyyy	This is the employment year when the faculty joined as employee in the university. Example:2009
cdepartmentID	Text		This is the foreign key from the department table. Example: CSE

tblsection

NAME	DATA TYPE	SIZE	REMARKS
csectionID	Text		This is auto-increment primary key.
ccousrseID	Text		This is the foreign key from the course table. Example: cse301
cfacultyID	Text		This is the foreign key from the faculty table. Exmaple:11554
csemesterName	Text		This is the name of the semester. Example: Spring 2016.
nsectioncapacity	Number		This is the total capacity of a section. Example: 30.

tblplo

NAME	DATA TYPE	SIZE	REMARK
cploID	Text		This is the primary key of the PLO (Program Learning Outcome). Example:”PLO1”
cprogramID	text		This is the foreign key from the program table. Example: M.Sc.
cdetails	Text		This is the details of the PLO.

tblco

NAME	DATA TYPE	SIZE	REMARKS
ccoID	Text		This is the primary key of the CO. Example “CO1”
ccourseID	Text		This is the foreign key from the course table. Example:”CSE303”
cploID	Text		This is the foreign key from the PLO table. Example: “PLO2”

tblassessmentID

NAME	DATA TYPE	SIZE	REMARKS
cassesmentID	Text		This is auto-increment primary key.
ctype	Text		This is the type of the assessment. Example:assignment
ntotalMarks	Number		This is the mark for the assessment. Example:25
ccoID	Text		This is foreign key from the CO table.example:CO1

tblassessment submission

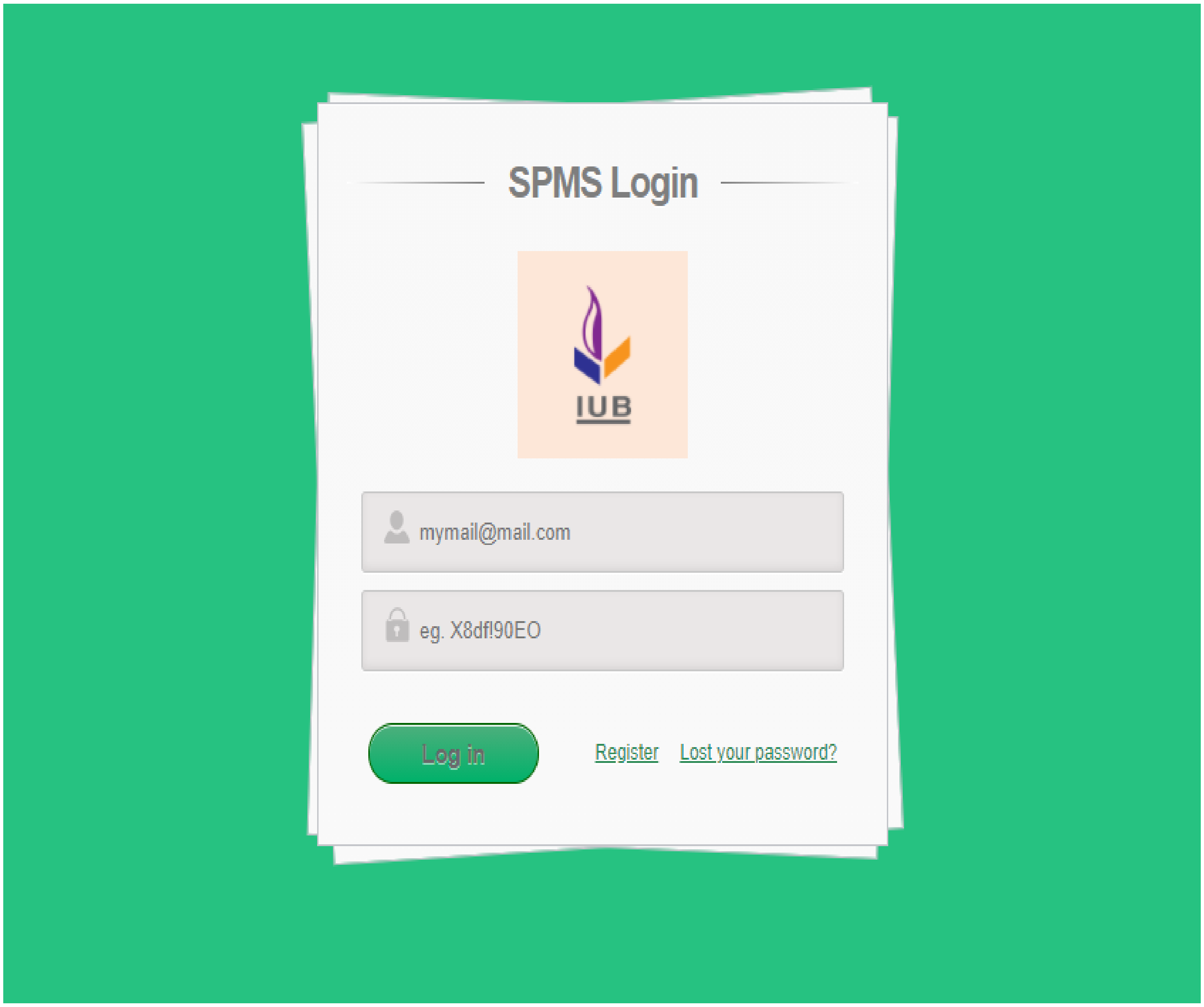
NAME	DATA TYPE	SIZE	REMARKS
csubmissionID	Text		This is auto-increment primary key.
nmarkObtained	Number		this is the mark obtained by a student. Example:7
cassessmentID	Text		This is foreign key of the assessment ID.

CHAPTER 4

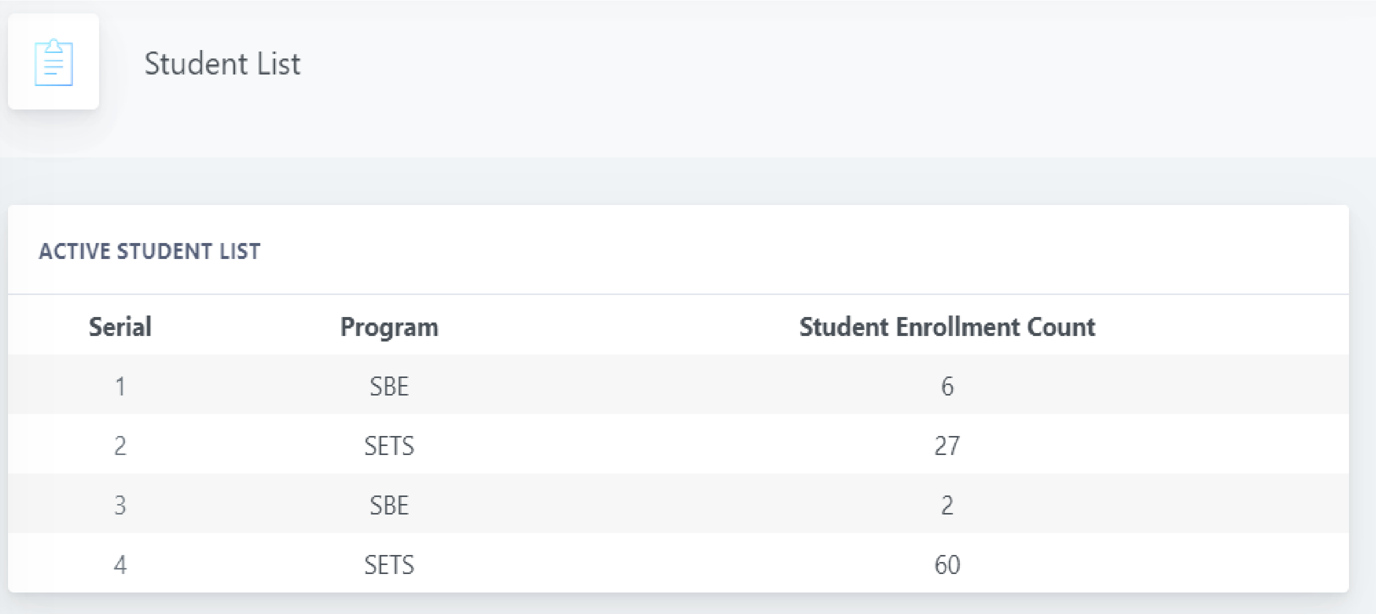
PHYSICAL SYSTEM DESIGN

- INPUT FORMS
- OUTPUT FORMS

Input Forms



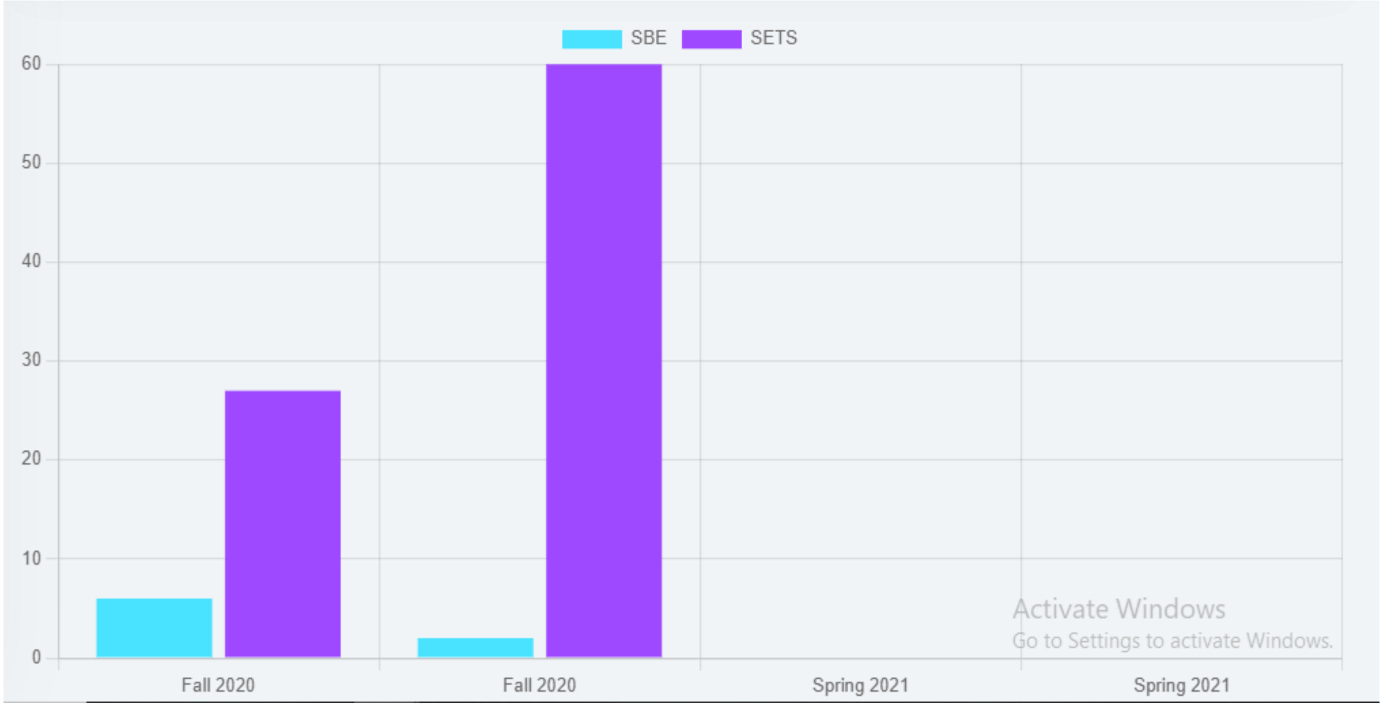
Output Form



Student List

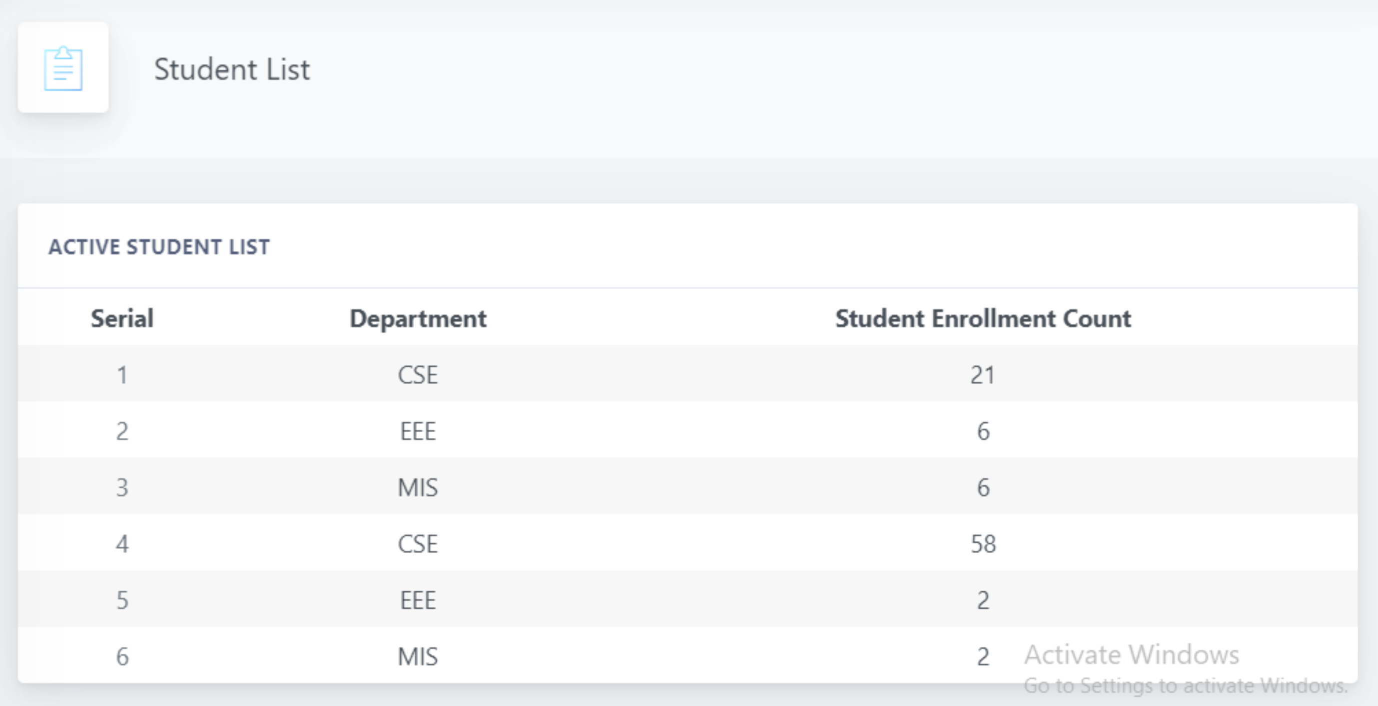
ACTIVE STUDENT LIST

Serial	Program	Student Enrollment Count
1	SBE	6
2	SETS	27
3	SBE	2
4	SETS	60



School-wise Student Enrollment:

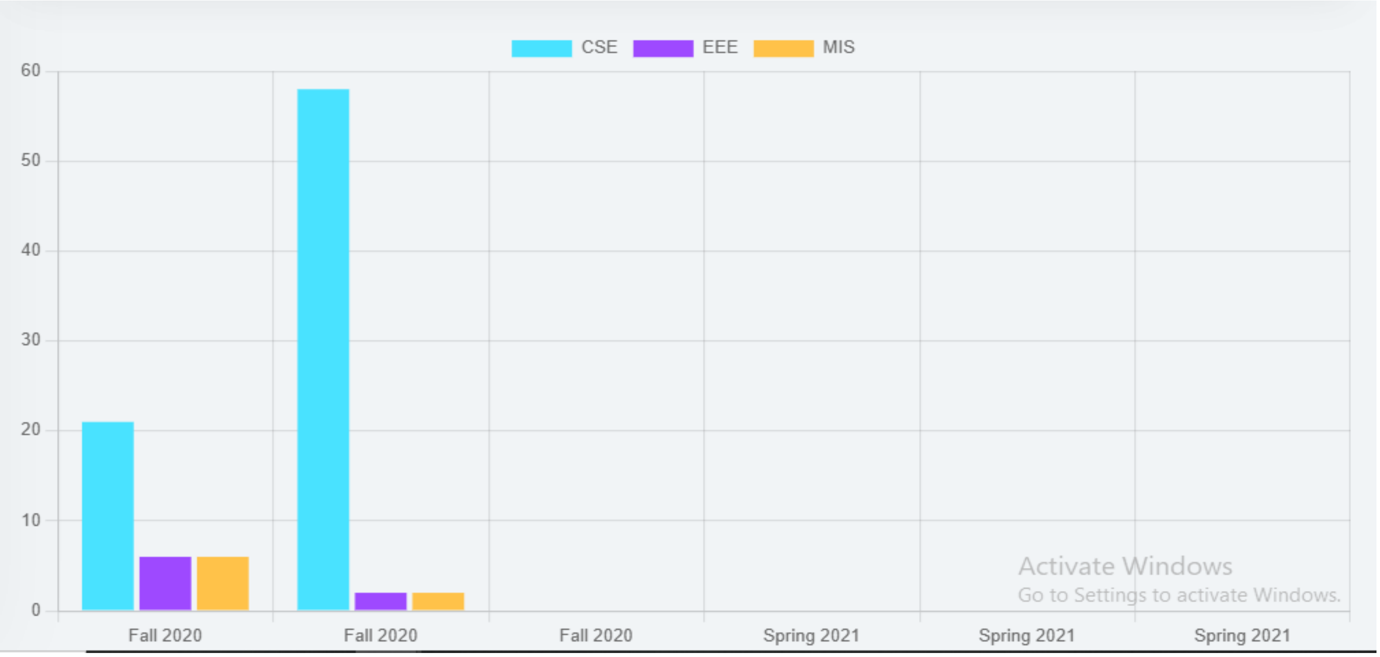
SELECT stu.semesterid, dept.schoolid, COUNT(*) as studentcount from tblstudent stu JOIN tbldepartment dept on stu.departmentid = dept.departmentid GROUP by stu.semesterid, dept.schoolid



Student List

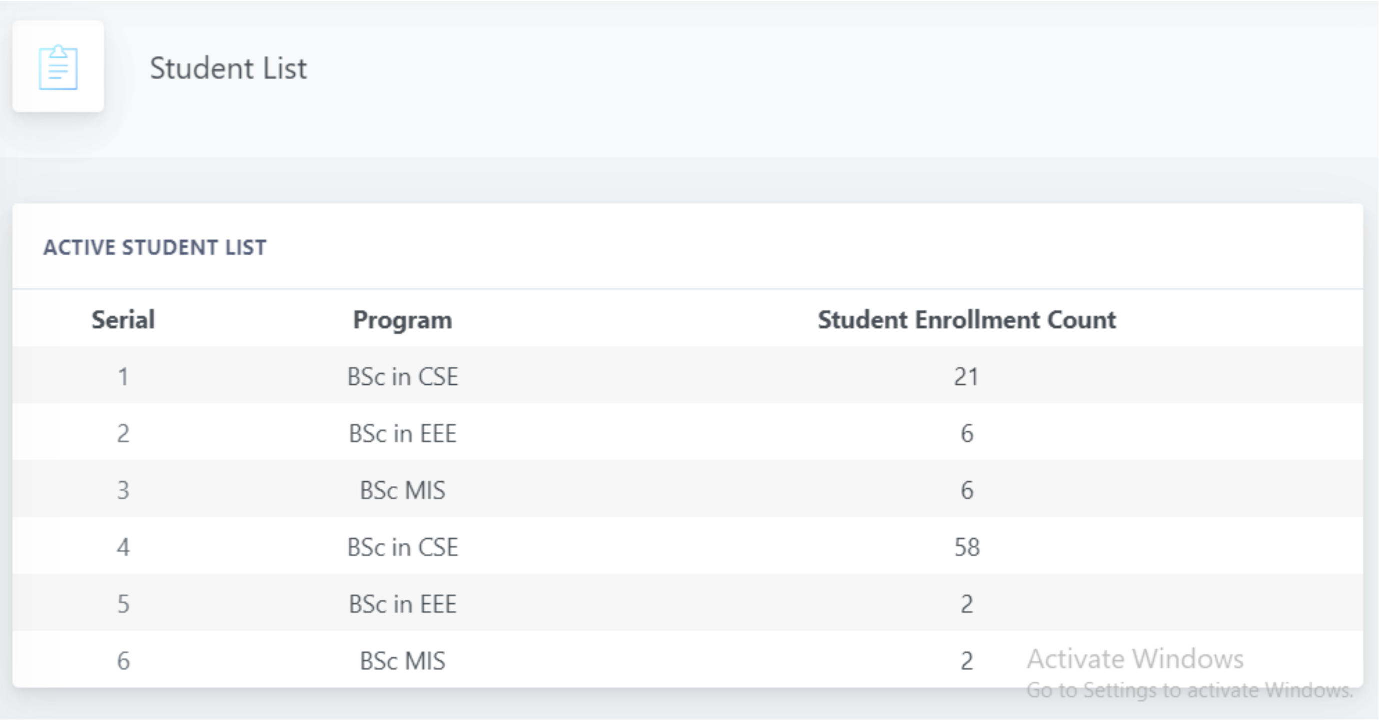
ACTIVE STUDENT LIST

Serial	Department	Student Enrollment Count
1	CSE	21
2	EEE	6
3	MIS	6
4	CSE	58
5	EEE	2
6	MIS	2



Department-Wise Student Enrollment:

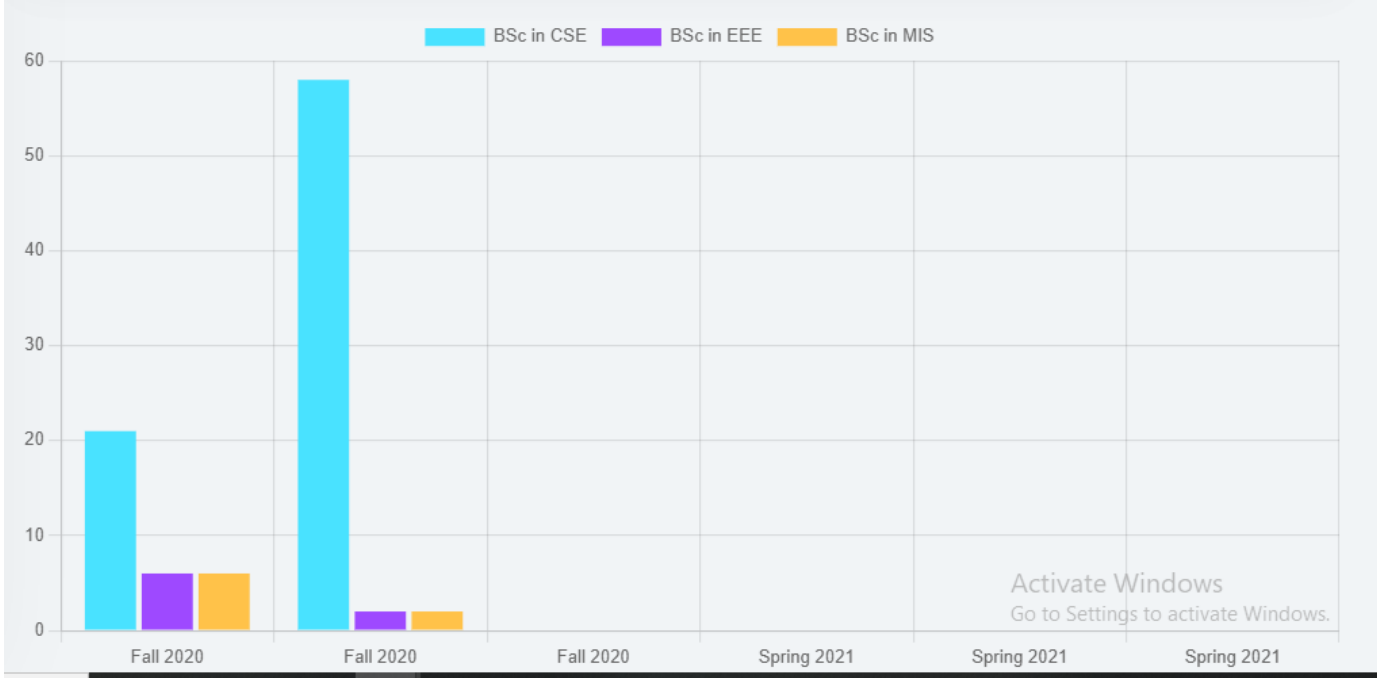
SELECT stu.semesterid, prog.departmentid, COUNT(*) as studentcount from tblstudent stu JOIN tblprogram prog on stu.programid = prog.programid GROUP by stu.semesterid, prog.departmentid



Student List

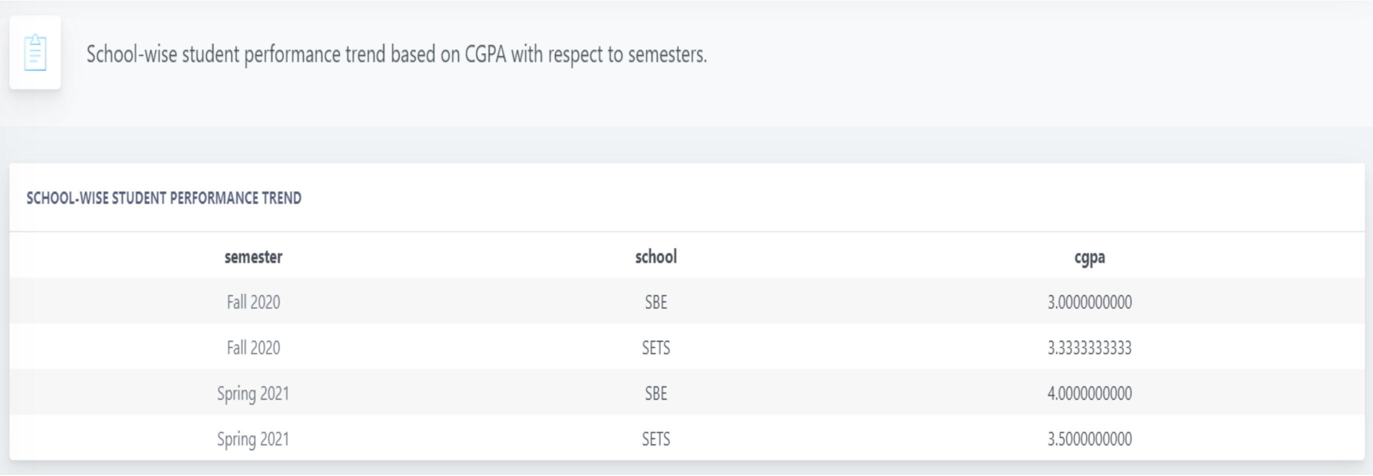
ACTIVE STUDENT LIST

Serial	Program	Student Enrollment Count
1	BSc in CSE	21
2	BSc in EEE	6
3	BSc MIS	6
4	BSc in CSE	58
5	BSc in EEE	2
6	BSc MIS	2



Program-wise Student Enrollment:

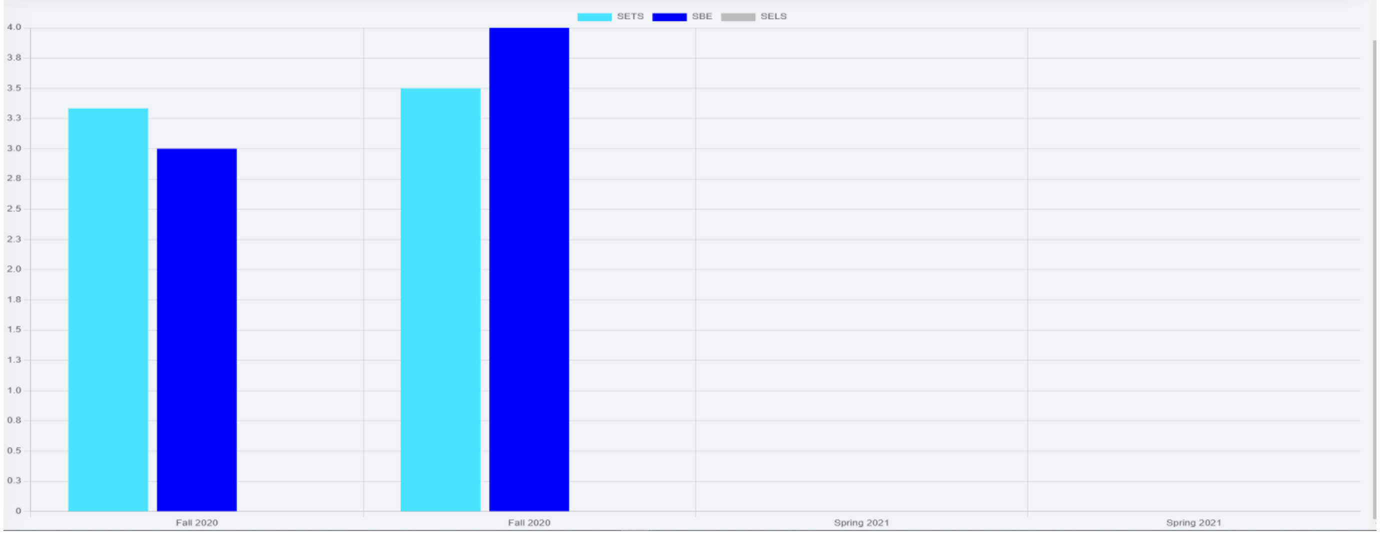
SELECT stu.semesterid, prog.programid, COUNT(*) as studentcount from tblstudent stu JOIN tblprogram prog on stu.programid = prog.programid GROUP by stu.semesterid, prog.programid



School-wise student performance trend based on CGPA with respect to semesters.

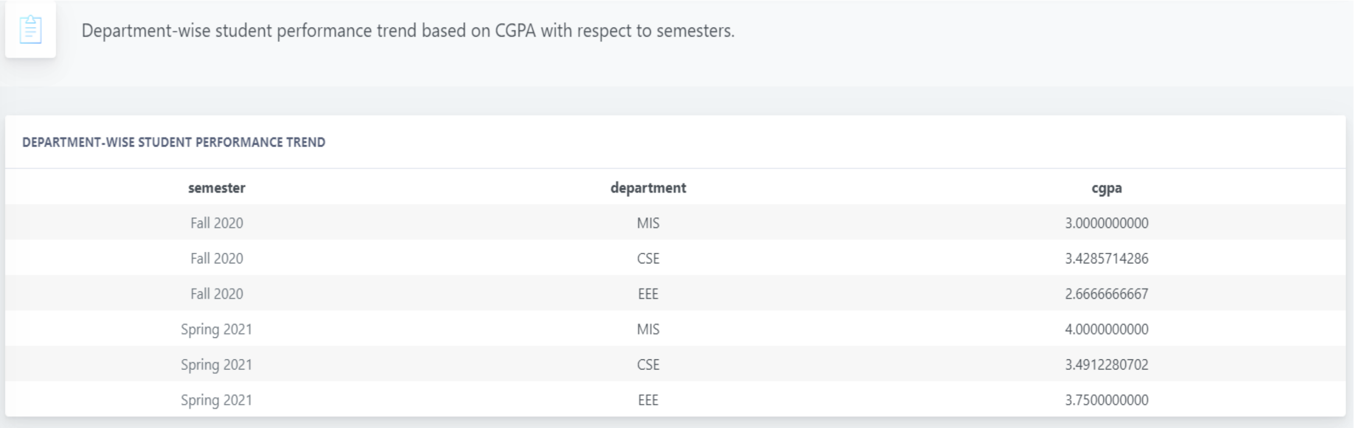
SCHOOL-WISE STUDENT PERFORMANCE TREND

semester	school	cgpa
Fall 2020	SBE	3.0000000000
Fall 2020	SETS	3.3333333333
Spring 2021	SBE	4.0000000000
Spring 2021	SETS	3.5000000000



School-wise Student Performance Trend:

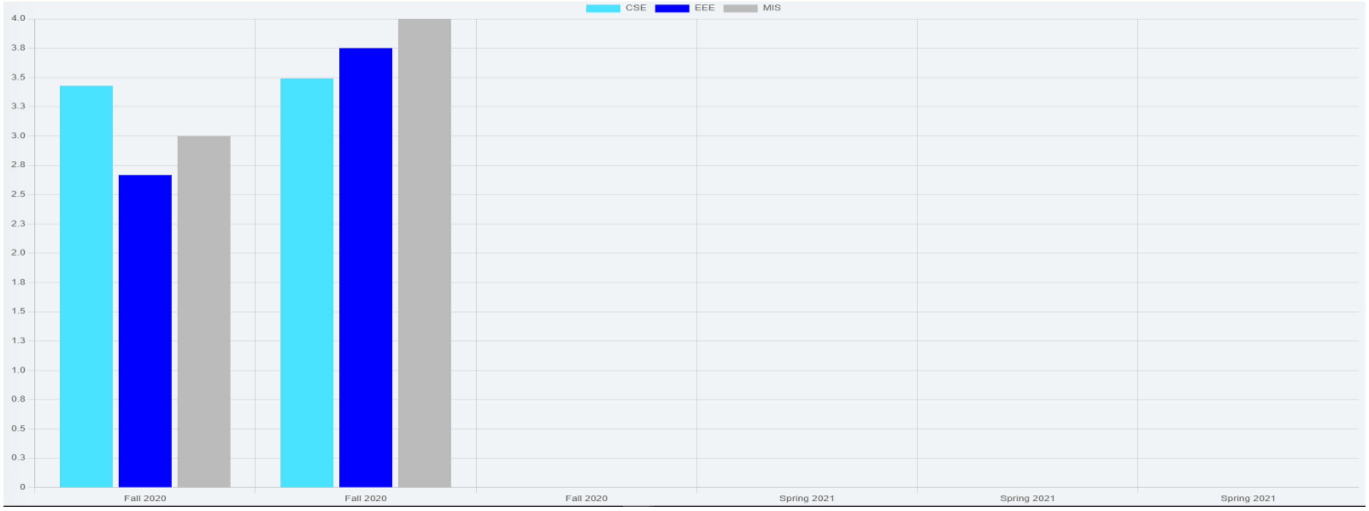
```
SELECT semesterid , schoolid,SUM(CGPA)/COUNT(studentid) AS cgpa FROM

(SELECT semesterid , schoolid, studentid, SUM(CG)/COUNT(courseid) AS cgpa FROM

(SELECT s.semesterid , d.schoolid,  c.courseid, s.studentid, (CASE WHEN SUM(sub.obtainedmark)>
85 then 4.00 WHEN SUM(sub.obtainedmark)> 80 then 3.75 WHEN SUM(sub.obtainedmark)> 75
then 3.50 WHEN SUM(sub.obtainedmark)> 70 then 3.25 WHEN SUM(sub.obtainedmark)> 65 then
3.00 WHEN SUM(sub.obtainedmark)> 60 then 2.75 WHEN SUM(sub.obtainedmark)> 55 then 2.50
WHEN SUM(sub.obtainedmark)> 50 then 2.25 WHEN SUM(sub.obtainedmark)> 45 then 2.00 WHEN
SUM(sub.obtainedmark)> 40 then 1.75 else 0.0 end) as CG

FROM tblstudent s, tbldepartment d, tblcourse c, tblassessment a, tblassessmentsubmission sub

WHERE s.departmentid = d.departmentid AND s.studentid = sub.studentid AND a.assessmentid =
sub.assessmentid

Group BY s.semesterid , d.schoolid, c.courseid, s.studentid) AS subquary

GROUP BY semesterid , schoolid,  studentid) AS subquary2

GROUP BY semesterid , schoolid
```



Department-wise student performance trend based on CGPA with respect to semesters.

DEPARTMENT-WISE STUDENT PERFORMANCE TREND

semester	department	cgpa
Fall 2020	MIS	3.0000000000
Fall 2020	CSE	3.4285714286
Fall 2020	EEE	2.6666666667
Spring 2021	MIS	4.0000000000
Spring 2021	CSE	3.4912280702
Spring 2021	EEE	3.7500000000



Department-wise student performance:

SELECT semesterid , schoolid, departmentid , programid, SUM(cgpa)/COUNT(studentid) AS cgpa FROM(SELECT semesterid , schoolid, departmentid , programid, studentid, SUM(CG)/COUNT(courseid) AS cgpa FROM(SELECT s.semesterid , d.schoolid, s.departmentid , s.programid, c.courseid, s.studentid, (CASE WHEN SUM(sub.obtainedmark)> 85 then 4.00 WHEN SUM(sub.obtainedmark)> 80 then 3.75 WHEN SUM(sub.obtainedmark)> 75 then 3.50 WHEN SUM(sub.obtainedmark)> 70 then 3.25 WHEN SUM(sub.obtainedmark)> 65 then 3.00 WHEN SUM(sub.obtainedmark)> 60 then 2.75 WHEN SUM(sub.obtainedmark)> 55 then 2.50 WHEN SUM(sub.obtainedmark)> 50 then 2.25 WHEN SUM(sub.obtainedmark)> 45 then 2.00 WHEN SUM(sub.obtainedmark)> 40 then 1.75 else 0.0 end) as CG

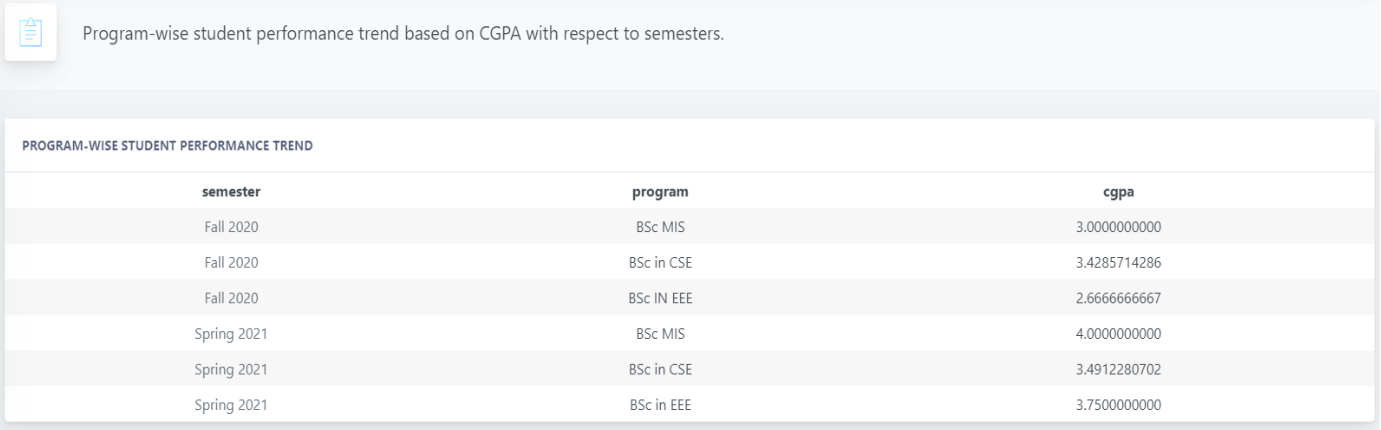
FROM tblstudent s, tbldepartment d, tblcourse c, tblassessment a, tblassessmentsubmission sub

WHERE s.departmentid = d.departmentid AND s.programid = c.programid AND s.studentid = sub.studentid AND a.assessmentid = sub.assessmentid

Group BY s.semesterid , d.schoolid, s.departmentid , s.programid, c.courseid, s.studentid) AS subquary

GROUP BY semesterid , schoolid, departmentid , programid, studentid) AS subquary2

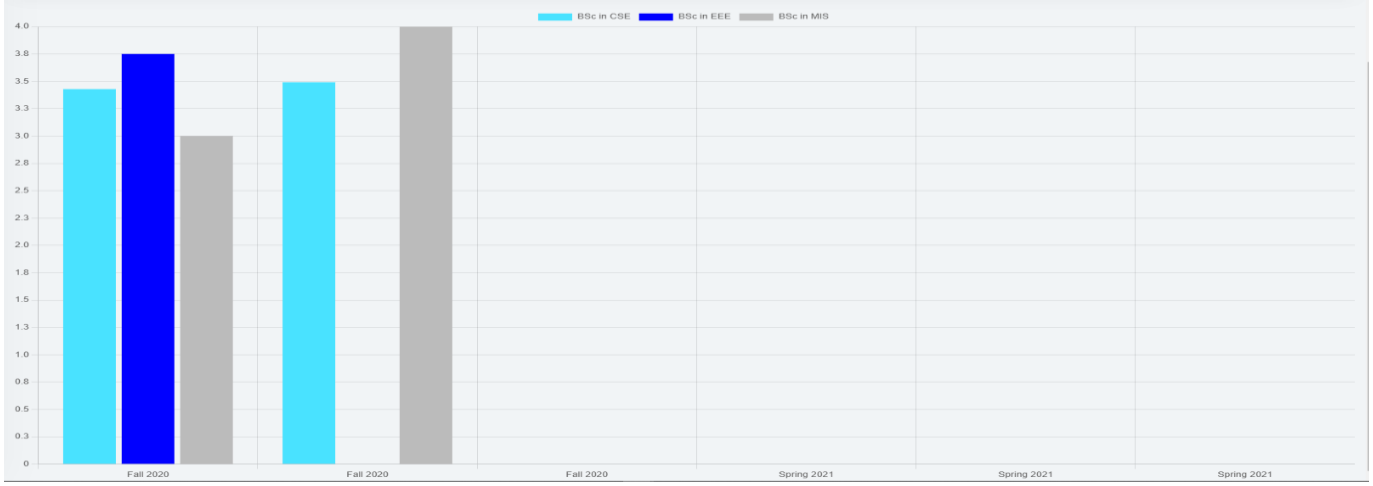
GROUP BY semesterid , schoolid, departmentid , programid



Program-wise student performance trend based on CGPA with respect to semesters.

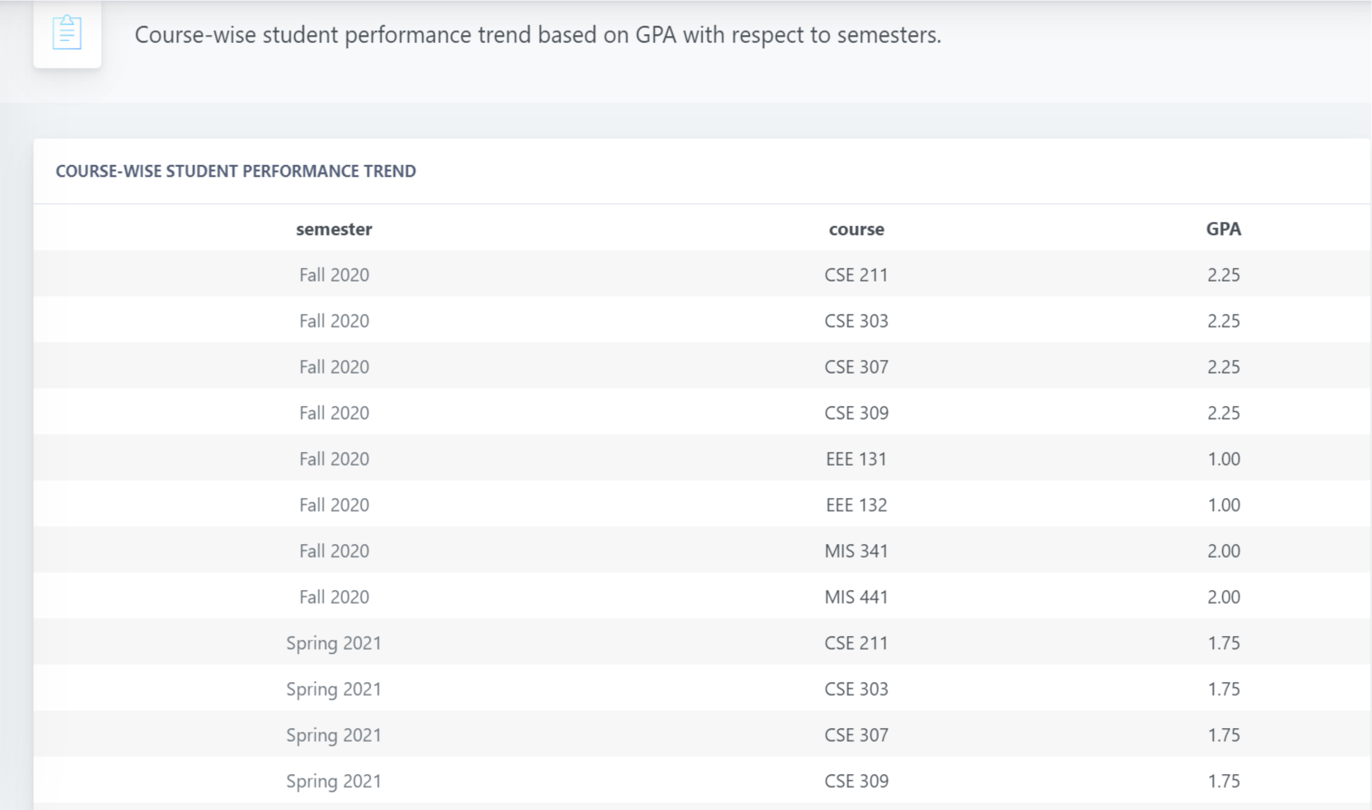
PROGRAM-WISE STUDENT PERFORMANCE TREND

semester	program	cgpa
Fall 2020	BSc MIS	3.0000000000
Fall 2020	BSc in CSE	3.4285714286
Fall 2020	BSc IN EEE	2.6666666667
Spring 2021	BSc MIS	4.0000000000
Spring 2021	BSc in CSE	3.4912280702
Spring 2021	BSc in EEE	3.7500000000



Program-wise student performance:

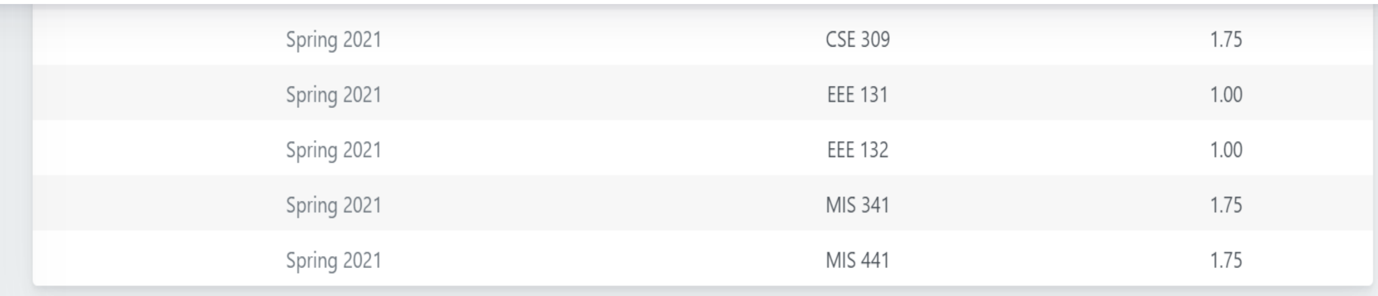
```
SELECT semesterid , schoolid, departmentid , programid, SUM(cgpa)/COUNT(studentid) AS cgpa
FROM(SELECT semesterid , schoolid, departmentid , programid, studentid,
SUM(CG)/COUNT(courseid) AS cgpa FROM

(SELECT s.semesterid , d.schoolid, s.departmentid , s.programid, c.courseid, s.studentid, (CASE
WHEN SUM(sub.obtainedmark)> 85 then 4.00 WHEN SUM(sub.obtainedmark)> 80 then 3.75 WHEN
SUM(sub.obtainedmark)> 75 then 3.50 WHEN SUM(sub.obtainedmark)> 70 then 3.25 WHEN
SUM(sub.obtainedmark)> 65 then 3.00 WHEN SUM(sub.obtainedmark)> 60 then 2.75 WHEN
SUM(sub.obtainedmark)> 55 then 2.50 WHEN SUM(sub.obtainedmark)> 50 then 2.25 WHEN
SUM(sub.obtainedmark)> 45 then 2.00 WHEN SUM(sub.obtainedmark)> 40 then 1.75 else 0.0 end)
as CG

FROM tblstudent s, tbldepartment d, tblcourse c, tblassessment a, tblassessmentsubmission sub

WHERE s.departmentid = d.departmentid AND s.programid = c.programid AND s.studentid =
sub.studentid AND a.assessmentid = sub.assessmentid

Group BY s.semesterid , d.schoolid, s.departmentid , s.programid, c.courseid, s.studentid) AS
subquary

GROUP BY semesterid , schoolid, departmentid , programid, studentid) AS subquary2

GROUP BY semesterid , schoolid, departmentid , programid
```



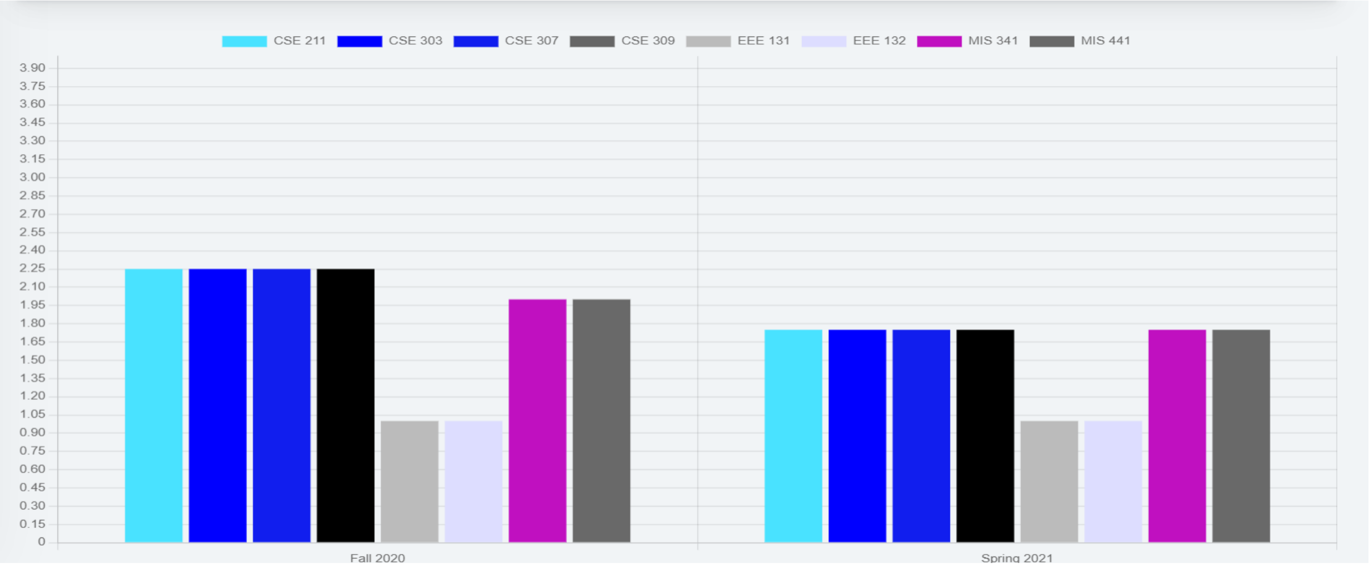
Course-wise student performance trend based on GPA with respect to semesters.

COURSE-WISE STUDENT PERFORMANCE TREND

semester	course	GPA
Fall 2020	CSE 211	2.25
Fall 2020	CSE 303	2.25
Fall 2020	CSE 307	2.25
Fall 2020	CSE 309	2.25
Fall 2020	EEE 131	1.00
Fall 2020	EEE 132	1.00
Fall 2020	MIS 341	2.00
Fall 2020	MIS 441	2.00
Spring 2021	CSE 211	1.75
Spring 2021	CSE 303	1.75
Spring 2021	CSE 307	1.75
Spring 2021	CSE 309	1.75

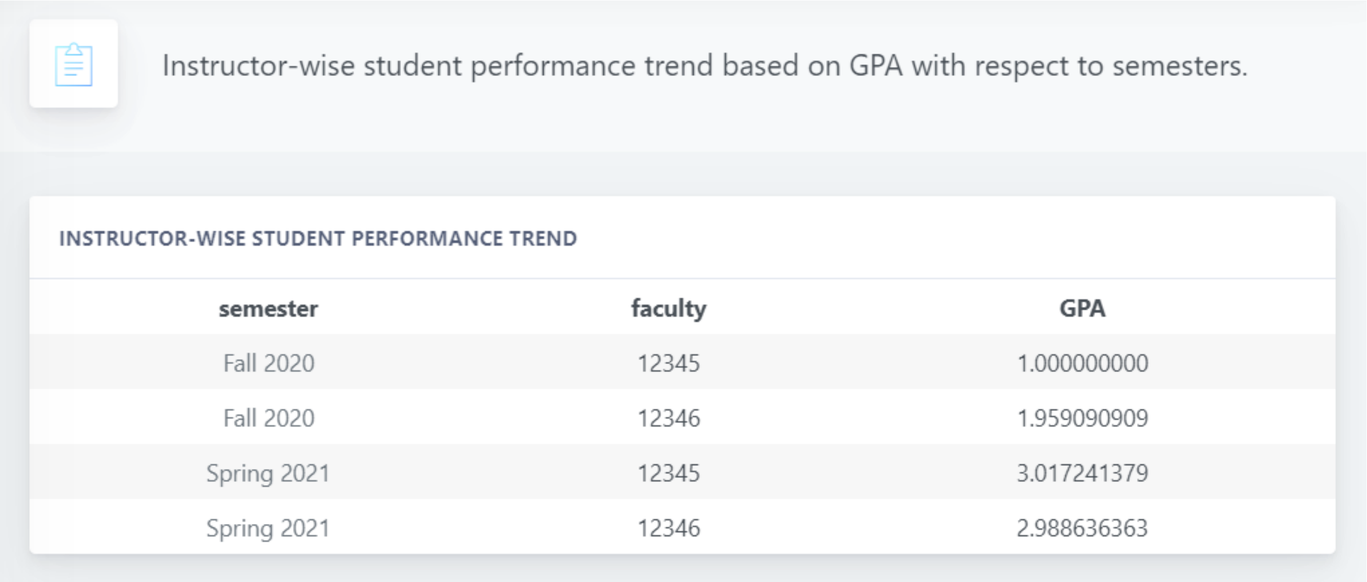


Spring 2021	CSE 309	1.75
Spring 2021	EEE 131	1.00
Spring 2021	EEE 132	1.00
Spring 2021	MIS 341	1.75
Spring 2021	MIS 441	1.75



Course-Wise Student performance based on GPA:

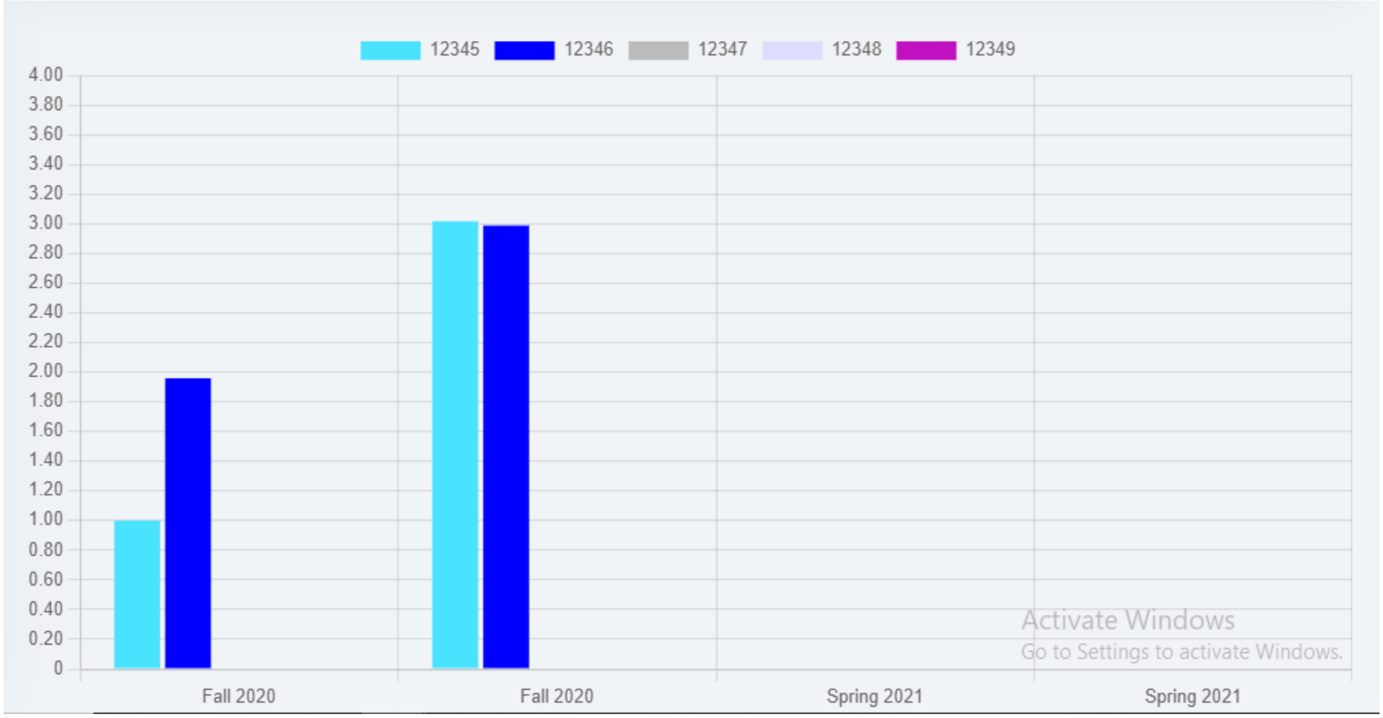
```
SELECT semesterid, courseid, (case when AVG(acivemark)>85 then 4.00  when AVG(acivemark)>80
then 3.75  when AVG(acivemark)>75 then 3.50 when AVG(acivemark)>70 then 3.25 when
AVG(acivemark)>65 then 3.00 when AVG(acivemark)>60 then 2.75  when AVG(acivemark)>55 then
2.50 when AVG(acivemark)>50 then 2.25 when AVG(acivemark)>45 then 2.00 when
AVG(acivemark)>40 then 1.75 else 1 end) as gpa

FROM

(SELECT s.semesterid, c.courseid, s.studentid, SUM(sub.obtainedmark)/sum(allocatedmark)*100 as
acivemark

from tblassessment a, tblassessmentsubmission sub,tblstudent s,tblcourse c

where a.assessmentid = sub.assessmentid AND sub.studentid= s.studentid AND s.programid=
c.programid

group by s.semesterid, c.courseid, s.studentid)

as Subquery

GROUP BY semesterid, courseid
```



Instructor-wise student performance trend based on GPA with respect to semesters.

INSTRUCTOR-WISE STUDENT PERFORMANCE TREND

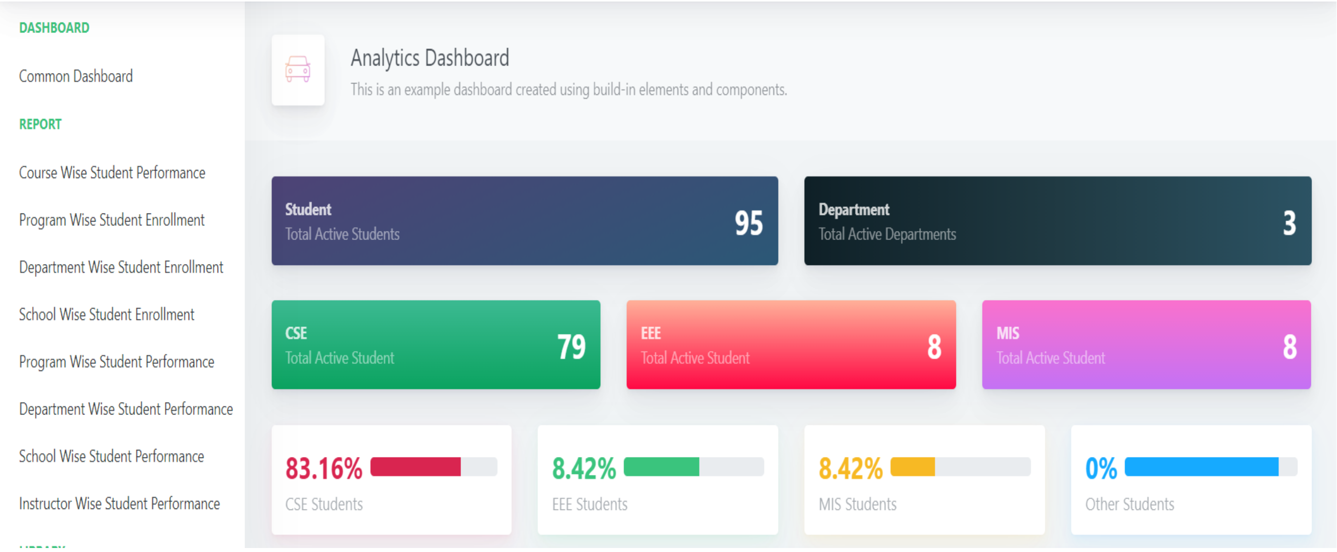
semester	faculty	GPA
Fall 2020	12345	1.000000000
Fall 2020	12346	1.959090909
Spring 2021	12345	3.017241379
Spring 2021	12346	2.988636363



```
SELECT semesterid, facultyid, SUM(gpa)/count(studentid) as gpa FROM

(SELECT semesterid, facultyid, studentid, SUM(gpa)/count(courseid) as gpa from

(SELECT sec.semesterid, faculty.facultyid, student.studentid, course.courseid, (case when
SUM(obtainedmark)>90 then 4.0 when SUM(obtainedmark)>85 then 3.7 when
SUM(obtainedmark)>80 then 3.3 when SUM(obtainedmark)>75 then 3.0 when
SUM(obtainedmark)>70 then 2.7 when SUM(obtainedmark)>65 then 2.3 when
SUM(obtainedmark)>60 then 2.0 when SUM(obtainedmark)>55 then 1.7 when
SUM(obtainedmark)>50 then 1.3 when SUM(obtainedmark)>45 then 1.0 else 0.0 end) as gpa
```

```
 FROM tblsection sec
JOIN tblcourse course on sec.courseid = course.courseid
JOIN tblfaculty faculty on sec.facultyid = faculty.facultyid
JOIN tblco co on co.courseid = course.courseid
JOIN tblassessment assess on co.coid = assess.coid
JOIN tblassessmentsubmission submission on assess.assessmentid = submission.assessmentid
JOIN tblstudent student on submission.studentid = student.studentid
GROUP BY sec.semesterid, faculty.facultyid, student.studentid, course.courseid) as subquery
GROUP BY semesterid, facultyid, studentid) as subquery2
GROUP BY semesterid, facultyid
```

Analytics Dashboard


DASHBOARD
Common Dashboard
REPORT
Course Wise Student Performance
Program Wise Student Enrollment
Department Wise Student Enrollment
School Wise Student Enrollment
Program Wise Student Performance
Department Wise Student Performance
School Wise Student Performance
Instructor Wise Student Performance

Analytics Dashboard

This is an example dashboard created using build-in elements and components.

Student
Total Active Students 95

Department
Total Active Departments 3

CSE
Total Active Student 79

EEE
Total Active Student 8

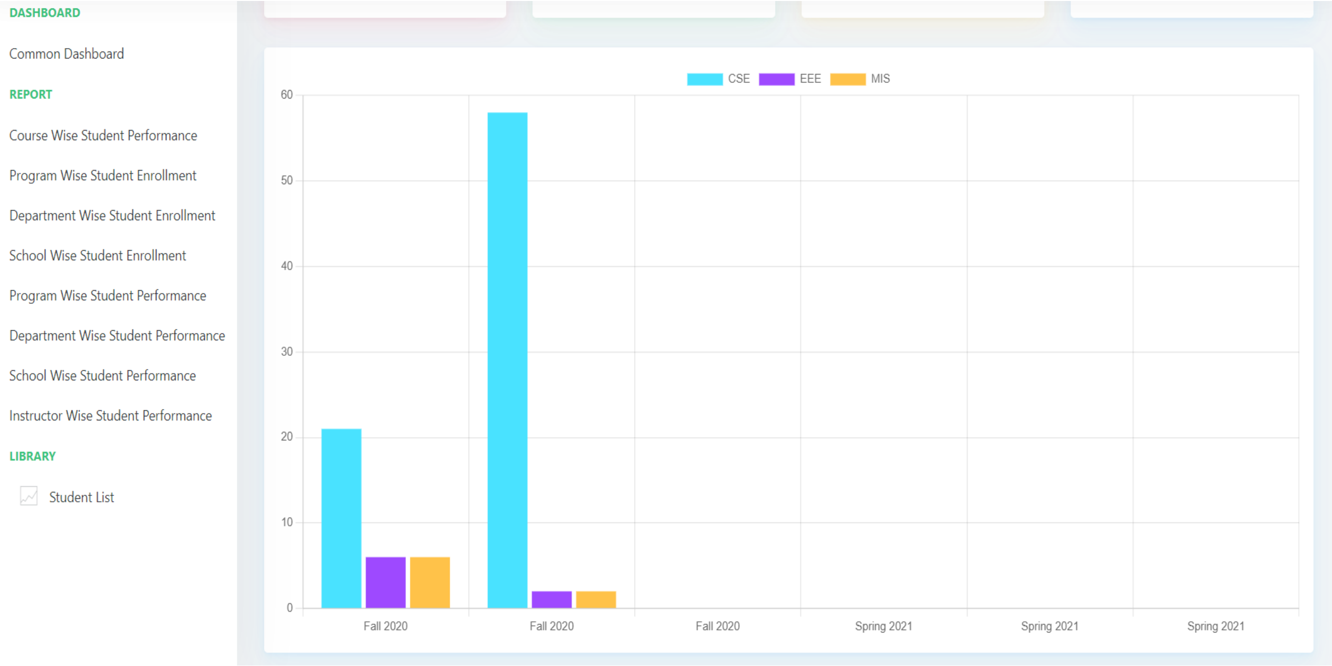
MIS
Total Active Student 8

83.16% CSE Students

8.42% EEE Students

8.42% MIS Students

0% Other Students


DASHBOARD
Common Dashboard
REPORT
Course Wise Student Performance
Program Wise Student Enrollment
Department Wise Student Enrollment
School Wise Student Enrollment
Program Wise Student Performance
Department Wise Student Performance
School Wise Student Performance
Instructor Wise Student Performance
LIBRARY
Student List
CSE
EEE
MIS
60
50
40
30
20
10
0
Fall 2020
Fall 2020
Fall 2020
Spring 2021
Spring 2021
Spring 2021

```
SELECT stu.semesterid, prog.departmentid, COUNT(*) as studentcount from
tblstudent stu JOIN tblprogram prog on stu.programid = prog.programid GROUP by
stu.semesterid, prog.departmentid
```

CHAPTER 5

- CONCLUSION
- Problem and Solution

Problem & Solution:

1.When drawing the Rich Picture, we had to face the challenge of keeping the sequence in the right order.

2.Had to take decisions on groups meetings for the BCNF part in Normalizations as we were not sure that it will exist or not.

3. We used draw.io for BPMN and had difficulties understanding the sequence.

4.When creating the database, we had to find the exact entity and attributes but we were not sure to include it from our report that we already created or the project templates which had been provided. We still have confusions in this specific part but we think it is perfect as the project template is only for data entry and our report is based on overall SPEMS system...

5.We had problems understanding the exact relation for the Entity in ERD. We had to go through our class sides to get an idea.