



Database Management Project

School of Engineering, Technology & Science

Computer Science and Engineering

Section 1, Final Report

Group 9

Name	ID
Sheikh Mehenaz Rahman	2021261
Taiseer Rakin Ahad	1911242
Zahin Faiza Ahmed	1921949
Syed Golam Sadik	1820236
Rubaiya Anwar Mahia	1830357
Syed Shahriyar Ahmed	1711283

Table of Contents

CHAPTER-1 INTRODUCTION.....	3
A. BACKGROUND OF THE ORGANIZATION:.....	4
B. BACKGROUND OF THE PROJECT SPMS 4.0:.....	6
C. OBJECTIVE OF THE PROJECT SPMS 4.0:	6
D. SCOPE OF THE PROJECT:	7
CHAPTER:2 REQUIREMENT ANALYSIS	8
A. Rich Picture- Existing System (SPMS 3.0).....	8
B. Six Elements Analysis: Existing System (SPMS 3.0).....	10
C. Existing Problems And Analysis Of The Problems.....	21
D. Rich Picture- Proposed System (SPMS 4.0).....	28
E. Six Elements Analysis- Proposed System (SPMS 4.0)	29
CHAPTER: 3 LOGICAL BUSINESS DESIGN	41
A. Business Rule – SPMS 4.0.....	41
B. Entity Relationship Diagram.....	43
C. Entity Relationship Diagram to Relational Schema	44
D. Normalization.....	45
E. Data Dictionary.....	59
CHAPTER-4 PHYSICAL SYSTEM DESIGN.....	69
A. INPUT FORMS:.....	69
B. OUTPUT FORMS:.....	78
CHAPTER-5 CONCLUSION.....	102
A.PROBLEM AND SOLUTION:	102
Analysis Phase:	102
Designing Phase:	102
Implementation Phase:	102
Additional Features and Future Development:	102

CHAPTER-1 INTRODUCTION

The Independent University, Bangladesh (IUB) has robust and versatile schools - notably consisting of following:

- Business & Entrepreneurship
- Engineering, Technology & Sciences
- Environment and Life Sciences
- Liberal Arts & Social Sciences
- Pharmacy and Public Health.

The university has been an active participant in the growth of the education sector in Bangladesh and produced capable and knowledgeable scholars contributing both here and abroad. IUB has done this by collaborating closely with the Ministry of Education, the University Grants Commission (UGC), and other appropriate institutions for each of the schools, routinely upgrading its curriculum, implementing a system to track student performance based on a quantitative approach between course curriculum content and standards set by UGC and the Bangladesh government, and constantly tracking student performance for every semester – mainly, using Outcome-Based Education (OBE) for monitoring performance and setting university curriculum.

The focus of this report is to study the current student performance monitoring system that IUB implements, do the required analysis of its processes, and propose a new and better improved system that reduces error, makes analysis of data and report generation easier by all vested quarters and produce/show valuable information needed for IUB and its collaborators in making necessary improvements in academia to produce better scholars. The first part focuses on the details of the organization in question and the project that we have undertaken for it. The second part focuses on the existing system and its shortcomings and an introduction of the proposed system that we plan to replace the existing system with. The third and fourth will be heavily technical and focus on how we plan to bring the proposed system into being.

During our research into the current system for tracking student performance, we identified a number of areas where beneficial changes could be made to speed up the monitoring of student performance processes, facilitate communication among key stakeholders, eliminate the possibility of errors and data duplication, and most importantly, make it simpler for all stakeholders to navigate through large datasets and find the relevant information they need. We'll delve deeper into the workings of the current student performance monitoring system as we read this report, as well as the business procedures involved, where there are data management concerns and problems, and how we can create a better system to address these problems for fixing and improvement.

A. BACKGROUND OF THE ORGANIZATION:

The top private institution in Bangladesh, Independent University, Bangladesh (IUB), was founded in 1993 and has a clear emphasis on research and international collaborations. IUB is attempting to ensure education to promote the national agenda of sustainable inclusive development with a current enrollment of roughly 10,000 at the undergraduate and graduate levels, more than 450 highly qualified faculty members, 47% of whom have PhDs, and an alumni strength of close to 14,000 members.

The objectives of IUB are to produce graduates of international standards in the local setting, equipped with the knowledge and necessary abilities to exercise leadership in business, government, and welfare; to encourage and support fruitful research; to generate knowledge; and to offer opportunities for adult learners to continue their education. Undergraduate curriculum goals include teaching communication skills (oral and writing communication, as well as computer skills) with a strong humanities background, as well as knowledge and proficiency in a subject (i.e., a major) and a sub-specialization (minor) in a related field. This is in line with the ultimate objective of giving students the background and middle management abilities needed to seek higher education. The mission is to achieve the goals of higher education and of sustainable economic growth in the country through a two-way relationship between community and university.

The University curriculum and courses of study are progressively revised and adjusted on the

basis of their relevance to national needs and the global market demand. The educational quality is maintained by exchange relationships with renowned universities overseas. As a private university, IUB is free to set its own curriculum and methods of procedure and to cooperate with academics at home and abroad. It produces graduates who are well grounded in the liberal arts with marketable skills to promote economic growth.

IUB, over-time, has shown remarkable outcomes in producing graduates with marketable skills only because of staying disciplined and up to date with the on-going curriculum and progress system. Dedicating attention towards IUB's Departments, and more specifically focusing the Department of Computer Science and Electrical science into a well-funded research hub running several research projects. IUB is also committed to curve potential graduates of international standard who are mainly equipped to provide new leadership to the national economy through skilled employment, entrepreneurship and/or applied research. This is successful due to the overwhelming support of the Bangladesh Government and the UGC for IUB to be able to create state-of-the-art lab facilities in their department. It is because of IUB's approach to academics as an "Application Oriented Learning" philosophy that "not only teaches students the fundamental principles of learning, situation -handling, and have better overall perception by providing them with hands-on training sessions."

Continuously growing its lab facilities and flourishing on its curriculum according to current market economic demands, the SECS and the Department of Computer Science and Engineering at IUB has constantly worked with IEB, UGC and the Ministry of Education to track their students' overall performance under specific periods by quantifying specific courses and its relating assessments into measurable trackers to gain valuable insights for improvement of students over the years as a student in a certain department.

These processes and criteria credentials courses are ultimately set by IEB along with relevant government potentials to set the bar for up-coming graduating engineers from top universities in Bangladesh. These sets of standards come in the form of Program Educational Objectives (PEO) and Program Learning Outcomes (PLO) [1] for specific departments in an Accreditation Manual which are mapped to specific courses by

relevant Course Instructors and Co-Ordinator. This allows the Department of CSE at IUB, SECS, IEB and all other relevant stakeholders to have a calculating assessment of the current state-of-affairs and the performance of each student under each course for every semester. This will also allow users to track performance of faculties, courses, departments and schools and provides valuable insight for making necessary improvements.

B. BACKGROUND OF THE PROJECT SPMS 4.0:

Measuring the output of students, faculties, departments, and their respective courses in order to measure their productivity in regard to the outcome relevance of the course activities. Basically, to provide a range of tools and data intended to help universities and education authorities such as IEB, UGC, as well as other stakeholders to evaluate the performance of students and inform strategies for improvements. Developing a national framework for Outcome-Based Education while at the same time leaving considerable freedom to universities in implementing local approaches.

C. OBJECTIVE OF THE PROJECT SPMS 4.0:

The SPMS 3.0 system monitors and summarizes the performances of the stakeholders - students, faculties, schools, and departments through the database of the assessments. For evaluation purposes the system would be able to store individual assessment marks (midterm, quizzes, assignment, projects, presentations and so on). As well as the marks of those assessments with respect to their Course Outcomes (CO) and Program Learning Outcomes (PLO) accordingly in the database of the system to observe the outcome and performance of the student's faculties, schools, and departments.

The students being the primary stakeholder, would be able to statistically directly monitor the overall performance to their satisfaction of certain course objectives. Hence based on their performances and faculty evaluation the higher stakeholders (Head of department and Admin) can understand and manage the degree in comparison to which different course outcomes targets and their achievements are being understood by the student, department, school, and university body as a whole. SPSMS 3.0 also monitors the impact of policies against overall administrative

goals and targets by the system. The system's main target is to monitor the whole university activities through the database and produce analytics for the Head of Department, Faculty, School, Students, and their Courses in a given period of time (yearly and semester wise).

D. SCOPE OF THE PROJECT:

We did a complete analysis of the existing system and found out places in the business processes which can cause severe lapses in time and communication, which we will discuss in the next chapter.

Our solution is to create a Web application, called SPMS 3.0 (Student Performance Monitoring System 3.0), using a Relational Database Management System (RDMS) to store, edit, add, and update necessary data for monitoring student performance and producing and storing related OBE data, reports, and documents.

We produced potential users for the web based SPMS 3.0 system and speculated how they would be using the system and the necessary information and data they would need access to. Since the problems can arise from many points of all business processes, we will make custom user interfaces and login capabilities for all stakeholders who will also be the users of this system.

Since we use a (RDBMS) for data storage, retrieving necessary files, tabular data, page layouts and reports becomes incredibly easy and allows us to interact with the necessary data to occur real-time. We also create interfaces for all users to easily access these data and use them to generate and download reports. We build an interface for faculties to be able to collaborate with each other on developing course outlines, course reports, marksheets, assessments, mapping assessments to CO's and PLOs for PLO achievements, and record assessments of students throughout the semester for all their courses.

Students, the IUB leadership team and government agencies can also access the systems for drawing conclusions. Data will also be protected, and each stakeholder will be shown only that data, which is relevant to them, respectively.

CHAPTER:2 REQUIREMENT ANALYSIS

Requirement Analysis is the process of determining what the database is used for. It involves interviews with stakeholders in order to identify the functionality and system requirements they expect and require from the database, what operations need to be performed and what data they need to process. By doing so, we can get a proper understanding of the stakeholders and how they interact with each other.

Requirements analysis is a team effort that demands a combination of hardware, software and human factors engineering expertise as well as skills in dealing with people.

The Requirement Analysis is the means of using industry tools, methods, and standards, to research and visualize the current system and the processes that go into the business operation of a certain organization. “Requirements Analysis is the process of determining what the database is to be used for. It involves interviews with user groups and other stakeholders to identify what functionality they require from the database, what kinds of data they wish to process and the most frequently performed operations.”

By doing this we can see each stakeholder and how they interact with each other. We use simple notations and symbols to give anyone the idea of how a business process works and dissect it accordingly.

A. Rich Picture- Existing System (SPMS 3.0)

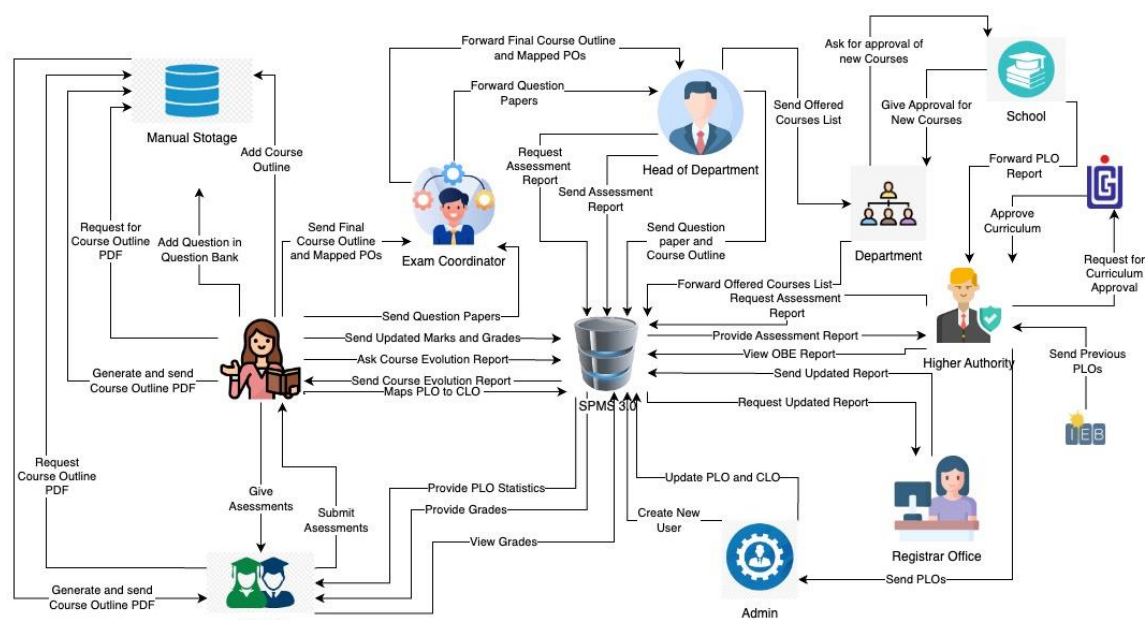


Figure 1.1 Rich Picture of Existing system (SPMS 3.0)

The Rich Picture Analysis shows us that we have the following types of stakeholders:

- Instructor
- Department Head
- Student
- Registrar Office
- Exam Coordinator
- Higher Authority (Dean, VC)
- IEB
- UGC

We can also identify three separate storage systems or facilities, namely:

- Department Storage
- Registrar's Office Storage
- Question paper Storage
- Course Outline Storage

B. Six Elements Analysis: Existing System (SPMS 3.0)

From this Rich Picture we have drawn out 7 processes that are key to monitoring student performance and improving curriculum. The processes are as follows:

- Map Course Outcomes (COs) to Program Learning Outcomes (PLOs).
- Record Student Assessment Data
- Produce Question Paper
- View Assessment Reports over a given time-period for inspection and analysis of student performance trends.
- Produce Course Outline
- View Records OBE Marksheets, Course
- Produce OBE Marksheet & Course Assessment Report

The Six Elements Analysis provides a detailed description of the role of each element in each process. It is clear from the table below that Human entities dominate all key functions of this system (especially in the most critical two processes- mapping course outcomes and viewing documents related to them.) For example, the current system is heavily dependent on manually processed and handled hardcopy databases. Thus, there is a significantly long chain of waiting between interdependent procedures before the Human elements can fulfill their end of the bargain in the process.

Process	Human	Non-Computing Hardware	Computing Hardware	Software	Database	Network and Communication
Performance of students based on CGPA	Student: The system calculates the CGPA and creates a transcript while also uploading the grades.		Laptop/Computer: All assessment s' data is recorded on a computer or laptop.	SPMS 3.0: It is an application that allows instructors to upload	SPMS 3.0 Database: The SPMS database stores	Internet: To log into the system and access the database, one uses the Internet.

	<p>Once they log into the system, the students may see all of these.</p> <p>Department Head: Logs into the system to monitor and analyze the departmental, school wise or program-level CGPA trend for a single student or a group of students for a given semester.</p> <p>Register's Office: Logs into the system to monitor and analyze the departmental, school wise or program-level CGPA trend for a single student or a group of students for a given semester.</p> <p>Faculty: Logs into the system to monitor and</p>		<p>To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>and updates student performance data per semester.</p>	
--	---	--	--	---	---	--

	<p>analyze the departmental, school wise or program-level CGPA trend for a single student or a group of students for a given semester.</p> <p>Dean/VC: Logs into the system to monitor and analyze the departmental, school wise or program-level CGPA trend for a single student or a group of students for a given semester.</p>					
GPA-based course wise students' performance	<p>Student: Logs into the system to obtain the mark for a specific course and to verify the GPA by reading the transcript.</p> <p>Faculty: Logs into the system to check the GPA trend of</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile:</p>	<p>SPMS 3.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result,</p>	<p>SPMS 3.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>students enrolled in a given course for a particular section.</p> <p>Department Head: Logs into the system to check the GPA trend of students enrolled in a given course for the semester.</p> <p>Register' Office: Logs into the system to check the GPA trend of students enrolled in a given course for the semester.</p> <p>Dean/VC: Logs into the system to check the GPA trend of students enrolled in a given course for the semester.</p>		<p>Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>the user may see the pattern here.</p>		
--	---	--	--	---	--	--

Selection of Instructors based on GPA and performance in class	<p>Department Head: By logging into the system, you may see the GPA trend of students taking courses from a particular semester from a particular group of professors.</p> <p>Faculty: By logging into the system, you may see the GPA trend of students taking courses from a particular semester from a particular group of professors.</p> <p>Dean: By logging into the system, you may see the GPA trend of students taking courses from a particular semester from a particular group of professors.</p>					
Student	Department		Laptop/Co	SPMS	SPMS	Internet:

<p>performance based on GPA is evaluated by the vice chancellor, dean, or department head.</p>	<p>Head: By logging into the system, users can monitor the GPA trends of students during the tenure of a particular department head, dean, or vice chancellor.</p> <p>Dean/VC: By logging into the system, users can monitor the GPA trends of students during the tenure of a particular department head, dean, or vice chancellor.</p> <p>Register's Office: By logging into the system, users can monitor the GPA trends of students during the tenure of a particular department head, dean, or vice chancellor.</p>		<p>Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>3.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>3.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>To log into the system and access the database, one uses the Internet.</p>
---	---	--	---	--	---	---

<p>Assessing student's GPAs to determine Instructor wise performance for a particular course</p>	<p>Department Head: By logging into the system, you can see the GPA trend of students who took a certain course from a particular instructor.</p> <p>Faculty: By logging into the system, you can see the GPA trend of students who took a certain course from a particular instructor.</p> <p>Register's Office: By logging into the system, you can see the GPA trend of students who took a certain course from a particular instructor.</p> <p>Dean/VC: By logging into the system, you can see the GPA trend of students who took a certain course</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>SPMS 3.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>SPMS 3.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>
---	---	--	--	---	--	--

	from a particular instructor.					
Total percentage of PLOs attempted and attained by the student, as well as the department's average percentage of PLOs attained.	<p>Student: By logging onto the system, users can check their own department's average PLO achievement as well as a comparison of PLOs attempted vs. achieved.</p> <p>Faculty: By logging onto the system, users can observe the department's average PLO achievement as well as a comparison of students' attempted vs. achieved PLOs.</p> <p>Department Head: By logging onto the system, users can observe the department's average PLO achievement as well as a</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>SPMS 3.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>SPMS 3.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>comparison of students' attempted vs. achieved PLOs.</p> <p>Register's Office: By logging onto the system, users can observe the department's average PLO achievement as well as a comparison of students' attempted vs. achieved PLOs.</p> <p>Dean/VC: By logging onto the system, users can observe the department's average PLO achievement as well as a comparison of students' attempted vs. achieved PLOs.</p>					
Comparison of PLO achieved vs attempted	<p>Student: By logging into the system, you can see a comparison of PLO achievements</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer</p>	<p>SPMS 3.0: It is an application that allows instructors</p>	<p>SPMS 3.0 Database: The SPMS database</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>vs attempts.</p> <p>Department Head: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p> <p>Register's Office: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p> <p>Faculty: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p> <p>Dean/VC: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p>		<p>or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>stores and updates student performance data per semester.</p>	
PLO achieved vs expected score	<p>Student: By logging into the system, you can see a comparison of</p>		<p>Laptop/Computer: All assessment s' data is</p>	<p>SPMS 3.0: It is an application that</p>	<p>SPMS 3.0 Database: The</p>	<p>Internet: To log into the system and access the database, one</p>

	<p>PLO achievements vs attempts.</p> <p>Department Head: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p> <p>Register's Office: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p> <p>Faculty: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p> <p>Dean/VC: By logging into the system, you can see a comparison of PLO achievements vs attempts.</p>		<p>recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>SPMS database stores and updates student performance data per semester.</p>	<p>uses the Internet.</p>
--	--	--	--	--	--	---------------------------

C. Existing Problems And Analysis Of The Problems

Based on the existing systems' Six Elements Analysis, the shortcomings in each process were identified. There is a repeating pattern in the far-right column of this table. It appears that the facilitation of a private online platform will improve the system in many ways.

Process Name	Stakeholders	Concerns (Problems)	Analysis (Reason of the problems)	Proposed Solution
Student performance based on CGPA	1. Student 2. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	School-wise, program-wise and department wise student performance trends based on CGPA for a respective semester.	All the mentioned stakeholders have been able to observe the updated CGPA status every semester individually.	The new system allows the analysis of the CGPA trend based on not only the Student's individual performance but also based on the School, department and program.
Course-wise performance based on GPA	1. Student 2. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	Course-wise student performance trends based on GPA for a respective semester.	The GPA of every individual semester was not associated with course wise performance based on the GPA.	The system allows the stakeholders to view the student performance based on GPA for each selected course for a respective semester.

Selective Number of Instructor- Wise student performance based on the GPA of the students	1. Student 2. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	Instructor- wise Student performance trends based on GPA for a respective semester.	The higher authorities were not able to understand the instructor's performanc e as well based on the GPA of the students.	The system will record the GPA of students taught by a specific instructor and generate graphs showing the trend of the performance of the Student based on the GPA to measure the performance of the instructor for the respective semester.
--	---	---	--	---

VC-wise Dean-wise or Department Head-wise student performance based on GPA	1. Student 2. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	Higher Authority-wise and department head wise Student performance trends based on GPA for a respective semester.	The higher authorities were not able to view higher authority wise or department head wise performance based on the GPA of the students.		The system will record the GPA of students under higher authorities and Department Head and generate graphs showing the trend of the performance of the Student based on the GPA for a respective semester.
Instructor wise student performance for a specific course based on GPA	1. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	Instructor wise Student performance trends based on GPA for a respective semester.	The higher authorities were not able to view instructor wise student performance for a specific course based on the GPA of the students.		The system will record the GPA of students taught by a specific instructor for a specific course and generate graphs showing the trend of the performance of the Student based on the GPA for a respective semester.

Total PLO percentage achieved and attempted by the student	1. Student 2. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	PLO total percentage score for each PLO calculated from the scores achieved in each CO associated with the corresponding	The PLO and corresponding CO for all the courses the student has done so far is never compared cumulatively		The system will provide the total of all PLO percentages corresponding to CO and calculate the score for all the courses a student has done
---	--	--	---	--	---

Along with the Departmental average PLO achievement		PLO among all the courses the student has done so far, along with the departmental average performance for comparison. Also, for each PLO, what percentage of it was achieved from each of the courses associated with the corresponding PLO, and what percentage was achieved via each of all the COs associated with the corresponding PLO. All of this	along the departmental average performance.	for a chosen school, program, or department.
--	--	---	---	--

		for a chosen school, program, or department.		
--	--	--	--	--

Comparison of PLO achieved vs attempted	1. Student 2. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	Comparison of PLO achieved percentage versus PLO attempted percentage.	Students are unable to compare progress of their PLO achieved vs PLO they should be aiming for with respect to courses they have done.	The system would allow the students and rest of the stakeholders to monitor automatically using relational data models using proper SQL operations-their PLO achieved vs attempted comparisons individually.
--	--	--	--	--

Expected PLO achieved vs attempted score	1. Student 2. Department Head 3. Registrar's office 4. Faculty 5. Higher Authority	Comparison of a course's, student's, department's, program's, or school's expected PLO achievement versus actual with respect to a given period of time/semesters .	The existing system does not provide adequate information for comparisons of PLO.	The system would allow the stakeholders to monitor automatically their PLO achieved vs attempted comparisons for courses, program's, departments, and school with respect to a given period of time/semesters .
---	--	---	---	---

Course outline generation	1. Department Head 2. Exam Coordinator 3. Faculty	The course outline will be generated for every semester by the instructor.	The course outline had to be generated in Google Docs or Microsoft Word and needed to be formatted accordingly by the faculty which can be very time consuming.	The template available will be used by the faculty to automatically generate the course outline. The faculties will have to map the CO and PLO.
Question bank process	1. Department Head 2. Exam Coordinator 3. Faculty	The question papers stored in the system each semester for each course will form a question bank eventually.	Faculties had to type Questions and map the CO and the questions in Google Docs or Microsoft Word and had to store them in their laptops or cloud storage space.	The system will let the faculties add questions and map the COs with the question and then the question paper will be stored in the system's database and with time a question bank will form.

D. Rich Picture- Proposed System (SPMS 4.0)

The Course Outcomes (COs) and Program Learning Outcomes (PLOs) will be visible in a new system, an online platform called SPMS, where it will have its own database that host the data of all the courses, faculties, as well as updated tables every semester to keep track of which courses have been assigned to which faculties in a given semester.

We are making the new system (to track student performance, but also to track faculties teaching a specific course or the performance of students in a course over a period) and why it is hard to track these trends and data right now.

Briefly, we can see that the SPMS relational database (a non-human) quite literally plays a significant role in the student performance monitoring system. Also, this entity holds the greatest number of interconnections between all other processes.

We will use different user interfaces designed for specific user needs based on the concerns and problems we found in the problem analysis. The Head of the Department/Dean of School, Course Instructor/Coordinator/Faculty, Admin, Student, IEB/UGC/Ministry of Education, VC/Board of Trustees, Department Staff, all these stakeholders mentioned will have access to view the report of a student.

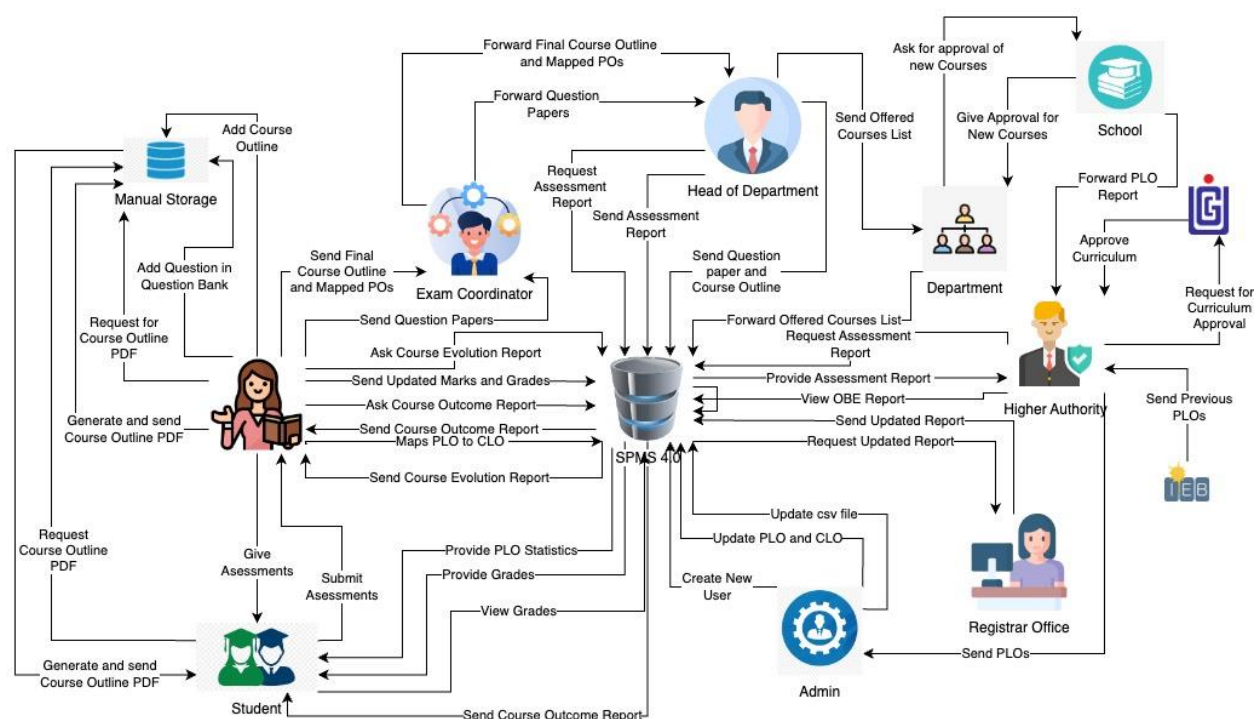


Fig: 1.2 Rich Picture of Proposed System (SPMS 4.0)

E. Six Elements Analysis- Proposed System (SPMS 4.0)

The suggested system's six-element analysis is the next step in a sequence of analysis where each analysis builds on the one that came before it. The table below provides additional insight into the function of each component in the new system based on the detailed picture.

Process	Human	Non-Computing Hardware	Computing Hardware	Software	Database	Network and Communication
Performance of students based on CGPA	Student: The technology calculates the CGPA, uploads the grades, and also generates		Laptop/Computer: All assessment s' data is recorded on	SPMS 4.0: It is an application that allows	SPMS 4.0 Database: The SPMS	Internet: To log into the system and access the database, one uses the

	<p>a transcript. Once they have logged into the system, the students can see all of these.</p> <p>Department Head: enters the system to examine and evaluate the departmental, academic institution, or program-level CGPA trend for a single student or a group of students for a given semester.</p> <p>Register's office: enters the system to examine and evaluate the departmental, academic institution, or program-level CGPA trend for a single student or a group of students for a given semester.</p> <p>Faculty: enters the system to examine and analyze the departmental, academic</p>		<p>a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>database stores and updates student performance data per semester.</p>	<p>Internet.</p>
--	---	--	--	---	---	------------------

	<p>institution, or program-level CGPA trend for a single student or a group of students for a given semester.</p> <p>Dean: enters the system to examine and evaluate the departmental, academic institution, or program-level CGPA trend for a single student or a group of students for a given semester.</p>					
GPA-based course wise students performance	<p>Department Head: enters the system to monitor the GPA trend of students enrolled in a particular course during a particular semester.</p> <p>Faculty: logs into the system to check the GPA trends of students in their own sections in</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>a particular course</p> <p>Register's office: enters the system to monitor the GPA trend of students enrolled in a particular course during a particular semester.</p> <p>Dean: Logging in will allow you to see the GPA trend for students taking a certain course during a particular semester.</p>		<p>mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	here.		
Selection of Instructors based on GPA and performance in class	<p>Department Head: By logging into the system, you can see the GPA trends of students taking courses from a certain group of professors during a particular semester.</p> <p>Faculty: By logging into the system, you can see the</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>GPA trends of students taking courses from a certain group of professors during a particular semester.</p> <p>Register's office: By logging into the system, users can monitor the GPA trends of students taking courses from a particular semester with a particular set of professors.</p> <p>Dean: By logging into the system, users can monitor the GPA trend of students taking courses from a particular semester from a particular group of professors.</p>		<p>access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	may see the pattern here.		
Student performance based on GPA is evaluated by the vice chancellor, dean, or department head.	<p>Department Head: By logging into the system, one can see the GPA patterns of the students during the tenure of a particular</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>Department Head or Dean.</p> <p>Dean: By logging into the system, users can monitor the GPA trends of students during the tenure of a particular Department Head or Dean.</p> <p>Register's office: By logging into the system, users can monitor the GPA trends of students during the tenure of a particular Department Head or Dean.</p>		<p>require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>performance data per semester.</p>	
--	--	--	--	---	---------------------------------------	--

<p>Assessing student's GPAs to determine Instructor wise performance for a particular course</p>	<p>Department Head: By logging into the system, you can see the GPA trend of students who took a certain course from a particular instructor.</p> <p>Dean: By logging into the system, you can see the GPA trend of students who took a certain course from a particular instructor.</p> <p>Faculty: By logging into the system, you can see the GPA trend for students who took a certain course from a particular professor.</p> <p>Register's Office: By logging into the system, you may see the GPA trends of the students who took a particular course from a particular instructor.</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>
---	--	--	--	---	--	--

<p>Total percentage of PLOs attempted and attained by the student, as well as the department's average percentage of PLOs attained.</p>	<p>Student: By logging onto the system, users can check their own department's average PLO achievement as well as a comparison of PLOs attempted vs. achieved.</p> <p>Register's office: By logging onto the system, users can observe the department's average PLO achievement as well as a comparison of students' attempted vs. achieved PLOs.</p> <p>Faculty: By logging onto the system, users can observe the department's average PLO achievement as well as a comparison of students' attempted vs. achieved PLOs.</p> <p>Department Head:</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>
--	--	--	--	---	--	--

	<p>By logging into the system, users can observe the average departmental PLO achievement as well as a comparison of the number of PLOs attempted and obtained by students.</p> <p>Dean: Logging into the system enables users to observe the average departmental PLO achievement as well as a comparison of students' attempted vs. achieved PLOs.</p>					
Comparison of PLO achieved vs attempted	<p>Student: By logging onto the system, you may view the comparison of PLO achievements vs. attempts.</p> <p>Department Head: By logging onto the system, you may view the</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students to view their results or</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student performance data per</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>comparison of PLO achievements vs. attempts.</p> <p>Faculty: By logging onto the system, you may view the comparison of PLO achievements vs. attempts.</p> <p>Register's office: By logging in, you can view a comparison of PLO accomplishments and attempts.</p> <p>Dean: By signing into the system, you can get a comparison of PLO accomplishments and attempts.</p>		<p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	CGPA. As a result, the user may see the pattern here.	semester.	
PLO achieved vs expected score	<p>Student: By logging onto the system, you can view the comparison between the expected PLO achievement and the actual score.</p> <p>Department Head:</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students to view their</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student performance data</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>By logging onto the system, you can view the comparison between the expected PLO achievement and the actual score.</p> <p>Register's office: Logging into the system will let you see the expected PLO achievement compared to the actual score.</p> <p>Faculty: By logging onto the system, you can view the comparison between the expected and actual PLO achievement.</p> <p>Dean: By logging onto the system, you can get a comparison between the expected PLO achievement and the actual score.</p>		<p>laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>results or CGPA. As a result, the user may see the pattern here.</p>	<p>per semester.</p>	
Course Outline Generation	Instructor: Create the course outline		Laptop/Computer: All	SPMS 4.0: It is an	SPMS 4.0 Databas	Internet: To log into the system and

	<p>by mapping the COs and POs using the provided template. Send it on to the register's office after that.</p> <p>Exam Coordinator: Examine the course description that the teachers have sent, then send it to the department head.</p> <p>Head of Department: View the course outline that the Register's office has forwarded, then save it to the database.</p>		<p>assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p> <p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	<p>application that allows instructors to upload grades and students to view their results or CGPA. As a result, the user may see the pattern here.</p>	<p>e: The SPMS database stores and updates student performance data per semester.</p>	<p>access the database, one uses the Internet.</p>
Question Bank Process	<p>Instructor: When creating the test questions, map out the COs and POs. Send it on to the exam coordinator after that.</p> <p>Exam Coordinator: Examine the question paper</p>		<p>Laptop/Computer: All assessment s' data is recorded on a computer or laptop. To view the data, the user will require a computer or laptop.</p>	<p>SPMS 4.0: It is an application that allows instructors to upload grades and students to view their results or CGPA. As</p>	<p>SPMS 4.0 Database: The SPMS database stores and updates student performance data per semester.</p>	<p>Internet: To log into the system and access the database, one uses the Internet.</p>

	<p>that the instructors have forwarded before sending it to the department head.</p> <p>Head of Department: View the question paper that the exam coordinator has forwarded before storing it in the database.</p>		<p>Mobile: Users may access the information using their mobile device.</p> <p>Printer: If necessary, the transcript may be printed.</p>	a result, the user may see the pattern here.		
--	---	--	---	--	--	--

CHAPTER: 3 LOGICAL BUSINESS DESIGN

A. Business Rule – SPMS 4.0

Business rules describe the operations, definitions and constraints that govern the data model. As opposed to the ERD, they are made using regular English sentences so that a non-technical stakeholder can decipher information about the data model without notation knowledge. The business rules that govern our data model are as follows:

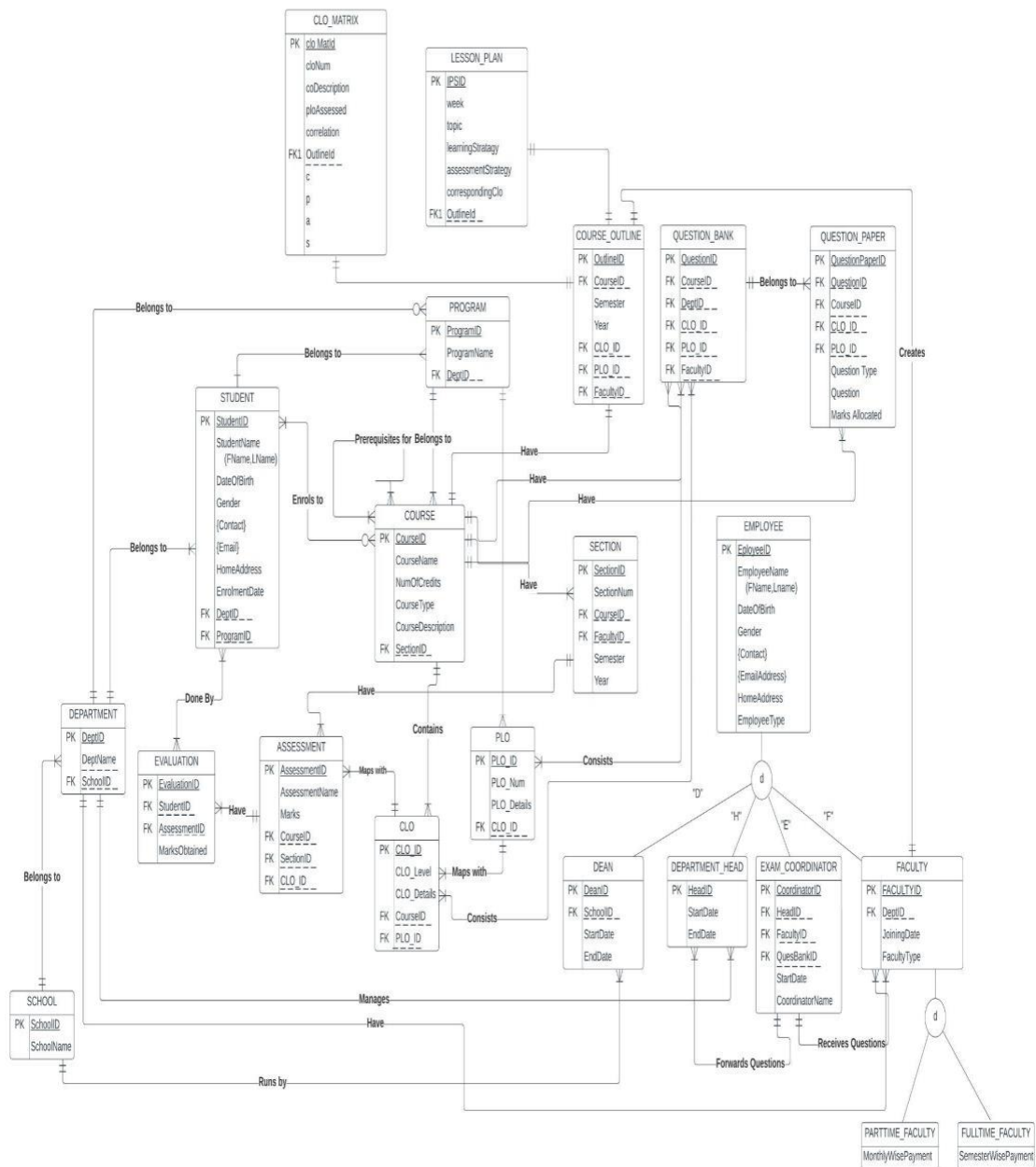
1. A student must belong to one department and a department must have one or more students. A student must belong to one program and a program may have many students. A STUDENT has StudentID, First Name, Last Name, Date of Birth, Gender, Email, Phone number, Address and Date of Enrollment.
2. An employee has 4 sub-types which are Dean, Department Head, Faculty and Exam Coordinator. An EMPLOYEE has EmployeeID, First Name, Last Name, Date of Birth, Gender, Email, Phone number, Address and Date of Enrollment.
3. A course can be a prerequisite of many courses, but a course must belong to one program. A COURSE has CourseID, CourseName, Number of Credits, Course Type and Course

Description. A course can have many sections but it must have at least one section.

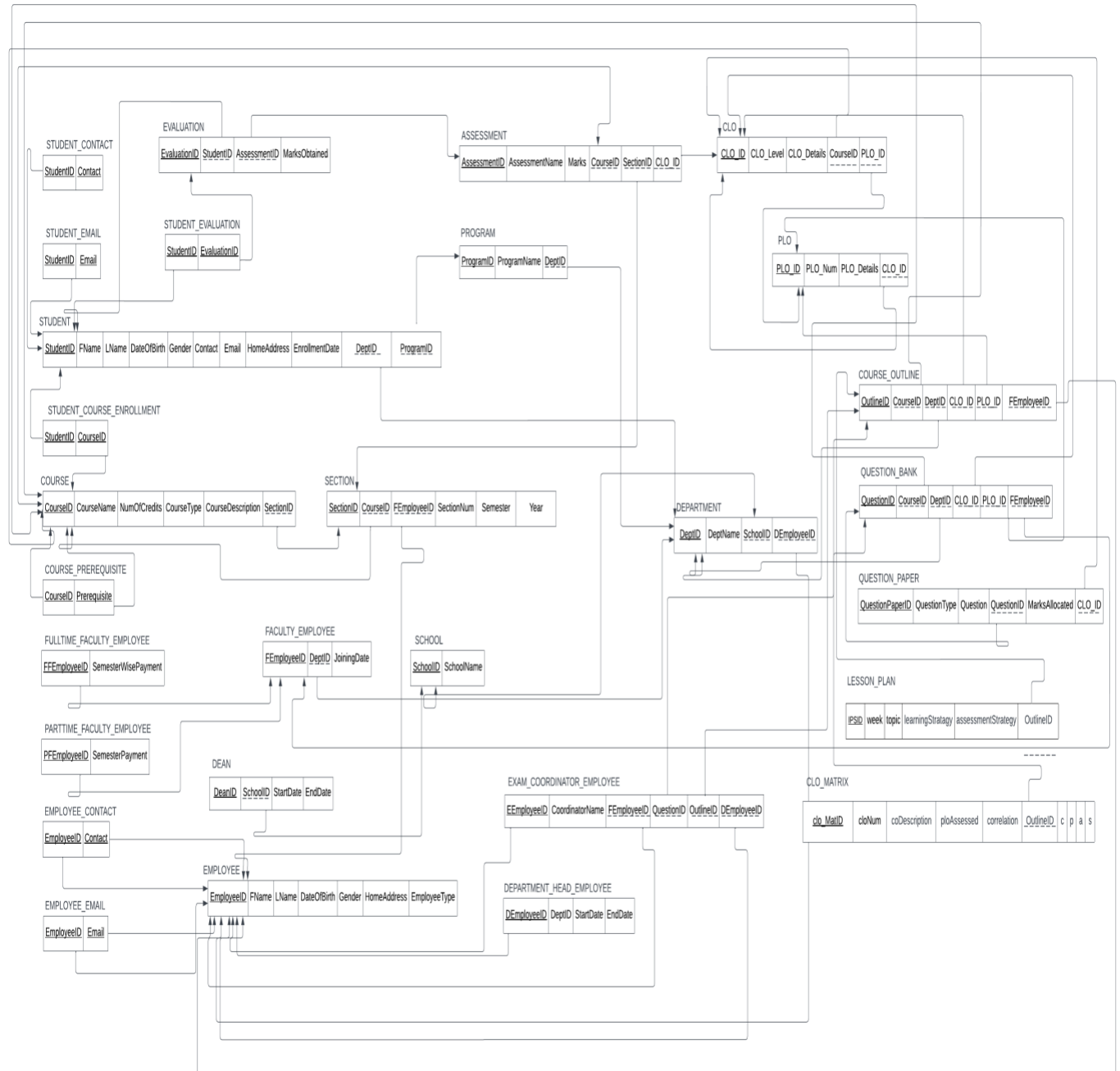
4. A section must belong to one course. A SECTION has SectionID, Section number, Semester and Year. A section must be taught by one faculty and a faculty may teach one or more sections.
5. A School must be run by one or more Deans. A SCHOOL has a SchoolID and SchoolName.
6. A Department must belong to one school and one school can have many departments. A DEPARTMENT consists of DepartmentID and Department Name.
7. A dean can run only one school. A DEAN has a start date and an end date.
8. One department head must manage one department but one department can be managed by one or more department heads. A DEPARTMENT HEAD includes Start Date and End Date.
9. A department must have one or more faculty and one faculty must belong to one department. A FACULTY has a rank and joining date.
10. An evaluation must be done by many students and a student must do one or more evaluations. An EVALUATION consists of EvaluationID and Marks obtained.
11. An assessment must have many evaluations and an evaluation must have one assessment. An ASSESSMENT as AssessmentID, AssessmentName and Marks. An assessment must belong to one section and one section must have one or more assessments.
12. An assessment must map with one CO's. A CO is mapped with one or many assessments. A CO includes COID, Number and CO details. A course must contain many CO's and a CO must be contained in a course.
13. A CO must map with one PLO. A PLO must map with one or many CO's. A PLO includes PLOID, PLO Number and PLO Details.
14. A PLO must contain one program and a program contains one or more PLO's. A PROGRAM has ProgramID and Program Name. A program must contain many courses.
15. An exam coordinator must receive questions from many faculties and one faculty must send questions from one exam coordinator. An exam coordinator must forward questions to many department heads and one department must receive questions from one exam coordinator. An EXAM COORDINATOR has a Coordinator Name and a start date.
16. A course must have a question bank and a question bank must belong to one course. QUESTION BANK consists of QuestionID, Question Type, Question, and Marks Allocated.

17. A COURSE OUTLINE consists of OutlineID, Semester, Year. A faculty must create a course outline and a course outline must be created by one faculty. A course must have a course outline and a course outline must belong to one course. A course outline must consist of one or more PLO's and CO's, and one PLO and one CO must be contained in one or more course outline.

B. Entity Relationship Diagram



C. Entity Relationship Diagram to Relational Schema



D. Normalization

Student	Student ID	a1
	firstname	a2
	last name	a3
	Date of Birth	a4
	Gender	a5
	Contact	a6
	Email	a7
	Address	a8
	Admission Date	a9
	Department ID	h1
	Program ID	g1

Employee	EmployeeID	b1
	First Name	b2
	Last Name	b3
	Date of Birth	b4
	Gender	b5
	Contact	b6
	Email	b7
	Address	b8
	Employee Type	b9

Higher Authority	Higher Authority ID	c1
	School ID	c2
	Start Date	c3
	End Date	c4

Head of the department	HOD_ID	d1
	Department ID	h1
	Start date	d2
	End date	d3

Full-Time Faculty	Monthly Payment	e3
--------------------------	-----------------	----

Part-time Faculty	Semester-wise payment	e4
--------------------------	-----------------------	----

Course	Course ID	f1
	Course Name	f2
	No. of credits	f3
	Section ID	j1

Program	Program ID	f1
	Program Name	f2
	Department ID	f3

Department	Department ID	h1
	School ID	h2
	HOD_ID	d1

School	School ID	i1
	School Name	i2

Section	Section ID	j1
	Section Number	j2
	Course ID	f1
	Faculty ID	e1
	Semester Year	j3

Question Bank	Question Bank ID	k1
	Course ID	f1
	Department ID	h1
	CLO_ID	o1
	PLO_ID	p1

	Faculty ID	b1

Course Outline	Course Outline ID	l1
	Course ID	f1
	Department ID	h1
	CLO_ID	o1
	PLO_ID	p1
	Faculty ID	b1

Evaluati on	Evaluation ID	m1
	Student ID	a1
	Assessment ID	n1
	Marks Obtained	m2

Assessment	Assessment ID	n1
	Assessment Name	n2
	Marks	n3
	Course ID	f1
	Section ID	j1
	CLO_ID	o1

CLO	CLO ID	o1
	CLO Level	o2
	CLO Details	o3
	Course ID	f1
	PLO ID	p1

PLO	PLO ID	p1
	PLO Name	p2
	PLO details	p3
	CLO_ID	o1

Exam Coordin ator	Exam Coordinator ID	q1
	First Name	q2
	Last Name	q3
	Faculty ID	e1

	Course Outline ID	l1
	Question Bank ID	k1
	HOD_ID	d1

Question Paper	Question Paper ID	r1
	Question Type	r2
	Question Bank ID	k1
	Question	r3
	Marks Allocated	r4
	Course ID	f1
	CLO_ID	o1
	CLO Level	o2

Student ID	FirstName,LastName,DateOFBirth,Gender,Contact, Email,Address,Admission Date,Department ID, Program ID
Employee ID	FirstName,LastName,DateOFBirth,Gender,Contact, Email,Address,Employee Type
Higher Authority ID	School ID, Start Date, End Date
HeadOFDepartment ID	Department ID,Start Date, End Date
Faculty ID	Department ID, Date Of Joining
Full-Time Faculty	Monthly Payment
Part-Time Faculty	Semester-wise payment
Course ID	CourseName,No. Of Credits,Section ID
Program ID	ProgramName,Department ID
Department ID	Department Name, School ID, HOD_ID
School ID	School Name

Section ID	Section Number, Course ID ,Faculty ID, Semester Year
Question Bank ID	Course ID, Department ID, CLO ID,Question ID
Course Outline ID	Course ID, Department ID, CLO ID,PLO ID, Faculty ID
Evaluation ID	Student ID, Assessment ID, Marks Obtained
Assessment ID	Assessment Name, Marks, Course ID, CLO ID ,Section ID
CLO ID	CLO Level, CLO details,Course ID,PLO ID
PLO ID	PLO Number, PLO details, CLO ID
Exam Coordinator ID	FirstName, LastName,Faculty ID, Course Outline ID, Question Paper ID,HOD_ID
Question Paper ID	Question ID,Question Type, Question Bank ID, Question, Marks Allocation, CLO ID, CLO Level

a1	a2,a3,a4,a5,a6,a7,a8,a9,h1,g1
-----------	-------------------------------

b1	b2,b3,b4,b5,b6,b7,b8,b9
c1	c2,c3,c4
d1	h1,d2,d3
e1	h1,e2,e3,e4
f1	f2,f3,j1
g1	g2,h1
h1	h2,i1,d1
i1	i2
j1	j2, f1, e1,j3
k1	f1,h1,o1,r1
l1	f1,h1,o1,p1,b1

m1	a1,n1,m2
n1	n2,n3,f1,j1,o1
o1	o2,o3,f1,p1
p1	p2,p3,o1
q1	q2,q3,e1,l1,k1,d1
r1	r2,r3,k1,r4,r5,o1,o2

1NF:-Identifying a primary key

T

a1	a2	a3	a4	a5	a6	a7	a8	a9	b1	b2	b3	b4	b5	b6	b7	b8	b9	<u>c1</u>	c2
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	-----------	----

c3	c4	d1	d2	d3	e1	e2	e3	e4	f1	f2	f3	g1	g2	h1	h2	i1	i2	j1	j2
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

j3	k1	l1	<u>m1</u>	m2	n1	n2	n3	o1	o2	o3	p1	p2	p3	<u>q1</u>	q2	q3	r1	r2	r3	r4	r5
----	----	----	-----------	----	----	----	----	----	----	----	----	----	----	-----------	----	----	----	----	----	----	----

2NF:- Removing partial dependency**T1**

a1	a2	a3	a4	a5	a6	a7	a8	a9	d1	d2	d3	e1	e2	e3	e4	f1	f2	f3	g1
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

g2	h1	h2	i1	i2	j1	j2	j3	<u>m1</u>	m2	n1	n2	n3	o1	o2	o3	p1	p2	p3
----	----	----	----	----	----	----	----	-----------	----	----	----	----	----	----	----	----	----	----

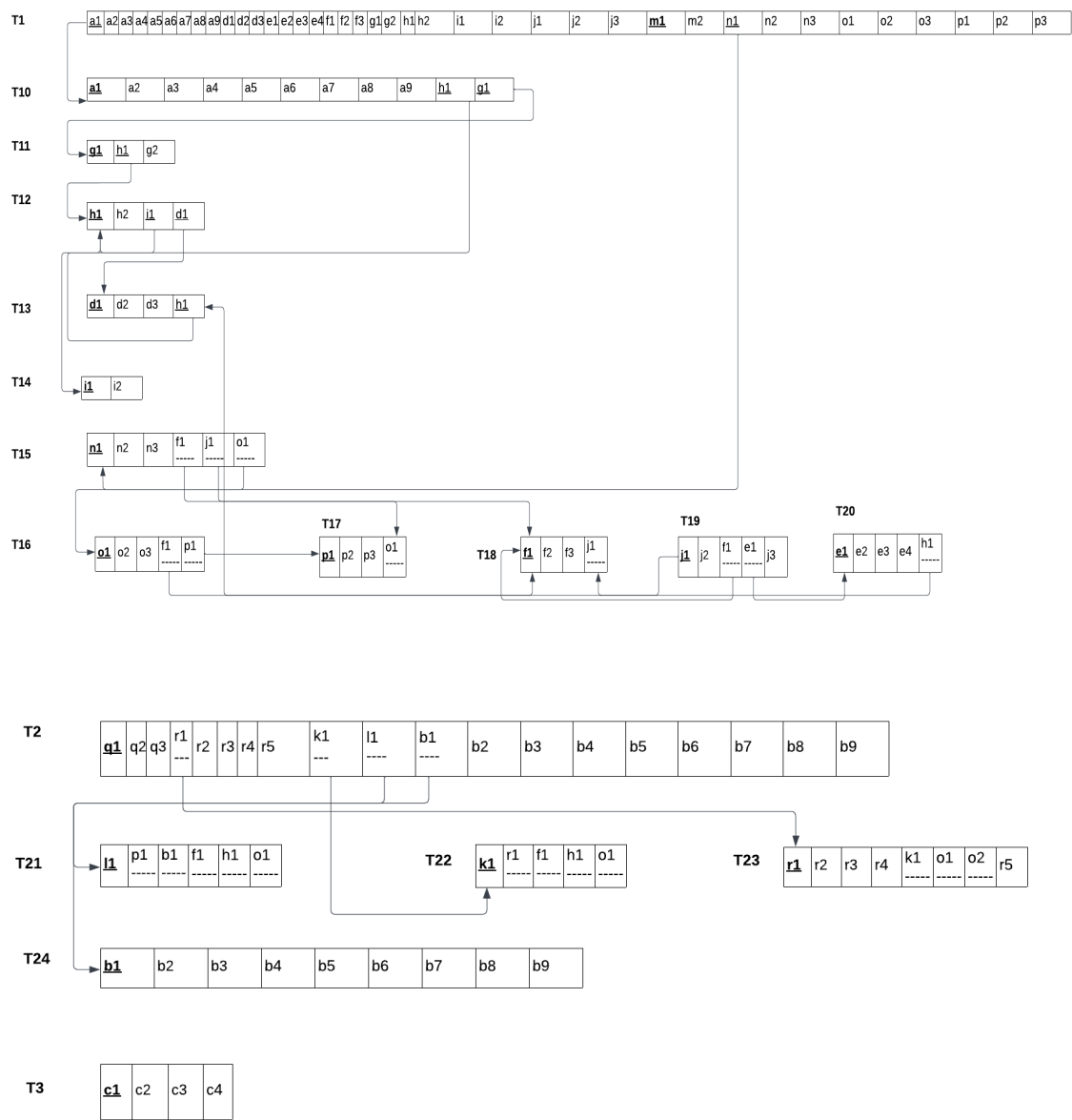
T2

<u>q1</u>	q2	q3	r1	r2	r3	r4	r5	k1	l1	b1	b2	b3	b4	b5	b6	b7	b8	b9
-----------	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

T3

<u>c1</u>	c2	c3	c4
-----------	----	----	----

3NF:-Removing Transitive Dependency



E. Data Dictionary

Student

Name	Data Type	Size	Remarks
------	-----------	------	---------

Student ID	INTEGER		This is the primary key for the student table
FirstName	VARCHAR	50	This is the first name of the student
LastName	VARCHAR	50	This is the last name of the student
Date Of Birth	DATE	DD-MM YYYY	This is the birth date of the student
Gender	VARCHAR	50	This is the gender of the student
Contact	NUMERIC	11	This is the contact of the student
Email	VARCHAR	50	This is the email of the student
Address	VARCHAR	50	This is the address of the student
Admission Date	DATE	DD--MM YYYY	This is the date of enrollment of the student
Departm ent ID	INTEGER		This is the department ID of the student and it is a foreign key in the student table
Program ID	INTEGER		This is the program ID of the student and it is a foreign key in the student table

Employee

Name	Data Type	Size	Remarks
Employee ID	INTEGER		This is the primary key for the employee table
FirstName	VARCHAR	50	This is the first name of the employee

LastName	VARCHAR	50	This is the last name of the employee
Date Of Birth	DATE	DD-MM YYYY	This is the birth date of the employee
Gender	VARCHAR	50	This is the gender of the employee
Contact	NUMERIC	11	This is the contact of the employee
Email	VARCHAR	50	This is the email of the employee
Address	VARCHAR	50	This is the address of the employee
Employee Type	VARCHAR	50	This is the type of the employee

Higher Authority

Name	Data Type	Size	Remarks
Dean ID	INTEGER		This is the primary key for the dean table
School ID	INTEGER		This is the school ID of the higher authority and a foreign key in the dean table
Start Date	DATE	DD-MM YYYY	This is the start date of working of the dean
End Date	DATE	DD-MM YYYY	This is the end date of working of the higher authority

Department Head

Name	Data Type	Size	Remarks

Head ID	INTEGER		This is the primary key for the head of the department table
Department ID	INTEGER		This is the department ID of the head of the department and a foreign key in the head of the department table
Start Date	DATE	DD-MM YYYY	This is the start date of working of the head of the department
End Date	DATE	DD-MM YYYY	This is the end date of working of the head of the department

Faculty

Name	Data Type	Size	Remarks
Faculty ID	INTEGER		This is the primary key for the faculty table
Department ID	INTEGER		This is the department ID of the head of the department and a foreign key in the faculty table
Joining Date	DATE	DD-MM YYYY	This is the start date of working of the faculty
Faculty type	VARCHAR		This is the type of faculty

Full-Time Faculty

Name	Data Type	Size	Remarks
------	-----------	------	---------

Semester-wise payment	VARCHAR		This is the payment type for the full-time faculty table
-----------------------	---------	--	--

Part-Time Faculty

Name	Data Type	Size	Remarks
Semester-wise payment	VARCHAR		This is the payment type for the full-time faculty table

Course

Name	Data Type	Size	Remarks
Course ID	INTEGER		This is the primary key for the Course table
Section ID	INTEGER		This is the section ID of theCourse and a foreign key in the Course table
Course name	VARCHAR	50	This is the name of the Course
No. of credits	INTEGER		This is the number of credits for the Course

Program

Name	Data Type	Size	Remarks
Program ID	INTEGER		This is the primary key for the Program table
Program name	VARCHAR	50	This is the name of the Program

Department ID	INTEGER		This is the department ID and a foreign key in the Program table
---------------	---------	--	--

Department

Name	Data Type	Size	Remarks
Department ID	INTEGER		This is the primary key for the head of the Program table
Department name	VARCHAR	50	This is the name of the Program
Head of department ID	INTEGER		This is the department head ID and a foreign key in the Program table

School

Name	Data Type	Size	Remarks
School ID	INTEGER		This is the primary key for the head of the School table
School name	VARCHAR	50	This is the name of the School

Section

Name	Data Type	Size	Remarks
Section ID	INTEGER		This is the primary key for the section table
Section number	INTEGER		This is the number of the section

Course ID	INTEGER		This is the course ID and a foreign key in the section table
Faculty ID	INTEGER		This is the faculty ID and a foreign key in the section table
Semester Year	DATE	DD-MM YYYY	This is the semester year for the section table

Question Bank

Name	Data Type	Size	Remarks
Question Bank ID	INTEGER		This is the primary key for the question bank table
Course ID	INTEGER		This is the course ID and a foreign key in the question bank table
Department ID	INTEGER		This is the department ID and a foreign key in the question bank table
CLO ID	INTEGER		This is the CLO ID and a foreign key for the question bank table
PLO ID	INTEGER		This is the PLO ID and a foreign key for the question bank table

Course Outline ID

Name	Data Type	Size	Remarks
------	-----------	------	---------

Course Outline ID	INTEGER		This is the primary key for the course outline table
Course ID	INTEGER		This is the course ID and a foreign key in the course outline table
Department ID	INTEGER		This is the department ID and a foreign key in the course outline table
CLO ID	INTEGER		This is the CLO ID and a foreign key for the course outline table
PLO ID	INTEGER		This is the PLO ID and a foreign key for the course outline table
Faculty ID	INTEGER		This is the faculty ID and a foreign key in the course outline table

Evaluation

Name	Data Type	Size	Remarks
Student ID	INTEGER		This is the foreign key for the evaluation table
Evaluation ID	INTEGER		This is the primary key in the evaluation table
Assessment ID	INTEGER		This is the assessment ID and a foreign key in the evaluation table
Marks Obtained	VARCHAR		This is the marks obtained for the evaluation table

Assessment

Name	Data Type	Size	Remarks
------	-----------	------	---------

Assessment ID	INTEGER		This is the primary key for the assessment table
---------------	---------	--	--

Assessment name	INTEGER		This is the name in the assessment table
Course ID	INTEGER		This is the course ID and a foreign key in the assessment table
Marks	VARCHAR		This is the marks obtained for the assessment table
Section ID	INTEGER		This is the section ID and a foreign key in the assessment table
CLO ID	INTEGER		This is the CLO ID and a foreign key in the assessment table

CLO

Name	Data Type	Size	Remarks
CLO ID	INTEGER		This is the primary key in the CLO table
CLO level	VARCHAR		This is the CLO level in the CLO table
CLO details	VARCHAR		This is the CLO details in the CLO table
Course ID	INTEGER		This is the course ID and a foreign key in the assessment table
PLO ID	INTEGER		This is the PLO ID and a foreign key in the assessment table

PLO

Name	Data Type	Size	Remarks
PLO ID	INTEGER		This is the primary key in the PLO table
PLO number	INTEGER		This is the PLO number in the PLO table
PLO details	VARCHAR		This is the PLO details in the PLO table
CLO ID	INTEGER		This is the CLO ID and a foreign key in the PLO table

Exam Coordinator

Name	Data Type	Size	Remarks
Exam Coordinator ID	INTEGER		This is the primary key in the PLO table
First Name	VARCHAR		This is the first name in the Exam coordinator table
Last Name	VARCHAR		This is the last name in the Exam coordinator table
Faculty ID	INTEGER		This is the faculty ID and the foreign key in the Exam coordinator table
Course outline ID	INTEGER		This is the course outline ID and the foreign key in the Exam coordinator table
Question Bank ID	INTEGER		This is the question bank ID and the foreign key in the Exam coordinator table

Head ID	INTEGER		This is the head ID and the foreign key in the Exam coordinator table
---------	---------	--	---

Question Paper

Name	Data Type	Size	Remarks
Question Paper ID	INTEGER		This is the primary key for the question paper table
Question bank ID	INTEGER		This is the question bank ID and a foreign key in the question paper table
Question type	VARCHAR		This is the type of question in the question paper table
Question	VARCHAR		This is the question for the question paper table
Marks allocated	INTEGER		This is the marks allocated for the question paper table
CLO ID	INTEGER		This is the CLO ID and a foreign key in the question paper table
Course ID	INTEGER		This is the course and a foreign key in the question paper table

CHAPTER-4 PHYSICAL SYSTEM DESIGN

A.INPUT FORMS:

SPMS 4.0
Dashboard
Manual Form
Import CO
Student performance
Student Enrollment
Logout

Student Enrollment Form

Student ID:

Educational year:

Educational semester:

Enrolled course:

Enrolled section:

Obtained grade:

Submit Query

```

<?php
include 'connect.php';

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $studentID = $_POST['studentID'];
    $edu_year = $_POST['edu_year'];
    $edu_sem = $_POST['edu_sem'];
    $enrolledCourse = $_POST['enrolledCourse'];
    $enrolledSection = $_POST['enrolledSection'];
    $Grade = $_POST['Grade'];

    $sql = "INSERT INTO course_outcome_t (studentID, edu_year, edu_sem, enrolledCourse, enrolledSection, Grade)
VALUES ('$studentID', '$edu_year', '$edu_sem', '$enrolledCourse', '$enrolledSection', '$Grade')";

    if ($con->query($sql) === TRUE) {
        echo "New record created successfully";
    } else {
        echo "Error: " . $sql . "<br>" . $con->error;
    }
}

$con->close();

```

?>

SPMS 4.0 [Dashboard](#) [Manual Form](#) [Import CO](#) [Student performance](#) [Student Enrollment](#) [Logout](#)

Import Course Outcome

Choose CSV File: No file selected.

Import

```
<?php
```

```
include 'connect.php';
```

```
if(isset($_POST["import"])){
```

```
    $fileName = $_FILES["file"]["tmp_name"];
```

```
    if($_FILES["file"]["size"] > 0){
```

```
        $file = fopen($fileName, "r");
```

```
        while(($column = fgetcsv($file, 10000, ",")) !== FALSE){
```

```
            // Check if the row contains valid data
```

```
            if(count($column) == 4 && !empty($column[0]) && !empty($column[1]) && !empty($column[2]) &&
!empty($column[3])){
```

```
                $sqlInsert = "Insert into co_t (coNum, courseID, ploID, poID) values ('" . $column[0] . "', '" . $column[1] . "', '" .
. $column[2] . "', '" . $column[3] . "')";
```

```
                $result = mysqli_query($con, $sqlInsert);
```

```
                if(!empty($result)){
```

```
                    echo "CSV imported into database";
```

```
                }
```

```

        else{
            echo "Problem importing CSV";
        }
    }
}
}
}
}
}
?>

```

SPMS 4.0 [Dashboard](#) [Manual Form](#) [Import CO](#) [Student performance](#) [Student Enrollment](#) [Logout](#)

Import Students performance

Choose CSV File: No file selected.

```

<?php

include 'connect.php';

if(isset($_POST["import"])){
    $fileName = $_FILES["file"]["tmp_name"];

    if($_FILES["file"]["size"] > 0){
        $file = fopen($fileName, "r");

        while(($column = fgetcsv($file, 10000, ",")) !== FALSE){
            // Check if the row contains valid data
            if(count($column) == 3 && !empty($column[0]) && !empty($column[1]) && !empty($column[2])){

```



```

        $sqlInsert = "Insert into student_course_performance_t (studentID, totalMarksObtained, gradePoint) values
('" . $column[0] . "', '" . $column[1] . "', '" . $column[2] . "')";

        $result = mysqli_query($con, $sqlInsert);

        if(!empty($result)){
            echo "CSV imported into database";
        }
        else{
            echo "Problem importing CSV";
        }
    }
}
}
}
}
?>

```

SPMS 4.0
Dashboard
Manual Form
Import CO
Student performance
Student Enrollment
Logout

Import Student Enrollment

Choose CSV File: No file selected.

Import

```

<?php

include 'connect.php';

if(isset($_POST["import"])){
    $fileName = $_FILES["file"]["tmp_name"];

```

```
if($_FILES['file']['size'] > 0){
    $file = fopen($fileName, "r");

    while(($column = fgetcsv($file, 10000, ",") != FALSE){
        // Check if the row contains valid data
        if(count($column) == 2 && !empty($column[0]) && !empty($column[1])){
            $sqlInsert = "Insert into registration_t (sectionID, studentID) values (" . $column[0] . ", " . $column[1] . ")";

            $result = mysqli_query($con, $sqlInsert);

            if(!empty($result)){
                echo "CSV imported into database";
            }
            else{
                echo "Problem importing CSV";
            }
        }
    }
}

?>
```

SPMS 4.0 [Dashboard](#) [Add Exam](#) [View Exam](#) [Logout](#)

Question Number	<input type="text"/>	Question Details	<input type="text"/>	Mark	<input type="text"/>	CO Number	<input type="text"/>
Question Number	<input type="text"/>	Question Details	<input type="text"/>	Mark	<input type="text"/>	CO Number	<input type="text"/>
Question Number	<input type="text"/>	Question Details	<input type="text"/>	Mark	<input type="text"/>	CO Number	<input type="text"/>
Question Number	<input type="text"/>	Question Details	<input type="text"/>	Mark	<input type="text"/>	CO Number	<input type="text"/>
Question Number	<input type="text"/>	Question Details	<input type="text"/>	Mark	<input type="text"/>	CO Number	<input type="text"/>
Question Number	<input type="text"/>	Question Details	<input type="text"/>	Mark	<input type="text"/>	CO Number	<input type="text"/>
Question Number	<input type="text"/>	Question Details	<input type="text"/>	Mark	<input type="text"/>	CO Number	<input type="text"/>

```

$examName=$_POST['examName'];
$sectionNum=$_POST['sectionNum'];
$questionCount=$_POST['questionCount'];
$courseID=$_POST['courseID'];
$semester=$_POST['semester'];
$year=$_POST['year'];

//Getting section ID from database
$result=mysqli_query($con,"SELECT sec.sectionID AS sectionID
FROM section_t AS sec
WHERE sec.sectionNum='$sectionNum' AND sec.courseID='$courseID'
AND sec.semester='$semester' AND sec.year='$year'");
$row=mysqli_fetch_assoc($result);
$sectionID=$row['sectionID'];

//storing exam in database
$query="INSERT INTO `exam_t` (`examID`, `examName`, `sectionID`)
VALUES (NULL, '$examName', '$sectionID')";

```

```

$result=mysqli_query($con,$query);

//getting the exam ID from database
$result=mysqli_query($con,"SELECT MAX(examID) AS examID
FROM exam_t");
$row=mysqli_fetch_assoc($result);
$examID=$row['examID'];

//Getting course ID from database
$result=mysqli_query($con,"SELECT sec.courseID AS courseID
FROM section_t AS sec
WHERE sec.sectionID='$sectionID'");
$row=mysqli_fetch_assoc($result);
$courseID=$row['courseID'];

//Storing questions in database
for($i=1;$i<=$questionCount;$i++){

    $difficultyLevel=0;

    $questionNum=$_POST["questionNum".$i];
    $questionDetails=$_POST["questionDetails".$i];
    $mark=$_POST["mark".$i];
    $coNum=$_POST["coNum".$i];

    if($difficultyLevel<=0){
        for($k=0;$k<sizeof($level1);$k++){
            if(strpos($questionDetails,$level1[$k])!==false){
                $difficultyLevel=1;
                break;
            }
        }
    }

    if($difficultyLevel<=0){
        for($k=0;$k<sizeof($level2);$k++){
            if(strpos($questionDetails,$level2[$k])!==false){
                $difficultyLevel=2;
            }
        }
    }
}

```

```

        break;
    }
}

if($difficultyLevel<=0){
for($k=0;$k<sizeof($level3);$k++){
    if(strpos($questionDetails,$level3[$k])!=false){
        $difficultyLevel=3;
        break;}
    }
}

if($difficultyLevel<=0){
for($k=0;$k<sizeof($level4);$k++){
    if(strpos($questionDetails,$level4[$k])!=false){
        $difficultyLevel=4;
        break;}
    }
}

if($difficultyLevel<=0){
for($k=0;$k<sizeof($level5);$k++){
    if(strpos($questionDetails,$level5[$k])!=false){
        $difficultyLevel=5;
        break;
    }
}
}

if($difficultyLevel<=0){
for($k=0;$k<sizeof($level6);$k++){
    if(strpos($questionDetails,$level6[$k])!=false){
        $difficultyLevel=6;
        break;
    }
}
}

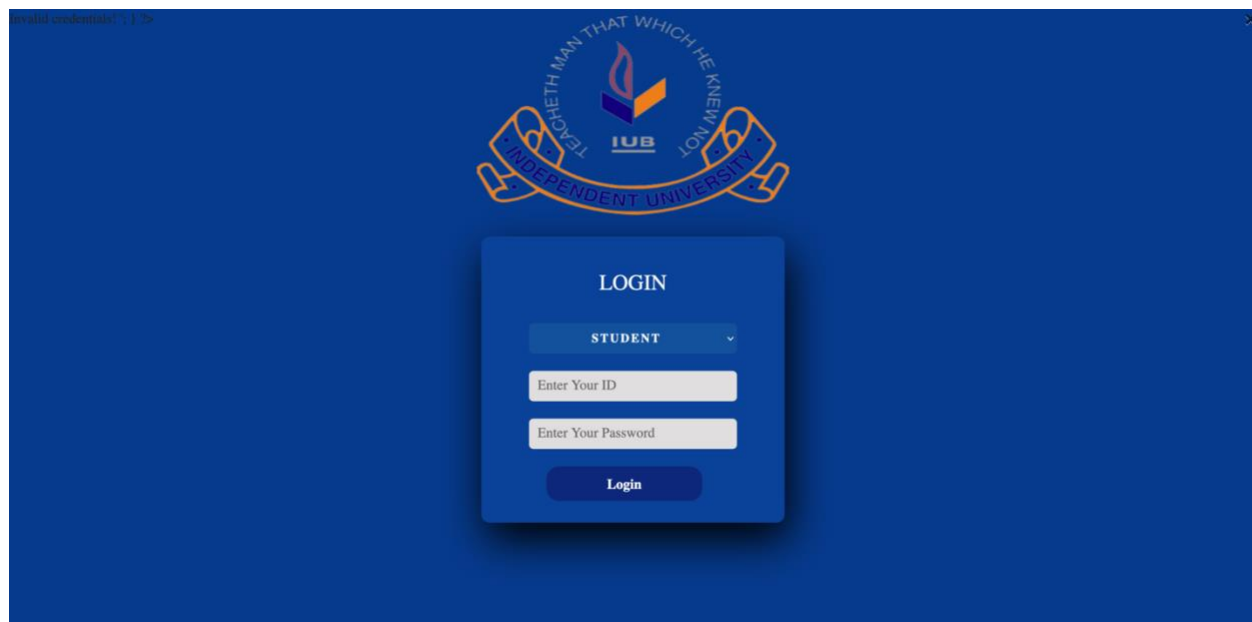
$query="insert into question_t (questionID,questionDetails,markPerQuestion,

```

```
questionNum,difficultyLevel,examID,courseID,coNum)
values('$questionDetails','$mark','$questionNum',
'$difficultyLevel','$examID','$courseID','$coNum');

$result=mysqli_query($con,$query);
}
header('location:addExam.php');
}
?>
```

B. OUTPUT FORMS:



```
<?php
```

```
$invalid=0;
```

```
if($_SERVER['REQUEST_METHOD']=='POST'){
```

```
    include 'connect.php';
```

```
    $userType=$_POST['userType'];
```

```
    $ID=$_POST['ID'];
```

```
    $password=$_POST['password'];
```

```
    if($userType!='student'){
```

```
        $sql="SELECT * from employee_t where employeeID='$ID' and password='$password'";
```

```
        $result=mysqli_query($con,$sql);
```

```
        if($result){
```

```
            $num=mysqli_num_rows($result);
```

```
            if($num>0){
```

```
                $invalid=0;
```

```
                session_start();
```

```
                $_SESSION['ID']=$ID;
```

```
        header('location:employee_dashboard.php');
    }
}

elseif($userType=='student'){
    $sql="SELECT * from student_t where studentID='$ID' and password='$password'";
    $result=mysqli_query($con,$sql);
    if($result){
        $num=mysqli_num_rows($result);
        if($num>0){
            $invalid=0;
            session_start();
            $_SESSION['ID']=$ID;
            header('location:employee_dashboard.php');
        }
    }
}

else{
    $invalid=1;
}

?>
```


SPMS 3.0 Dashboard PLO Analysis With Department/Program/School Average PLO Analysis (Overall, CO Wise, Course Wise) Logout

```
<?php
```

```
$sql="SELECT plo.ploNum AS ploNum,
AVG((ans.markObtained/q.markPerQuestion)*100) AS percent
FROM registration_t AS r, answer_t AS ans, question_t AS q,
co_t AS co, plo_t AS plo
WHERE r.registrationID=ans.registrationID
AND ans.examID=q.examID AND ans.answerNum=q.questionNum AND q.coNum=co.coNum
AND q.courseID=co.courseID AND co.ploID=plo.ploID
AND r.studentID='$studentID'
GROUP BY plo.ploNum,r.studentID";
```

```
$result=mysqli_query($con,$sql);
```

```
$sql2="SELECT plo.ploNum AS ploNum, AVG((ans.markObtained/q.markPerQuestion)*100)
AS percent
FROM registration_t AS r, answer_t AS ans, question_t AS q,
co_t AS co, plo_t AS plo, student_t AS s WHERE r.studentID=s.studentID
AND r.registrationID=ans.registrationID AND ans.examID=q.examID
AND ans.answerNum=q.questionNum
```

```

AND q.coNum=co.coNum AND q.courseID=co.courseID AND co.ploID=plo.ploID
AND s.departmentID=(SELECT s.departmentID FROM student_t AS s
WHERE s.studentID='$studentID')
GROUP BY plo.ploNum";

$result2=mysqli_query($con,$sql2);

?>

google.charts.load('current', {'packages':['bar']});
google.charts.setOnLoadCallback(drawAutumnChart);

function drawAutumnChart() {
    var data = google.visualization.arrayToDataTable([
        [ploNum,'Individual','Dept Average'],

<?php
        while($data=mysqli_fetch_array($result)){
            $data2=mysqli_fetch_array($result2);
            $ploNum="PLO".$data[ploNum];
            $percent=$data[percent];
            $percent2=$data2[percent];

            ?>

            ['<?php echo $ploNum;?>','<?php echo $percent;?>','<?php echo $percent2;?>'],

        <?php
        }
    ?>

```

SPMS 4.0
Dashboard
Exam
Create Course Outline
View Course Outline
Logout

COURSE

SECTION

SEMESTER

YEAR

SUBMIT

```
<?php
```

```
include 'connect.php';
```

```
if(isset($_POST['submit'])){
```

```
    $sectionNum=$_POST['sectionNum'];
```

```
    $courseID=$_POST['courseID'];
```

```
    $semester=$_POST['semester'];
```

```
    $year=$_POST['year'];
```

```
//Getting section ID from database
```

```
$result=mysqli_query($con,"SELECT sec.sectionID AS sectionID
```

```
FROM section_t AS sec
```

```
WHERE sec.sectionNum='$sectionNum' AND sec.courseID='$courseID'
```

```
AND sec.semester='$semester' AND sec.year='$year'");
```

```
$row=mysqli_fetch_assoc($result);
```

```
$_SESSION['sectionID']=$row['sectionID'];
```

```
header('location:createpdf.php');
```

```
}?>
```

Course Outline			
Course Code	Enter Course Code	Course Title	Enter Course Title
Section	Enter Section	Prerequisite(if any)	Enter Prerequisite
Credit Value	Enter Credit Value	Semester	Enter Semester

Course Description
Enter Course Description

Course Objective
Enter Course Objective

Course Policy
<p>a. It is the student's responsibility to gather information about the assignments/project and cover topics during the lectures missed. Regular class attendance is mandatory. Points will be taken off for missing classes. Without 70% of attendance, sitting for the final exam is NOT allowed. Students should come on time to get the attendance. In the event of failing 70% of attendance, a student will receive a W grade automatically.</p> <p>b. Same project work is assigned to all sections. Students should work in groups for the project. They are required to prepare a final report on the project which will be incrementally developed through assignments.</p> <p>c. The date and syllabus of class tests, Mid-Term and Final-Term will be announced in the class. There is NO provision for make-up.</p> <p>d. Both the Mid-Term and Final-Term exams will be coordinated exams and will be held on a specific date for all the sections.</p> <p>e. The reading materials for each class will be given prior to that class so that students may have a cursory look into the materials.</p> <p>f. Class participation is vital for a better understanding of the topics of this course. Students are invited to raise questions.</p> <p>g. Students should take tutorials with the instructor during office hours. Prior appointment is required.</p> <p>h. Students must maintain the IUB code of conduct and ethical guidelines offered by the school of computer science and engineering.</p> <p>i. No working mobile phones are allowed in class. Using one for any purpose will result in serious consequences.</p>

ACADEMIC DISHONESTY

- a. A student who cheats, plagiarizes, or furnishes false, misleading information in the course is subject to disciplinary action up to and including an F grade in the course and/or suspension/expulsion from the University.
- b. Students must maintain the code of IUB.
- c. The goal of homework is to give you practice in mastering the course material. Consequently, you are encouraged to collaborate on problem sets. In fact, students who form study groups generally do better on exams than do students who work alone. If you do work in a study group, however, you owe it to yourself and your group to be prepared for your study group meeting. Specifically, you should spend at least 30-45 minutes trying to solve each problem beforehand by yourself. If your group is unable to solve a problem, talk to other groups or ask your recitation instructor or teaching assistant assigned to your class.
- d. You must write up each problem solution by yourself without assistance. It is a violation of this policy to submit a problem solution that you cannot orally explain to a member of the course staff.
- e. No collaboration whatsoever is permitted during examination.
- f. Plagiarism and other anti-intellectual behavior cannot be tolerated in any academic environment that prides itself on individual accomplishment. If you have any questions about the collaboration policy, or if you feel that you may have violated the policy, please talk to one of the course staff. Although the course staff is obligated to deal with cheating appropriately, we are more understanding and lenient if we find out from the transgressor himself or herself rather than from a third party or by ourselves.

STUDENT WITH DISABILITIES AND STRESS

Students with disabilities are required to inform the Department of Computer Science & Engineering of any specific requirement for classes or examination as soon as possible. Additionally, if you experience significant stress or worry, changes in mood, or problems eating or sleeping this semester, whether because of this or any other courses or factors, please do not hesitate to reach out immediately, at any hour, to any of the course's heads to discuss.

NON DISCREMINATION POLICY

The course and University policy prohibit discrimination based on race, color, religion, sex, marital or parental status, national origin or ancestry, age, mental or physical disability, sexual orientation, military status. If you see either by course instructor or any other person related to course showing any form of discrimination, please inform the proctors office of the wrongdoing.

COURSE CONTENT

Enter Course Content

NO. OF CLOS

COURSE LEARNING OUTCOME (CLO) MATRIX

CLOs	CO Description	Blooms Taxonomy				PLO Assessed	CLO – PLO Correlation **
		C	P	A	S		
Enter Description	Enter Description	Enter	Enter	Enter	Enter	Enter PLO	Enter Correlation
Enter Description	Enter Description	Enter	Enter	Enter	Enter	Enter PLO	Enter Correlation
Enter Description	Enter Description	Enter	Enter	Enter	Enter	Enter PLO	Enter Correlation
Enter Description	Enter Description	Enter	Enter	Enter	Enter	Enter PLO	Enter Correlation
Enter Description	Enter Description	Enter	Enter	Enter	Enter	Enter PLO	Enter Correlation
Enter Description	Enter Description	Enter	Enter	Enter	Enter	Enter PLO	Enter Correlation
Enter Description	Enter Description	Enter	Enter	Enter	Enter	Enter PLO	Enter Correlation

*Bloom's Learning Level: Numbers signifies the Level of Bloom's skills.

**CLO – PLO Correlation: 3 – high, 2 – medium, 1- low

NO. OF TOPIC

Week	Topic	Teaching-Learning Strategy	Assessment Strategy	Corresponding CLOs
1	Enter	Enter	Enter	Enter
2	Enter	Enter	Enter	Enter
3	Enter	Enter	Enter	Enter
4	Enter	Enter	Enter	Enter
5	Enter	Enter	Enter	Enter
6	Enter	Enter	Enter	Enter
7	Enter	Enter	Enter	Enter
8	Enter	Enter	Enter	Enter
9	Enter	Enter	Enter	Enter
10	Enter	Enter	Enter	Enter
11	Enter	Enter	Enter	Enter
12	Enter	Enter	Enter	Enter
13	Enter	Enter	Enter	Enter
14	Enter	Enter	Enter	Enter

ASSESSMENT AND EVALUATION				
Assessment Type	Assessment Tools	Marks Distribution	Blooms Category	Sub Total
Enter Assessment Type	Enter	Enter	Enter	Enter
	Enter	Enter	Enter	
	Enter	Enter	Enter	
Enter Assessment Type	Enter	Enter	Enter	Enter
	Enter	Enter	Enter	
	Enter	Enter	Enter	
Enter Assessment Type	Enter	Enter	Enter	Enter
	Enter	Enter	Enter	
	Enter	Enter	Enter	
Enter Assessment Type	Enter	Enter	Enter	Enter
	Enter	Enter	Enter	
	Enter	Enter	Enter	
Enter Assessment Type	Enter	Enter	Enter	Enter
	Enter	Enter	Enter	
	Enter	Enter	Enter	
Enter Assessment Type	Enter	Enter	Enter	Enter
	Enter	Enter	Enter	
	Enter	Enter	Enter	
				Total : 100%

* Numbers are inclusive

A	A-	B+	B	B-	C+	C	C-	D+	D	F
90-100	85-89	80-84	75-79	70-74	65-69	60-64	55-59	50-54	45-49	0-44

Enter Reference Materials

Page-86

```

$courseCode=NULL;
$secNum=NULL;
$courseTitle=NULL;
$prerequisite=NULL;
$creditValue=NULL;
$semesterName=NULL;
$courseDescription=NULL;
$courseObjective=NULL;
$sectionID=NULL;
$refMaterials=NULL;
$topic=NULL;
$learningStrategy=NULL;
$assessmentStrategy=NULL;
$correspondingclo=NULL;
$queryResult=NULL;
$courseContent=NULL;


$courseCode=$_POST['courseCode'];
$secNum=$_POST['secNum'];
$courseTitle=$_POST['courseTitle'];
$prerequisite=$_POST['prerequisite'];
$creditValue=$_POST['creditValue'];
$semesterName=$_POST['semesterName'];
$courseDescription=$_POST['courseDescription'];
$courseObjective=$_POST['courseObjective'];
$sectionID=$_SESSION['sectionID'];
$refMaterials=$_POST['refMaterials'];
$courseContent=$_POST['courseContent'];


$queryResult=mysqli_query($con,"INSERT INTO `course_outline_t`
(`courseOutlineID`, `sectionID`, `semester`, `courseDescription`,
`objective`, `content`, `refMaterials`, `sectionNum`, `courseTitle`,
`prerequisiteCode`, `creditValue`)
VALUES (NULL, '$sectionID', '$semesterName', '$courseDescription', '$courseObjective',

```

```

'$courseContent', '$refMaterials', '$secNum', '$courseTitle',
'$prerequisite', '$creditValue');"");

//Getting course Outline ID from Database
$res=mysqli_query($con,"SELECT courseOutlineID
FROM course_outline_t AS cot
WHERE cot.sectionID='$sectionID'");
$row=mysqli_fetch_assoc($res);
$courseOutlineID=$row['courseOutlineID'];

for($i=1;$i<=$_POST['topicNum'];$i++){
$queryResult=NULL;
$topic=$_POST["topic".$i];
$learningStrategy=$_POST["learningStrategy".$i];
$assessmentStrategy=$_POST["assessmentStrategy".$i];
$correspondingclo=$_POST["correspondingclo".$i];

$queryResult=mysqli_query($con,"INSERT INTO `lesson_plan_strategy_t` (`IPSID`,
`week`, `topic`, `learningStrategy`, `assessmentStrategy`,
`correspondingClo`, `courseOutlineID`) VALUES (NULL, '$i', '$topic', '$learningStrategy',
'$assessmentStrategy', '$correspondingclo', '$courseOutlineID')");
}

$j=1;

for($i=1;$i<=$_POST['assessmentCount'];$i++){
$queryResult=NULL;
$assessmentType=$_POST["assessmentType".$i];
$subTotal=$_POST["subTotal".$i];

$assessmentTool1=$_POST["assessmentTool".$j];
$markDistribution1=$_POST["markDistribution".$j];
$bloomsCategory1=$_POST["bloomsCategory".$j];

$j++;

$assessmentTool2=$_POST["assessmentTool".$j];

```



```

$markDistribution2=$_POST["markDistribution".$j];
$bloomsCategory2=$_POST["bloomsCategory".$j];

$j++;

$assessmentTool3=$_POST["assessmentTool".$j];
$markDistribution3=$_POST["markDistribution".$j];
$bloomsCategory3=$_POST["bloomsCategory".$j];

$j++;

$queryResult=mysqli_query($con,"INSERT INTO `evaluation_strategy_t`
(`eSID`, `assessmentTool1`, `markDistribution1`, `bloomsCategory1`,
`courseOutlineID`, `assessmentTool2`, `assessmentTool3`,
`markDistribution2`, `markDistribution3`, `bloomsCategory2`,
`bloomsCategory3`, `assessmentType`, `subTotal`) VALUES
(NULL, '$assessmentTool1', '$markDistribution1', '$bloomsCategory1',
'$courseOutlineID', '$assessmentTool2', '$assessmentTool3',
'$markDistribution2', '$markDistribution3',
'$bloomsCategory2', '$bloomsCategory3', '$assessmentType', '$subTotal')");
}

```

```

for($i=1;$i<=$_POST['cloCount'];$i++){
    $queryResult=NULL;
    $cloNum=$_POST["cloNum".$i];
    $clo=$_POST["clo".$i];
    $c=$_POST["c".$i];
    $p=$_POST["p".$i];
    $a=$_POST["a".$i];
    $s=$_POST["s".$i];
    $plo=$_POST["plo".$i];
    $correlation=$_POST["correlation".$i];

```

```

$queryResult=mysqli_query($con,"INSERT INTO `clo_matrix_t`
(`clo_MatID`, `cloNum`, `coDescription`, `ploAssessed`,
`correlation`, `courseOutlineID`, `c`, `p`, `a`, `s`)
VALUES (NULL, '$cloNum', '$clo', '$plo', '$correlation', '$courseOutlineID', '$c', '$p', '$a', '$s')");

```

```

    }

    header('location:createCourseOutline.php');
}

?>

```

SPMS 4.0 Dashboard Data Entry GPA Analysis Logout

Course Outcome Percentage

Course	Average Grade	Course Percentage
ACN101	3.28	82%
CSE101	3.73	93.33%
CSE104	3.47	86.87%
CSE201	2.9	72.5%
CSE203	3.18	79.5%
CSE210	2.75	68.75%
CSE310	3	75%
CSE780	2.8	70%
EEE101	3.15	78.75%
EEE233	2.7	67.5%

```

<?php
include 'connect.php';

$sql = "SELECT enrolledCourse, AVG(Grade) as avgGrade FROM course_outcome_t GROUP BY enrolledCourse";
$result = $con->query($sql);
?>

<!doctype html>
<html lang="en">
<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

```

```

<!-- Bootstrap CSS -->
<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.4.1/dist/css/bootstrap.min.css"
integrity="sha384-Vkoo8x4CGsO3+Hhvx8T/Q5PaXtkKtu6ug5TOeNV6gBiFeWPGFN9MuhOf23Q9Ifjh"
crossorigin="anonymous">

<title>Couse Outcome Percentage</title>
<!--Google Font-->
<link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swap" rel="stylesheet">
<link rel="stylesheet" href="style.css">
<style>
  body{
    background-repeat:no-repeat;
    background-attachment:fixed;
    background-size:50% 80%;
    background-position:center;
    background-color:#d1e3ea;
  }
  h1{
    margin-top: 20px;
    justify-content: left;
    display: flex;
    margin-left: 450px;
    color: #0d287b;
    font-size: bold;
  }
  table {
border-collapse: collapse;
width: 100%;
}

th, td {
  text-align: center;
  padding: 10px;
}

tr:nth-child(even){background-color: #f2f2f2}

```

```

th {
  background-color: #6698FF;
  color: white;
}

</style>
<body>
<div class="nav">
  <input type="checkbox" id="nav-check">
  <div class="nav-header">
    <div class="nav-title">
      SPMS 4.0

    </div>
  </div>
  <div class="nav-btn">
    <label for="nav-check">
      <span></span>
      <span></span>
      <span></span>
    </label>
  </div>

  <div class="nav-links">
    <ul>
      <li><a href="employee_dashboard.php" target="_self">Dashboard</a></li>
      <li><a href="ploAnalysis.php" target="_self">PLO Analysis</a></li>
      <li><a href="ploAchieveStats.php" target="_self">PLO Achievement Stats</a></li>
      <li><a href="dataEntry.php" target="_self">Data Entry</a></li>
      <li><a href="course_percentage.php" target="_self">Course Outcomes</a></li>
      <li><a href="viewCourseOutline.php" target="_self">View course Outline</a></li>
      <li><a href="enrollmentStatistics.php" target="_self">Enrollment Stats</a></li>
      <li><a href="performanceStats.php" target="_self">GPA Analysis</a></li>
      <li><a href="logout.php" target="_self">Logout</a></li>
    </ul>
  </div>
</div>

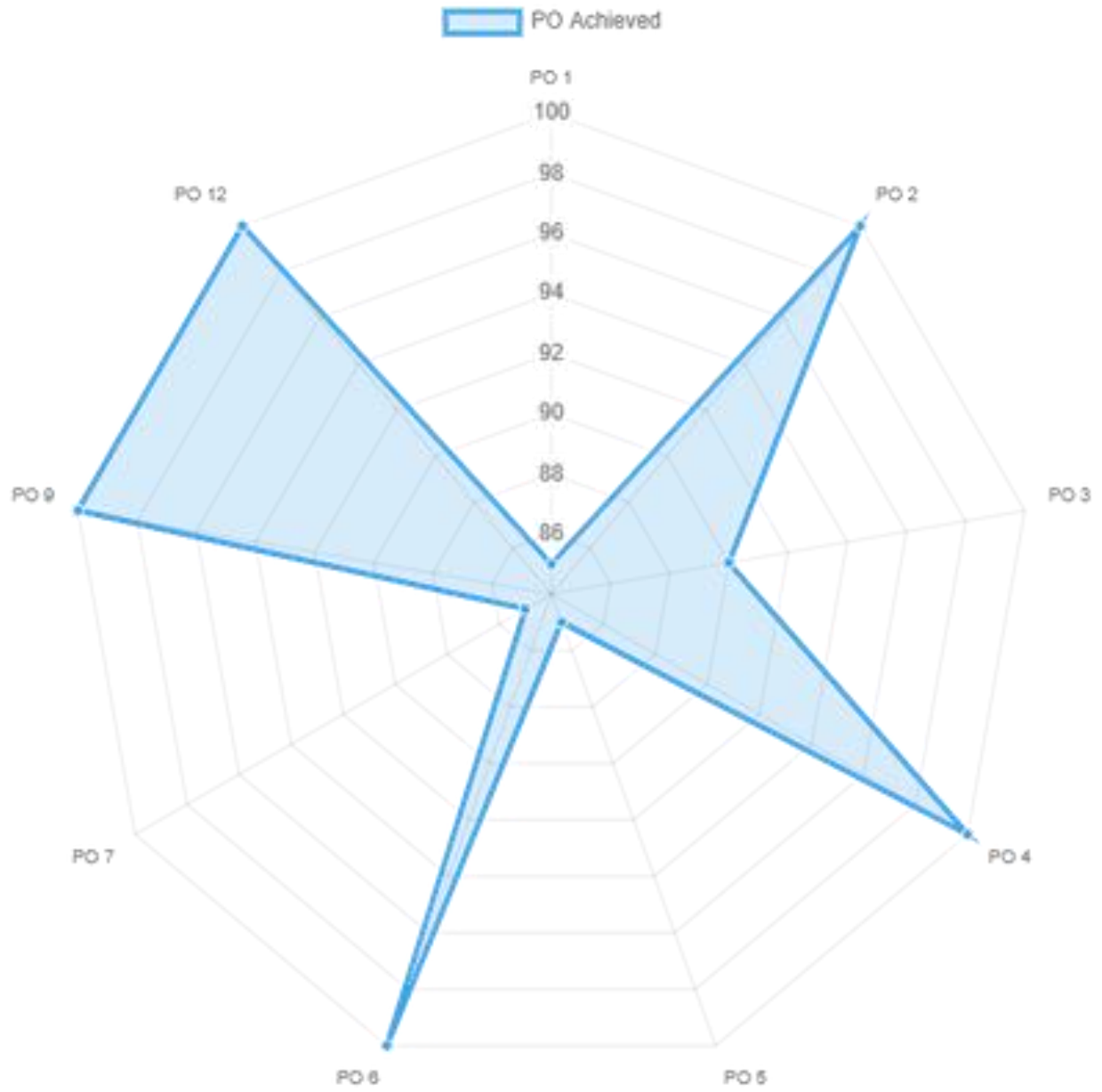
```

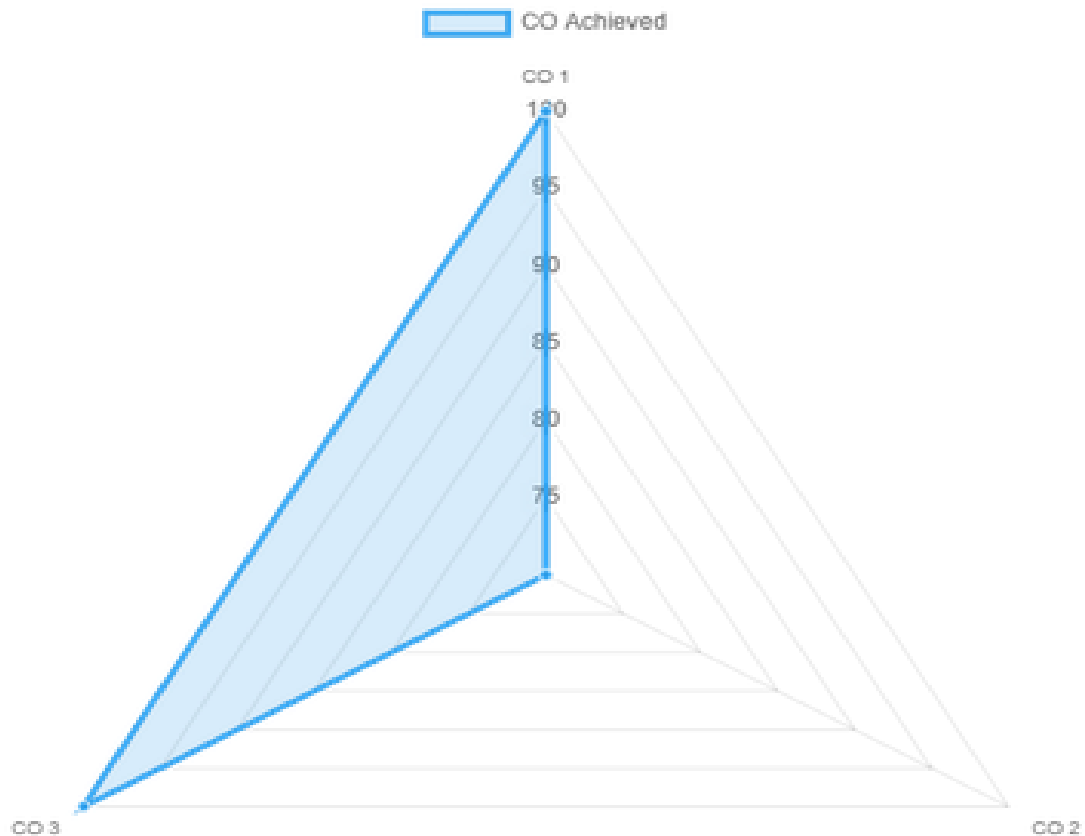
```

<b><h1>Course Outcome Percentage</h1></b>
<table class = "course">
  <tr>
    <th>Course</th>
    <th>Average Grade</th>
    <th>Course Percentage</th>
  </tr>
  <?php
  if ($result->num_rows > 0) {
    // output data of each row
    while($row = $result->fetch_assoc()) {
      echo "<tr>";
      echo "<td>" . $row["enrolledCourse"] . "</td>";
      echo "<td>" . round($row["avgGrade"], 2) . "</td>";
      echo "<td>" . round((( $row["avgGrade"]/4) * 100), 2) . "%</td>";
      echo "</tr>";
    }
  } else {
    echo "<tr><td colspan='2'>No data found</td></tr>";
  }
  ?>
</table>
</body>
</html>

<?php
$con->close();
?>

```





```

<?php
    if(isset($_POST['submit'])){
        $studentID=$_POST['studentID'];
    }?>

<script>

function poView(){
    <?php
        $sql="SELECT po.poNum AS poNum,
        AVG((ans.markObtained/q.markPerQuestion)*100) AS percent
        FROM registration_t AS r, answer_t AS ans, question_t AS q,

```

```

co_t AS co, po_t AS po
WHERE r.registrationID=ans.registrationID
AND ans.examID=q.examID
AND ans.answerNum=q.questionNum AND q.coNum=co.coNum
AND q.courseID=co.courseID AND co.polD=po.polD
AND r.studentID='$studentID'
GROUP BY po.poNum";

$result=mysqli_query($con,$sql);

$po=array();
$percent=array();

while($data=mysqli_fetch_array($result)){

    array_push($po,"PO ".$data['poNum']);
    array_push($percent,$data['percent']);

}

?>

var po=<?php echo json_encode($po); ?>;
var percent=<?php echo json_encode($percent); ?>;

for(var i=0;i<percent.length;i++){
    percent[i]=parseFloat(percent[i]);
}

const ctx = document.getElementById('myChart');

new Chart(ctx, {
    type: 'radar',
    data: {
        labels: po,
        datasets: [{

```



```

        label: 'PO Achieved',
        data: percent,
        fill: true,
        backgroundColor: 'rgba(54, 162, 235, 0.2)',
        borderColor: 'rgb(54, 162, 235)',
        pointBackgroundColor: 'rgb(54, 162, 235)',
        pointBorderColor: '#fff',
        pointHoverBackgroundColor: '#fff',
        pointHoverBorderColor: 'rgb(54, 162, 235)']
    },
    options: {
        elements: {
            line: {
                borderWidth: 3
            }
        }
    }
});

}

function coView(){
<?php
$sql="SELECT q.coNum,
AVG((ans.markObtained/q.markPerQuestion)*100) AS percent
FROM registration_t AS r, answer_t AS ans, question_t AS q,
co_t AS co, po_t AS po
WHERE r.registrationID=ans.registrationID
AND ans.examID=q.examID
AND ans.answerNum=q.questionNum AND q.coNum=co.coNum
AND r.studentID='$studentID'
GROUP BY q.coNum";

$result=mysqli_query($con,$sql);

$co=array();
$percent=array();

```

```

while($data=mysqli_fetch_array($result)){

    array_push($co,"CO ".$data['coNum']);
    array_push($percent,$data['percent']);

}

?>

var co=<?php echo json_encode($co); ?>;
var percent=<?php echo json_encode($percent); ?>;

for(var i=0;i<percent.length;i++){
    percent[i]=parseFloat(percent[i]);
}

const ctx = document.getElementById('myChart');

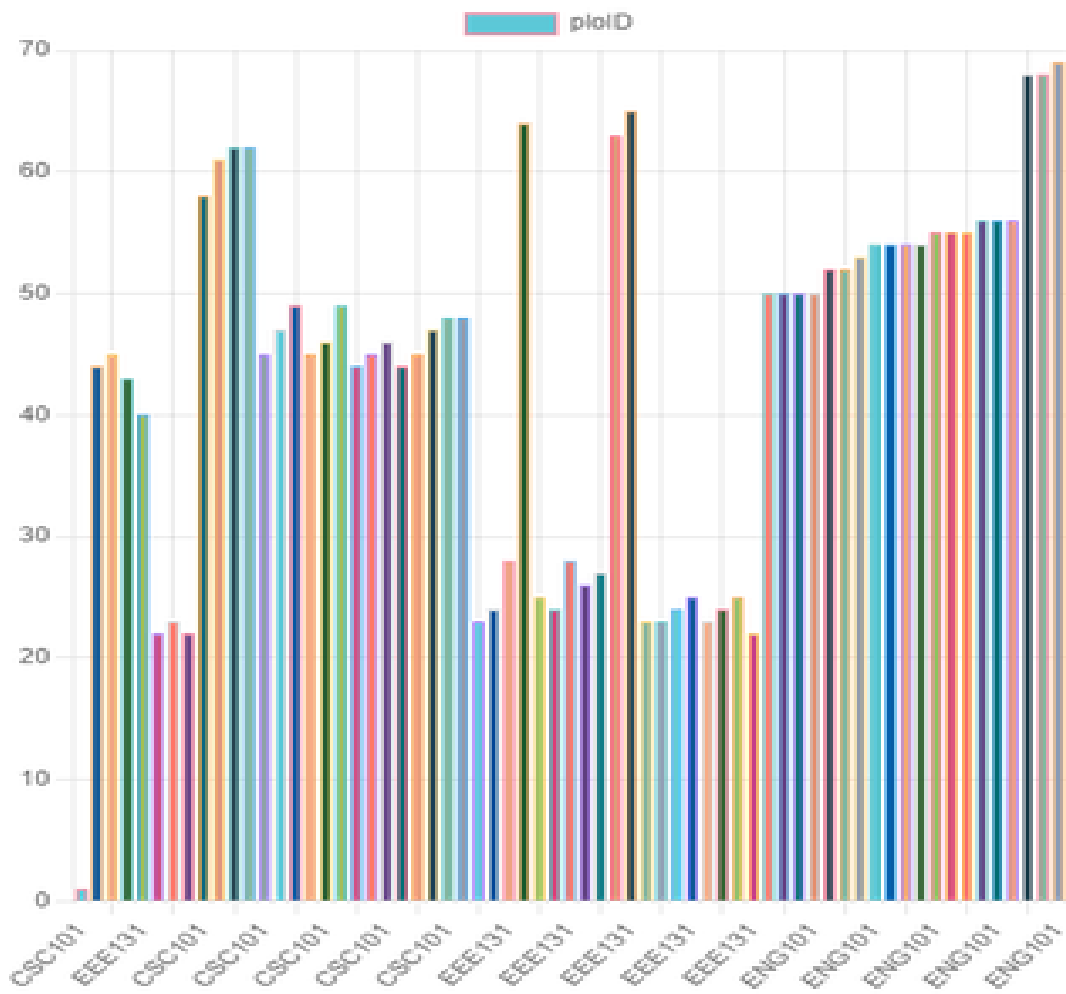
new Chart(ctx, {
    type: 'radar',
    data: {
        labels: co,
        datasets: [{
            label: 'CO Achieved',
            data: percent,
            fill: true,
            backgroundColor: 'rgba(54, 162, 235, 0.2)',
            borderColor: 'rgb(54, 162, 235)',
            pointBackgroundColor: 'rgb(54, 162, 235)',
            pointBorderColor: '#fff',
            pointHoverBackgroundColor: '#fff',
            pointHoverBorderColor: 'rgb(54, 162, 235)'}]
    },
    options: {
        elements: {

```

```

line: {
  borderWidth: 3
}
}
}
});
}

```



```
<?php
    $con = new mysqli('localhost','root','','hope');
    $query = $con->query("
    SELECT * FROM `co_t` WHERE 1
    ");
```

```

foreach($query as $data)
{
    $course[] = $data['courseID'];
    $gpa[] = $data['plolID'];
}

?>

<script>
// === include 'setup' then 'config' above ===
const labels = 'plolID';
const data = {
labels: <?php echo json_encode($course) ?>,
datasets: [{
    label: labels,
    data: <?php echo json_encode($gpa) ?>,
    backgroundColor: [
'#5CC8D7FF', '#00539CFF', '#EEA47FFF', '#2C5F2D', '#97BC62FF', '#CE4A7EFF', '#FC766AFF', '#603F83FF', '#006D77
',
    , '#E29578', '#264653', '#81b29a', '#8d99ae'
],
borderColor: [
    'rgb(255, 99, 132)',
    'rgb(255, 159, 64)',
    'rgb(255, 205, 86)',
    'rgb(75, 192, 192)',
    'rgb(54, 162, 235)',
    'rgb(153, 102, 255)',
    'rgb(201, 203, 207)'
],
borderWidth: 1,
hoverOffset: 4

```

```
    ]
  };

  const config = {
    type: 'bar',
    data: data,

    options: {
      scales: {
        xAxes: [{
          stacked: true,
          gridLines: {
            display: false,
          }
        }],
        yAxes: [{
          stacked: true,
          ticks: {
            beginAtZero: true,
            max: 100,
          },
          type: 'linear',
        }]
      },
      responsive: true,
      maintainAspectRatio: true
    }
  }

  var myChart = new Chart(
    document.getElementById('myChart'),
    config
  );
```

CHAPTER-5 CONCLUSION

A. PROBLEM AND SOLUTION:

Analysis Phase:

During the Analysis Phase, one of the major problems faced was the confusion around the Rich Picture and Six Element Analysis of the organizational operations since there was no data available regarding those operations. However, Faculty members and other stakeholders were interviewed in order to overcome such confusions, and information received during the interview was collected in order to get a better understanding of the system that was being developed.

Designing Phase:

Some problems were faced while creating the EERD and Relational Schema during the Design Phase, However, constant feedbacks from the faculty were enough to overcome those issues.

Implementation Phase:

All the System Requirements were completed successfully.

Front-End Developing tools: HTML, CSS, JavaScript, Google Charts, Chart JS

Back-End Developing tools: PHP, JSON

Database-integration: MySQL

Additional Features and Future Development:

One new feature could be added to this system in the near future which can monitor a student's extracurricular activities and then provide reports and analytics based on the student's extracurricular activity performance.