



Database Management Project

Final Report

(Group – 17)

Name	ID
Khademul Islam	1921717
Abu Sayed	2030155
Al Rawnak Shafin	2030543
Soumik Alam	2030479
Mahabuba Akter Thithi	2030462
Rasel Hossain	2030480

CHAPTER 1 - INTRODUCTION:

The Independent University, Bangladesh (IUB) has a number of strong and adaptable institutions, including the following:

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

The university has actively contributed to the development of the education industry in Bangladesh and has created competent and knowledgeable scholars who have made contributions both domestically and internationally. [1]

IUB has achieved this through working closely with relevant government education institutions and organizations such as the University Grants Commission (UGC), Ministry of Education, and other necessary institutes for each of the schools, regularly updating its curriculums and putting in a system to monitor student performance based on a quantified approach between course curriculum and standards set by UGC and the Bangladesh government and continuously monitoring student achievement for each semester, primarily using outcome-based education (OBE) also to set university curriculum.

The focus of this report is to study the current student performance monitoring system that IUB uses, 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 organization in issue and the project we undertook for it are both highlighted in the first section. In the second section, we introduce the proposed system that will take the position of the current system and discuss its flaws. The third and fourth sections will be highly technical and concentrate on how we intend to implement the suggested method.

During our research into the existing system for student performance monitoring we have found many areas where valuable changes could be made to make each process of monitoring student performance faster, make communication between necessary stakeholders easier, take away chances for errors and data duplication, and most importantly make it easier for all stakeholders to easily surf through large datasets to get meaningful information to their requirement. As we go through this report, we will dig deeper into how the current student performance monitoring system operates, the business

processes involved, where there are concerns and issues related to data management, and how we can make a better system to address these issues for fixing and improvement.

A. ORGANIZATION BACKGROUND- IUB:

One of the first private universities in Bangladesh, Independent University, Bangladesh (IUB), was founded in 1993. It is presently home to more than 7,048 undergraduate and graduate students as well as more than 10,455 alumni. Most experts project that this student population will increase by 10% a year.

IUB has produced graduates with useful skills in the long run, and this is only possible because of its strict adherence to the ongoing curriculum and progress system.

Concentrating efforts on the departments at IUB, and in particular, transforming the department of computer and electrical science into a well-funded research hub that manages a number of research initiatives. IUB is also dedicated to developing future international-standard graduates who are primarily prepared to lead the nation's economy in new directions through skilled employment, entrepreneurship, and/or applied research. IUB was able to build cutting-edge lab facilities in their department thanks to the strong backing of the Bangladeshi government and the UGC. It is due to IUB's "Application Oriented Learning" ideology, which views education as a "application," and teaches students "not only the fundamental principles of learning, situation-handling, and have better overall perception by providing them with hands-on training sessions," why this is the case.

Continuously growing it's 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 . To set the standard for incoming graduating engineers from top institutions in Bangladesh, IEB and pertinent government potentials set these procedures and criteria for credentials courses. These standards are presented as Program Educational Objectives (PEO) and Program Learning Outcomes (PLO) [1] for particular departments in an

accreditation manual. The appropriate course instructors and coordinators then map these PEOs and PLOs to the appropriate classes. This makes it possible for the Department of CSE at IUB, SECS, IEB, and all other pertinent stakeholders to calculate the present state of affairs and the performance of each student in each course for each semester. Users will be able to monitor the performance of the faculties, courses, departments, and schools, which offers important insight for making the required changes.

B. BACKGROUND OF THE PROJECT SPMS 4.0:

Measuring the production of students, faculties, departments, and their individual classes to gauge productivity in relation to the course activities' relevance to the outcomes. Basically, to offer a variety of tools and data to assist colleges, educational authorities like IEB, UGC, and other stakeholders in assessing student performance and informing improvement strategies. Creating a national framework for outcome-based education while giving colleges a lot of latitude in putting local strategies into practice.

C. OBJECTIVE OF THE PROJECT SPMS 4.0:

The SPMS 4.0 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 grades and showing the percentages of students as Program Learning Outcomes (PLO) or assessments such as 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. Additionally, SPSMS 4.0 tracks how well policies perform in comparison to the system's general administrative objectives. The system's primary goal is to use the database to track all actions at the university and to

generate analytics for the head of department, faculty, school, students, and their courses over a specified time period (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 2.0 (Student Performance Monitoring System 2.0), 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 2.0 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 realtime. 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, mark-sheets, 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:

The Requirement Analysis is a technique for investigating and visualizing the current system and business processes used by a specific company using tools, methods, and standards from the industry. "The process of figuring out the purpose of the database is known as requirements analysis. Interviews with user groups and other stakeholders are conducted to determine the functionality they need from the database, the types of data

they want to handle, and the routine operations. By doing this, we can observe how each party interacts with one another. We break down each business process in accordance with its components using straightforward notations and symbols that anyone can understand. As we will see, this process of analyzing lets us find out apparent and not so apparent problems with an existing system of monitoring student performance that is manual and depends on involving third party actors and stakeholders causing errors in the system.

A. RICH PICTURE – EXISTING BUSINESS SYSTEM:

A Rich Picture is a method for investigating, acknowledging, and defining a business process and then expressing it through diagrams to produce a rough conceptual model. A detailed description facilitates conversation and leads to a comprehensive knowledge that is shared by all parties. [5]The comprehensive picture that is produced can help other stakeholders understand the issues with a current system while also allowing them to take into account a wide range of relevant factors. Rich images focus on the processes and structure of a particular scenario.

The Rich Picture Analysis also takes in to account the following:

- Structures
- Processes
- Climate
- People
- Issues expressed by people.
- Conflict

As we can see, this rich picture was prepared keeping exactly those things in mind.

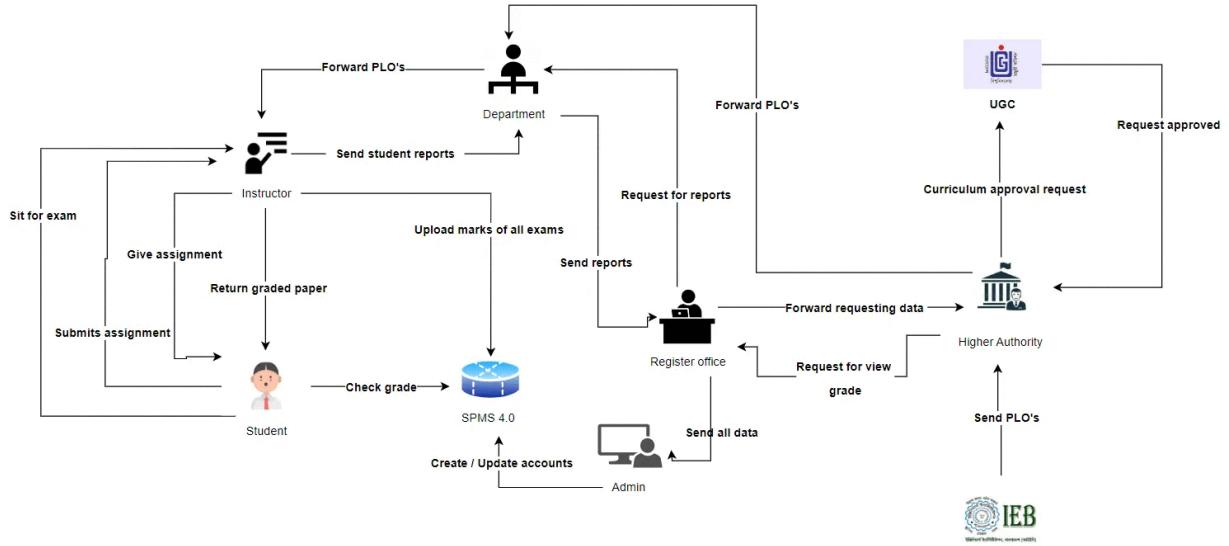


Figure 1.0: Rich Picture of Existing System to Monitor SPMS.

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

1. IEB/UGC
2. Head of Department/Dean of School
3. Department (working under Head of Department/Dean of School)
4. Course Instructor.
5. Registrar's Office
6. Admin (working under Registrar's Office)
7. Students

2.

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

1. The Department Storage
2. SPMS
3. From this Rich Picture we have drawn out 7 process that are key to monitoring student performance and improving curriculum. The processes are as follows:

1. Map Course Outcomes (COs) to Program Learning Outcomes (PLOs).
2. Record Student Assessment Data.
3. Produce OBE Marksheets & Course Assessment Report.

4. Create student/faculty account and enter/customize necessary data.
5. Request for review and change of grades.
6. View grading percentage.

B. SIX ELEMENTS ANALYSIS - EXISTING BUSINESS SYSTEM:

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 document 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	System Roles
---------	--------------

	Human	Non-Comp Hardware	Computing Hardware	Software	Database	Network & Communication
Mapping of COs(Course Outcomes) to PLOs(Program Learning Outcomes)	<p>IEB/UGC:</p> <ul style="list-style-type: none"> 1. Give the department heads a copy of the accreditation manual with the PLOs defined. 2. Department: <ul style="list-style-type: none"> 1. Receive Accreditation Manual from IEB. 2. Forward PLOs to Instructor . 3. Request for reports to the Register office and get student reports from the Instructor . 3. Instructor : <ul style="list-style-type: none"> 1. Get PLOs from the 	<p>Pen and Paper:</p> <ul style="list-style-type: none"> 1. Is utilized to record ideas from intermediate brainstorming sessions. <p>Board and marker:</p> <ul style="list-style-type: none"> 1. Is utilized to record ideas from intermediate brainstorming sessions. 	<p>Computer:</p> <ul style="list-style-type: none"> 1. Course coordinators create softcopies of the Course Outcomes (COs) of the particular courses in which they specialize using computers. <p>Printer:</p> <ul style="list-style-type: none"> 1. To print out hardcopies of Course Outcomes (COs). 	<p>MS Excel:</p> <ul style="list-style-type: none"> 1. Course coordinators utilize an Excel sheet to link certain questions on the midterm, final exam, and project work to certain course goals (COs). <p>MS Word:</p> <ul style="list-style-type: none"> 1. Course coordinators create thorough course outlines, Course Assessment Reports, and grade reports using Microsoft Word, linking Course Outcomes (COs) to Program 		<p>Internet & Email:</p> <ul style="list-style-type: none"> 1. Use the internet and emails to communicate with UGC/IEB. <p>Phone:</p> <ul style="list-style-type: none"> 1. Use phones to discuss important topics related to mapping CO to PLO.

	Department. 2. Prepare test questions in accordance with the mapped COs.			Learning Objectives (PLOs).		
Record Student Assessment Data	Instructor : 1. Assign assignments and project tasks. 2. Complete tests and quizzes during the semester. 3. Keep a	Pen and Paper: 1. Write down assessment information and grades on paper using a pen and a tabular format.	Computer: 1. Creating softcopies of records of all data for specific courses are done on Computers. final grades on Excel sheets.	MS Excel: 1. Enter the essential evaluation information and final grades on Excel sheets.	Database server: 1. IRAS stores and maintains student grade information on a database server.	Internet: 1. The Internet is used to communicate with IRAS to Store final grades of students.

	<p>record of each student's evaluation information for each semester.</p> <p>4. Determine the combined scores for all quizzes, homework assignments, midterm and final examinations, and award final grades to each student in the relevant courses.</p> <p>5. Enter final grades for students on IRAS.</p> <p>6. Give the Department the Marksheets.</p>			<p>IRAS:</p> <p>1. Upload students' final grades to IRAS for viewing by students or the Registrar's office.</p>		
View grades and download Transcripts	<p>Student:</p> <ol style="list-style-type: none"> Log into IRAS. Find results for the intended 	<p>Pen and Paper:</p> <ol style="list-style-type: none"> We can print tabulated transcripts onto 	<p>Computer/Device:</p> <ol style="list-style-type: none"> Used for IRAS access. <p>Printer:</p> <ol style="list-style-type: none"> Used for 		<p>Database Server:</p> <ol style="list-style-type: none"> The information on student grades in 	<p>Internet/Email:</p> <ol style="list-style-type: none"> The Internet is used to communicate

	<p>semester by semester.</p> <p>3. Examine the grades for semesters .</p> <p>4. Download the transcript to our hard drive using our browser.</p> <p>Registrar's Office:</p> <ul style="list-style-type: none"> 1. Access IRAS. 2. View students' grades if and when it is necessary. 3. Download them transcript s. 	<p>paper. The primary source of information for applications and other papers is hard copy.</p>	<p>printing the tabulated transcript.</p>		<p>IRAS is stored, maintained, edited, and received by a database management service.</p> <p>Web Server:</p> <ul style="list-style-type: none"> 1. User interface and website pages are served using a remote web server. 	<p>with IRAS to store final grades of students.</p>
Create / Update Account and Data	<p>Admin:</p> <ul style="list-style-type: none"> 1. Registration procedure s allow for the collection of new students' personal data. 2. Create a faculty and student account. 	<p>Pen and Paper:</p> <ul style="list-style-type: none"> 1. Used for writing/ copying student or faculty's information for account creation. 	<p>Computer:</p> <ul style="list-style-type: none"> 1. Used for accessing & updating data to IRAS. 		<p>Database Server:</p> <ul style="list-style-type: none"> 1. The student and faculty data in IRAS is stored, maintained, edited, and received via a database manage 	<p>Internet:</p> <ul style="list-style-type: none"> 1. In order to interface with IRAS and save account information on a remote database server, the internet is required. 2. Internet connectivity is used to offer

	3. Update some account details when necessary, for students or faculty.				ment service. Web Server: 1. A remote web server is used to offer user interface and website pages.	website pages and user interfaces.
Request for review and change of grades.	Students: 1. Request for grade change and review to faculty. Instructor : 1. Upon request, examine test results and other evaluations. 2. If a modification has to be made, send an admin grade change request for a specific student; otherwise , the procedure should be	Pen and Paper: 1. Used to note down key points or marks on the students' answer sheets.	Computer/ Phone: 1. Used for communication with the faculty.		Database server: 1. Update student grade data. Department Storage: 1. Update student grade data.	Internet: The main purpose of email is communication. Phone: 1. Can be used for communication.

	<p>ended.</p> <p>Admin:</p> <ol style="list-style-type: none"> 1. Get a request to modify a student's grade. 2. Adjust student's grade in response to faculty request. 					
--	---	--	--	--	--	--

C. PROBLEM ANALYSIS – EXISTING BUSINESS SYSTEM:

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
Assessments And Grading	1. Faculty 2. Student	1. Answer Script and question paper condition 2. Giving and Receiving Process 3. Unreliable Storage	1. The answer scripts and test papers that are kept in physical storage might become damaged or disappear. 2. The Process of completing the assessment and giving it to	The Answer Scripts and question paper are stored into the database so there is no problem in physical storage space or condition of the paper. Once a question is placed inside the question bank, the

			<p>the teacher in person is slow.</p> <p>3. Physical Storage space may become full very quickly as more answer scripts are taken.</p> <p>4. Need to find the domain of learning and difficulty of the question manually and that also takes a lot of time.</p>	<p>question gets its difficulty level and domain of learning automatically assigned according to the bloom's taxonomy.</p> <p>Online submission of assessment saves time as it negates the necessity to submit a physical copy in person.</p>
--	--	--	--	---

D. RICH PICTURE – PROPOSED SYSTEM:

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.

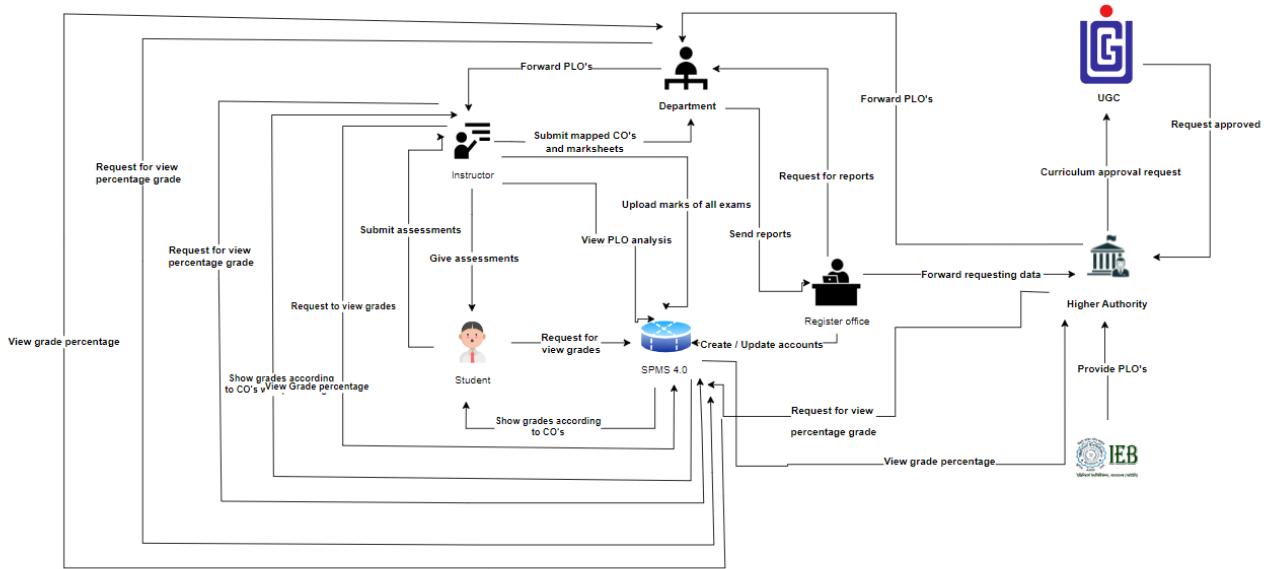


Figure 1.0: Rich Picture of Proposed System to Monitor SPMS.

In this rich picture the stakeholders are:

- 1) UGC
- 2) IEB
- 3) Higher Authority (VC, Dean etc)
- 4) Department Head
- 5) Register Office
- 6) SPMSV4.0 Admin (SPMS Manager) SPMS
- 7) Faculty
- 8) Student

E. SIX ELEMENTS ANALYSIS – PROPOSED SYSTEM :

The six elements analysis of the proposed system is a continuation of an analysis process where each analysis is based on the one that comes before it. Based on the rich picture, the role of each element in the new system is further understood in the table below.

Process	System Roles					
	Human	Non-Comp Hardware	Computing Hardware	Software	Database	Network & Communication
Set Assessment's according to CO's	<p>Faculty Member</p> <p>a) Set question papers for examinations according to mapped CO's.</p> <p>b) Set classroom, time and date of examinations.</p> <p>c) Invigilate examinations and collect test papers.</p> <p>Student</p> <p>a) Sit for examinations and submit attempted test papers to faculty.</p>	<p>Table and Chair</p> <p>a) To use during examinations. Pen and Paper</p> <p>b) For attempting the examinations.</p> <p>c) Questions may be printed on paper.</p> <p>Clock</p> <p>d) Setting time for the examination.</p> <p>Room</p> <p>a) Designated room for examination.</p>	<p>Computer</p> <p>a) Used by faculties to access the CO's from the Excel sheet.</p> <p>b) Faculties may also use it to take online examinations and interact with students.</p> <p>c) Students may use to attend online examinations by using their mobile phones or laptop. Scanning and uploading pdfs to virtual examinations by using their electronics devices.</p> <p>Printer</p> <p>a) Used by faculties to print out question papers for students.</p> <p>Networking</p>	<p>Microsoft Office</p> <p>a) The software from which the faculty will collect CO's.</p> <p>Google Classroom</p> <p>a) Used by faculties and students during examinations.</p> <p>Operating System</p> <p>a) Any OS used by the users, e.g. Windows, Mac.</p> <p>Printing Software</p> <p>a) Printing software used for printing the question paper.</p> <p>PDF viewer</p> <p>a) To view questions in PDF or send the answer in</p>	<p>Microso ft</p> <p>Excel</p> <p>Databas e</p> <p>a) Faculty access CO's from this.</p>	<p>Internet</p> <p>a) Used by faculties to access the excel file via email.</p> <p>b) Used by faculties and students during examinations.</p>

			devices (Router, Switch, Bridge, Hub) a) Used by faculties and students to access the Internet.	PDF.		
UGC approves Curriculu m based on PLO and CO's	Higher Authority a) Sends the Curriculu m booklet to UGC. b) If it gets approved by the UGC then the Higher Authority publishes the curriculum booklet. c) If it doesn't get approved the Higher Authority sets the curriculum according to the demands of the UGC. d) Authority sends the updated curriculum to the departme	Paper a) It is used to print the booklet.	Computer a) Used by the Higher Authority to send the curriculum by mail to the UGC. Also used for editing and updating the curriculum booklet doc file. Printer a) Used by the Higher Authority to print the curriculum booklet. Networking devices (Router, Internet Cable by ISP Providers) a) Used by Authority/U GC to access the internet.	Microsoft Office a) Used to edit or update the Curriculu m file. Gmail a) Used to send mail to the UGC/High er Authority. Operating System a) Any type of OS used by the users. e.g. Windows, Linux. Adobe Acrobat a) Used to view the PDF files. Printing Software a) Used for printing the	Microso ft Excel Files a) Higher Authorit y access the data to edit or update the curriculu m.	Internet Connection a) It is used by the Higher Authority/UGC to send or receive mail.

	<p>nt. UGC: a) Receives the curriculum booklet from the Higher Authority. b) Reviews the booklet if it requires changes it sends back feedback to the Higher Authority regarding the changes as needed else it is approved by the UGC.</p>			curriculum doc.		
Generatin g Student Transcript s	<p>Student a) Requests informatio n regarding one or more courses from Register office. b) Provides register office with necessary info such as ID/Course ID etc. Register</p>	<p>Paper a) The student may have to fill out a form with necessary information (Student ID, CourseIDs, etc.). b) Their course results/transcri pt information may be printed into a report.</p> <p>Pen a) The student may have to fill out a form with necessary information (Student ID/CourseID's</p>	<p>Computer a) To access the internet and Excel files that hold information about student marks for a course. b) Students may request information via email. Printer</p> <p>a) To print out forms the student will have to fill out</p>	<p>Microsoft Office a) The data is stored in MS Excel files.</p> <p>Operating System a) Any OS used by the users, e.g. Windows, Mac, Linux etc.</p> <p>Printing Software a) Printing software used</p>	<p>Microso ft Excel File System a) The marks for each course separate d. by CO's and PLO's are kept in excel files.</p>	<p>Internet Connection a) Used to access MS Excel files. b) Used to request information from Register office. c) Used to send reports to students.</p>

	office a) Receives requests for Student transcript information. b) Asks for Student's identifier information and list of courses. c) Goes through Excel File to collect the necessary information d) Compiles necessary information into a file and passes it to student.	etc).	to give required information . b) To print out transcript/g pa Networking devices (Router, Switch, Bridge, Hub) a) Used by register office and students to access the internet.	for printing the forms and transcripts . PDF viewer b) To view the transcript in PDF form.		
Mapping of CO's from PLO's	Faculty Member a) Maps the CO's from PLOs based on the syllabus covered in the course. b) Sends the mapped CO's to the register	Paper a) Used if the faculty member or the register wishes to print out the mapped CO's.	Computer a) Used to edit the CO's Excel file. Printer a) Used to print out the CO's for hardcopy storage backup in case something happens to	Microsoft Excel a) Used to store the mapped CO's. Web Browser a) To send and receive the CO's through email.	Microso ft Excel File System a) Contains the mapped CO's. Hardcop y storage a) Contains the	Internet a) Used to send the emails containing CO's.

	<p>office through email.</p> <p>Register office</p> <p>a) Receives the mapped CO's from the faculty member.</p> <p>b) Updates it in the excel file containing the CO's.</p>		the digital version.		hardcopy version of the CO's' Excel file for backup.	
Teachers evaluate students and submit CO's to Register office to update in Excel Database	<p>Faculty Member</p> <p>a) Mark exam papers and list CO's for each student.</p> <p>b) Submit CO's to register office</p> <p>c) Send the exam paper back to student.</p> <p> </p> <p>Register office</p> <p>a) Update CO's to Excel Database.</p> <p>Student</p> <p>a) Request the faculty to receive the exam paper.</p>	<p>Pen and Paper</p> <p>a) Used by faculty to mark exam papers and list CO's.</p> <p> </p> <p>Printer</p> <p>a) Used by Faculty to print out physical copies of CO's.</p>	<p>Computer</p> <p>a) Used by the faculty to store the exam result.</p> <p>b) Used to send CO's to register office via email.</p> <p>c) Used by Register office to update CO's in excel database.</p> <p> </p> <p>Networking devices (Router,</p>	<p>Microsoft Office</p> <p>a) Register office stores CO's in MS Excel files.</p> <p>b) Faculty stores student's CO's.</p> <p> </p> <p>Operating System</p> <p>a) Any OS used by the users, windows, mac.</p> <p> </p> <p>PDF viewer</p> <p>a) To view questions and mark them digitally in PDF gmail.</p> <p>a) Used by faculty to send CO's</p>	<p>Microso ft Excel Databas e</p> <p>a) Used by register office to update CO's.</p> <p> </p> <p>Internet</p> <p>a) Used by faculties to send CO's to register office via email.</p> <p>b) Used by Register office to update CO's in excel database.</p>	

			Internet Cable by ISP Providers a) Used by faculty and register office to access the Internet.	to register office for update.		
Higher Authority collects PLOs	IEB a) Provides PLO. Faculty a) Retrieve PLO from Higher Authority. Higher Authority a) Collects PLO from IEB.	Book a) Details containing the PLO. Pen and Paper a) For collecting and storing all the PLO.	Computer a) Higher Authority will access the collected PLO which will be given by the IEB. Printer a) IEB can use printers to print the required documents for PLO. b) Faculties can use it to request for PLO from Higher Authority.	Microsoft Excel / Google Sheets a) Used to create the PLO. b) Used to save the retrieved PLO. Email Software a) Used for communication between Higher Authority, faculties and IEB.	Marks Database a) For storing the mapped CO. b) For storing the PLO. IEB Database a) Retrieving the PLO details from IEB.	Internet a) Used by faculties to access IEB website. b) Used by higher authority to store and update the PLO. c) Used by IEB to update and store PLO in their database.

			Authority. c) Higher Authority can use it to print the retrieved PLO.			
SMPS 4.0	Internet a) Used by faculties and students to view progress report of students.	Computer a) Used by the faculties and students to show grades according to CO's with percentages.	Students a) Requests information regarding grades and can show their CO's.	Faculty a) Used if the faculty member needs to view student's CO's.		

CHAPTER-3 LOGICAL SYSTEM DESIGN:

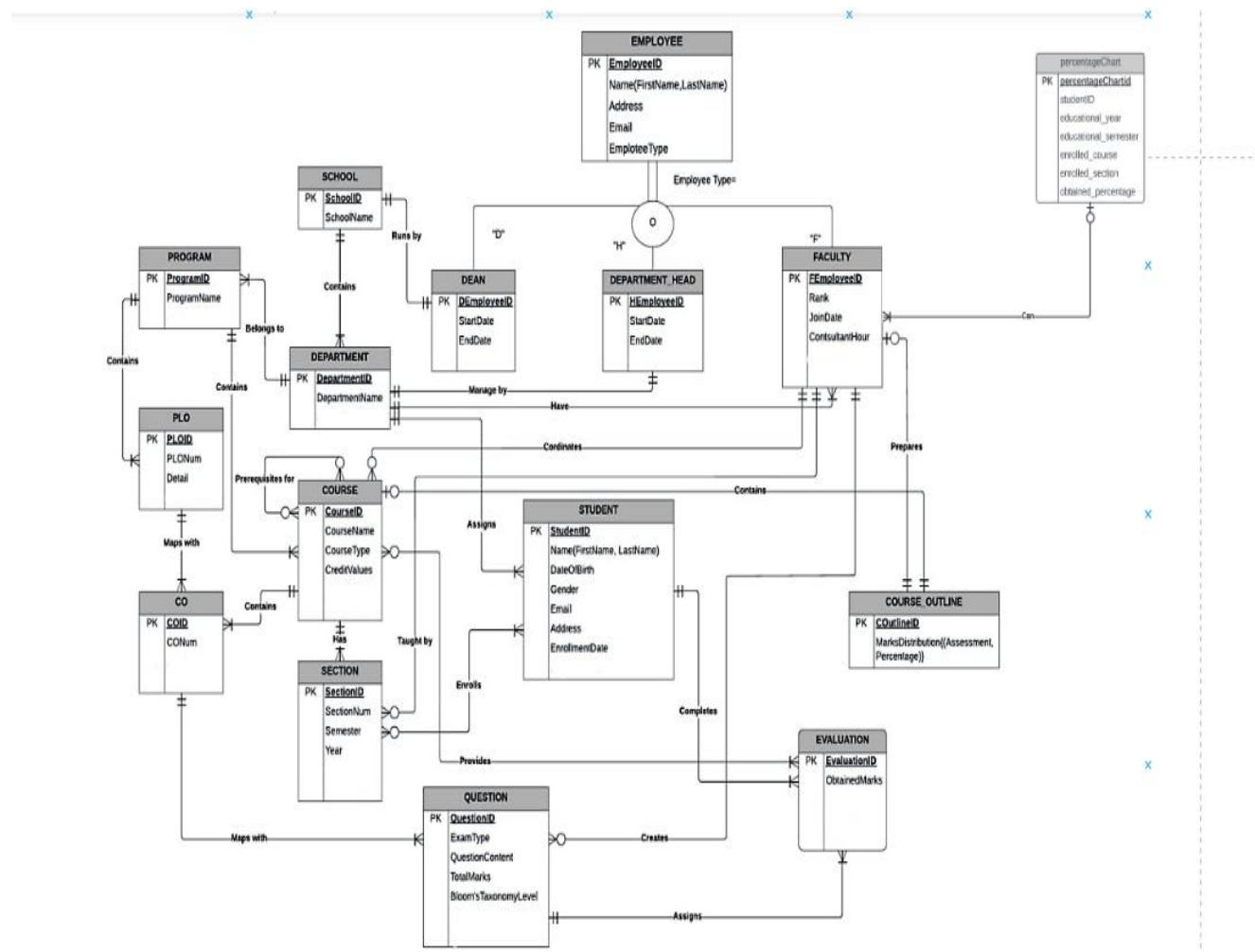
A. BUSINESS RULES – SPMS 4.0:

1. A student must have one department. A STUDENT has StudentID, FirstName, LastName, DateofBirth, Gender, Email, Phone, Address, EnrollmentDate. A department must have many students.
2. A section mandatorily has many students. A student may enroll in many sections. A section includes SectionID, SectionNum, CourseID, FEmployeeID, Semester, Year.
3. Students may complete many evaluations. An EVALUATION includes EvaluationID, ObtainedMarks, StudentID, CourseID, QuestionID. An evaluation must be performed by at least one student.
4. An evaluation must have one Question. A Question must have many evaluations. Question assigns QuestionID, ExamType, TotalMarks, QuestionContent, Bloom'sTaxonomyLevel, FEmployeeID, COID. A question must create one faculty. A faculty creates many questions.

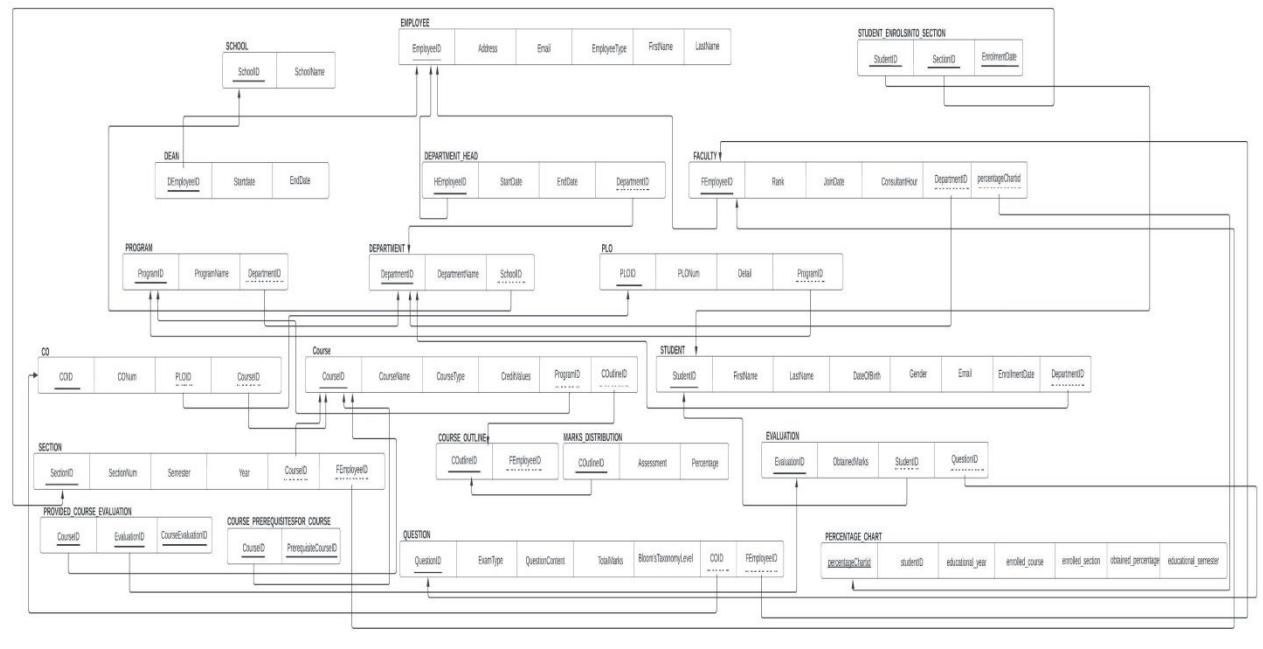
5. A CO's must map with one PLO's. A PLO's must map with one or many CO's. PLO includes PLOID, PLONum, Details, ProgramID.
6. A PLO must contain one program. A program contains one or many PLO's. A program has ProgramID, ProgramName, DepartmentID. A program must contain one or many courses. A Course must contain one program.
7. A program must belong to one department. A department must belong to one or many programs. A department contains DepartmentID, DepartmentName, SchoolID.
8. A department must contain one school. A School must contain one or many departments. A school includes SchoolID, SchoolName.
9. An employee has three sub-types (Dean, Department Head, Faculty). An employee includes EmployeeID, FirstName, LastName, Email, Address, EmployeeType.
10. A school must be run by one or many Deans. A dean must run one school. A Dean has SchoolID, StartDate, EndDate.
11. A Department must manage one or many Department head. A department head must manage one department. A department head includes DepartmentID, StartDate, EndDate.
12. A Faculty must have one Department. A department must have one or many Faculties. A Faculty includes FEmployeeID, DepartmentID, COutlineID, Rank, JoinDate, ConsultantHour. A faculty may teach many sections. A section must be taught by one faculty.
13. A question must map with one CO's. A CO's maps with one or many questions. A CO's includes CONum, COID, CourseID, PLOID. And questions include QuestionNum, ExamType, TotalMarks, Bloom'sTaxonomyCategory, Bloom'sTaxonomyLevel. A CO must contain one Course. A Course contains one or many CO's.
14. A course may have many prerequisites. A course includes CourseID, CourseName, CourseType, CreditValues, ProgramID, COutlineID. A course must contain one course outline. A course outline may be one course.
15. A course outline must affiliate one mark distribution. A mark distribution may affiliate many courses. A course outline includes MarkDistribution. Mark distribution represents multi valued course assessment and percentage exam type wise (quiz, midterm, final, project) and CoursePolicy. Faculty must prepare one course outline. A course outline may prepare one faculty.
16. The percentage chart has to be accessed by one or many faculty, where percentageChartID is shown as a primary key. Here studentID, educational_year,

educational_semester, enrolled_course, enrolled_section, obtained_percentage have been shown as attributes.

B. ENTITY RELATIONSHIP DIAGRAM:



C. ENTITY RELATIONSHIP DIAGRAM TO RELATIONAL SCHEMA:



D. NORMALIZATION:

SchoolID S1	SchoolName S2
DepartmentID D1	DepartmentName, SchoolID D2, S1
ProgramID P1	ProgramName, DepartmentID P2, D1
PLOID L1	PLONum, Detail, ProgramID L2, L3, P1
COID C1	CNum, PLOID C2, L1

CourseID A1	CourseName, CourseType, CreditValues, ProgramID A2, A3, A4, P1
SectionID G1	SectionNum, Semester, Year, CourseID, FEmployeeID G2, G3, G4, A1, F1
StudentID H1	Firstname, LastName, DateOfBirth, Gender, Email, address, EnrollmentDate, DepartmentID H2, H3, H4, H5, H6, H7, H8, D1
QuestionID Q1	ExamType, QuestionContent, TotalMarks, Category, Level, COID, FEmployeeID Q2, Q3, Q4, Q5, Q6, Q7, F1
EmployeeID E1	FirstName, LastName, Address, Email E2, E3, E4, E5
DEmployeeID J1	StartDate, EndDate, SchoolID J2, J3, S1
HEmployeeID K1	StartDate, EndDate, DepartmentID K2, K3, D1
FEmployeeID F1	Join date, Rank, ConsultantHour, DepartmentID, COutlineID F2, F3, F4, D1, O1
COOutlineID O1	Assessment, Percentage O2, O3
EvalutionID N1	ObtainedMarks, StudentID, CourseID, QuestionID N2, H1, A1, Q1
PercentageChartID Z1	Student_ID, Educational_year, Educational_semester, H1 Z2 Z3 Enrolled_course, Enrolled_section, Obtained_percentage Z4 Z5 Z6

S1 > S2

D1 > D2, S1

P1 > P2, D1

L1 > L2, L3, P1

C1 > C2, L1

A1 > A2, A3, A4, P1

G1 > G2, G3, G4, A1, F1

H1 > H2, H3, H4, H5, H6, H7, H8, D1, Z1

Q1 > Q2, Q3, Q4, Q5, Q6, Q7, F1

E1 > E2, E3, E4, E5

J1 > J2, J3, S1

K1 > K2, K3, D1

F1 > F2, F3, F4, D1, O1

O1 > O2, O3

N1 > N2, H1, A1, Q1

Z1 > Z2, Z3, Z4, Z5, Z6

T1

G1	S1	S2	D1	D2	P1	P2	L2	L3	C2	G3	G4	A1
A2	A3	A4	F1	N1	N2	H1	H2	H3	H4	H5	H6	H7
H8	Q1	Q2	Q3	Q4	Q5	Q6	Q7	F2	F3	F4	O1	O2
O3	J1	J2	J3	K1	K2	K3	Z1	Z2	Z3	Z4	Z5	Z6

T2

C1	C2	L1	E1	E2	E3	E4	E5

1NF:T11

G1	G2	G3	G4	A1	A2	A3	A4	F1	F2	F3	F4

T12

N1	N2	H1	H2	H3	H4	H5	H6	H7	H8	Q1	Q2	Q3	Q4
Q5	Q6	Q7	O1	O2	O3	P1	P2	L1	L2	L3	Z1	Z2	Z3
Z4	Z5	Z6											

T13

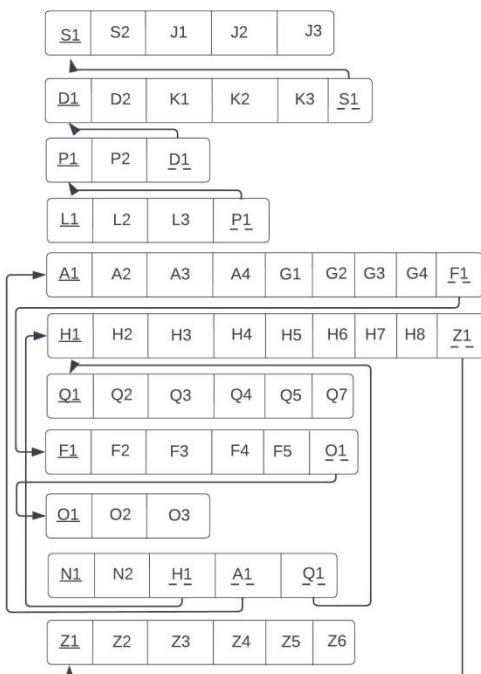
K1	K2	K3	D1	D2

T14

J1	J2	J3	S1	S2

T2 already exists**2NF:**

Partial dependency has been removed

3NF:**T2 already exists****BCNF:**

Already in BCNF Form as there is no determinant that is not a unique identifier

E. DATA DICTIONARY:

School_T:

Name	Data Type	Size	Remarks
SchoollID	VARCHAR	5	This is the primary key of School. E.g: “SETS” or “SLASS”
SchoolName	VARCHAR	45	This is the name of the School. E.g: “School of Engineering, Technology & Science”.

Program_T:

Name	Data Type	Size	Remarks
ProgramID	INTEGER		This is the primary key for a program. E.g: “1”
ProgramName	VARCHAR	30	This is the name of the program. E.g: “Bachelor of Science”
DepartmentID	VARCHAR	4	This is the foreign key from the Page 48 Department table. E.g: “CSE” or “BBA”

Department_T

Name	Data Type	Size	Remarks
DepartmentID	VARCHAR	5	This is the primary key for the Department table. E.g: “CSE”
DepartmentName	VARCHAR	45	This is the name of the department. E.g: “Computer

			Science and Engineering”.
SchoolID	VARCHAR	5	This is a foreign key from the School table. E.g: “SETS” or “SLASS”.

Student_T

Name	Data Type	Size	Remarks
StudentID	INTEGER		This is the primary key for the Student table. E.g: “2030479”.
FirstName	VARCHAR	20	This is the first name of the student. E.g: “Soumik”.
LastName	VARCHAR	20	This is the last name of the student. E.g: “Alam”.
DateOfBirth	DATE DD-MM-YYYY		This is the birth date of the student. E.g: “31-12-2001”.
Gender	VARCHAR	6	This is the gender of the student. E.g: “Male”.
Email	VARCHAR	30	This is the email of the student. E.g: “2030479@iub.edu.bd”
Phone	NUMERIC	11	This is the phone of the student. E.g: “01XXXXXXXXX”.
Address	VARCHAR	50	This is the address of the student. E.g: “House 238,Road 8,Tejgaon,Dhaka”
DepartmentID	VARCHAR	5	This is the foreign key from the Department table. E.g: “CSE”
ProgramID	INTEGER		This is the foreign key from the Program table. E.g: “1”
EnrollmentDate	DATE (dd-mm-yyyy)		This is the enrollment date of the student. Page 50 E.g.: “1-1-2020”

CO_T

Name	Data Type	Size	Remarks
COID	VARCHAR	5	This is the primary key for the CO table. E.g: "CO1".
CONum	INTEGER		This is the CO number. E.g: 1,2 etc.
CourseID	VARCHAR	8	This is the foreign key from the Course table. E.g: "CSE303"
PLOID	VARCHAR	5	This is the foreign key from the PLO table. E.g: "PLO1"

PLO_T

Name	Data Type	Size	Remarks
PLOID	VARCHAR	5	This is the primary key for Program Learning Outcome. E.g: "PLO1"
PLONum	INTEGER		This is the PLO number. E.g: "1"
Details	VARCHAR	50	This is the details for Program Learning Page 51 Outcome. E.g: "An ability to select and apply the knowledge, technique, skills and modern tools of the computer science and engineering discipline"
ProgramID	INTEGER		This is a foreign key from the Program table. E.g: "1"

Employee_T

Name	Date Type	Size	Remarks
EmployeeID	INTEGER		This is the primary key for Employee table. E.g: “1001”
FirstName	VARCHAR	20	This is the first name of the faculty. E.g: “Sadita”
LastName	VARCHAR	20	This is the last name of the faculty. E.g: “Ahmed”
Email	VARCHAR	30	This is the email address of the Student. E.g: “1675231@iub.edu.bd”
Address	VARCHAR	30	This is the address of the Faculty. E.g: “House 14, Road 21, Section 2, Mirpur,Dhaka, Bangladesh”
EmployeeType	CHAR	1	This is the type of the employee. E.g: “F”

Course_T

Name	Data Type	Size	Remarks
CourseID	VARCHAR	8	This is the Primary Key for the Course. E.g: “CSE203”
CourseName	VARCHAR	40	This is the name of the Course. E.g: “Database Management System”
CreditValues	INTEGER		This is the number of credits for the Course. E.g: “3”
CourseType	VARCHAR	10	This is the type of the Course. E.g: “Core”
ProgramID	INTEGER		This is the foreign key from the program table. E.g: “1”
COutlinID	INTEGER		This is the Foreign

			Key from Course table.
--	--	--	------------------------

Section_T

Name	Data Type	Size	Remarks
SectionID	INTEGER		This is the Primary Key for Section. E.g: “1”
SectionNum	INTEGER		This is the section number. E.g: “1”
CourseID	VARCHAR	8	This is the foreign key from the Course table. E.g: “CSE101”
FEmployeeID	NUMERIC	4	This is the foreign key from the Faculty table. E.g: “1001”
Semester	VARCHAR	6	This is the semester of the section. E.g: “Spring”

Question_T

Name	Data Type	Size	Remarks
QuestionID	INTEGER		This is the Primary Key for Question.
ExamType	VARCHAR	10	This is the name of the question. E.g: “Midterm”
TotalMarks	NUMBER		This is the total marks of the question. E.g: “30”
Category	VARCHAR	10	This is the category of the question. E.g: “Creating”
Level	VARCHAR	10	This is the difficulty of the question. E.g: “Medium”
COID	INTEGER		This is the Foreign Key from the Course Outcome table.

QuestionContent	INTEGER		This is the question number for question. E.g: “1,2,3....”
SectionID	INTEGER		This is the Foreign Key from Section table.
FEmployeeID			This is the Foreign Key from Faculty table.

Evaluation_T

Name	Data Type	Size	Remarks
EvaluationID	INTEGER		This is the Primary Key for Enrollment.
ObtainedMarks	DECIMAL	5,2	This is the obtained marks of the student. E.g: “24.5”
QuestionID	INTEGER		This is the foreign key Page 55 from the Question table.
CourseID	VARCHAR	8	This is the foreign key from the Course table. E.g: “CSE101”
StudentID	INTEGER		This is the foreign key from the Student table.

Dean_T

Name	Data Type	Size	Remarks
DEmployeeID	INTEGER		This is the foreign key from the Employee table. E.g: “4250”
SchoolID	VARCHAR	5	This is the SchoolID of the school DEAN manages. E.g: “SETS”
StartDate	DATE (dd-mm-yyyy)		This is the starting date. E.g: “01-03-2020”
EndDate	DATE (dd-mm-		This is the date

	yyyy)		DEAN retire from his post. E.g: “01-03- 2024”
--	--------	--	---

DepartmentHead_T

Name	Data Type	Size	Remarks
HEmployeeID	INTEGER		This is the foreign key Page 56 from the Employee table. E.g: “4250”
DepartmentID	VARCHAR	5	This is the DepartmentID of the department HEAD manages. E.g: “CSE”
StartDate	DATE (dd-mm-yyyy)		This is the starting date. E.g: “01-03-2020”
EndDate	DATE (dd-mm-yyyy)		This is the date HEAD retire from his post. E.g: “01-03- 2024”

Faculty_T

Name	Data Type	Size	Remarks
FEmoployeeID	INTEGER		This is the foreign key from the Employee table. E.g: “4250”
DepartmentID	VARCHAR	5	This is the DepartmentID of the department faculty belongs to. E.g: “CSE”
JoinDate	DATE (dd-mm-yyyy)		This is the starting date. E.g: “01-03-2020”
Rank	VARCHAR	20	This is the rank of the faculty. E.g: “Assistant Professor”
COutlineID	INTEGER		This is the Foreign Key from Course Outline table.

PreReqCourse_T

Name	Data Type	Size	Remarks
CourseID	VARCHAR	8	This is the foreign key from the Course table. E.g: “CSE303”
PreReqCourseID	VARCHAR	8	This is the foreign key from the Course table. E.g: CSE203

CourseOutline_T

Name	Data Type	Size	Remarks
COutlineID	INTEGER		This is the primary key from the Course Outline table. E.g: “1233”
MarkDistribution	VARCHAR	15	This is the percentage range for assessment. E.g: “Project- 50%, Assessment-50%”.

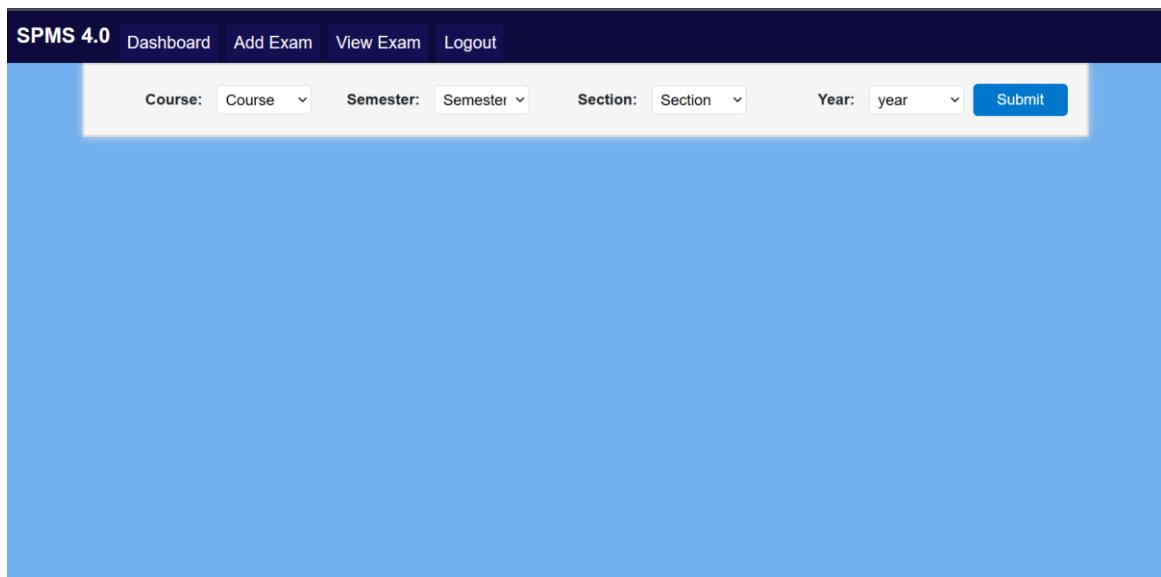
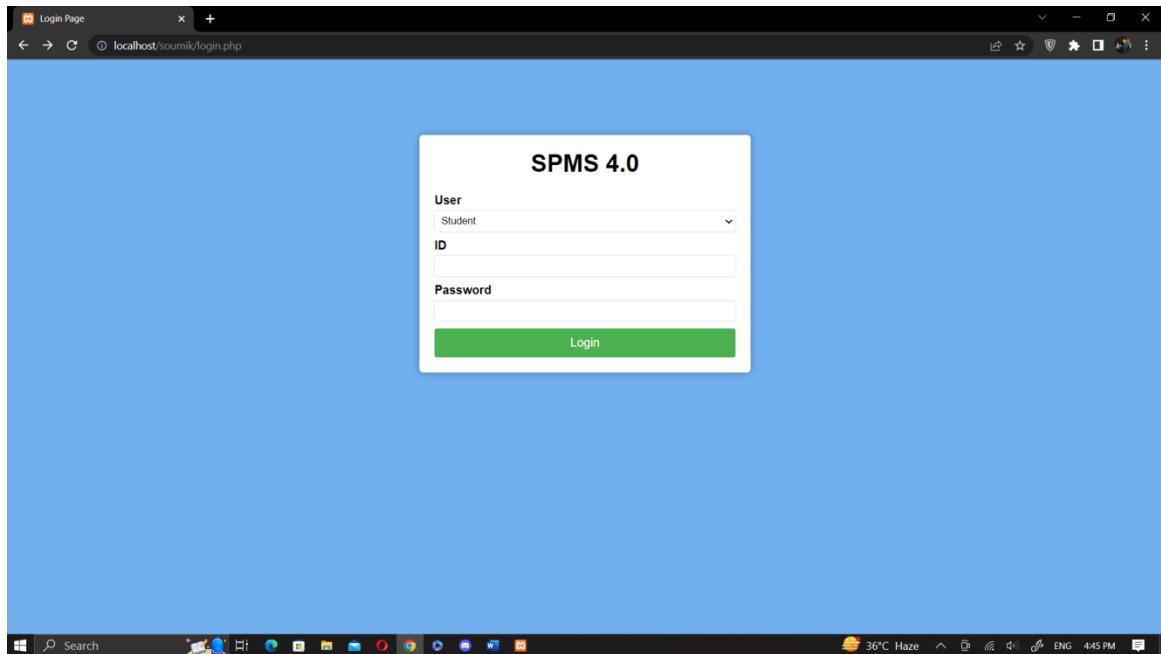
PercentageChart_T

Name	Data Type	Size	Remarks
PercentageChartid	INTEGER		This is the primary key from the PercentageChart table. E.g: “1,2,3”
StudentID	CHAR	7	This is the primary key for the Student table. E.g: “2030479”.
Educational_Year	NUMERIC	4	This represent students academic performance of

			selective year. E.g: “2023”
Educational_Semester	VARCHAR	10	This is the semester of the section. E.g: “Spring”
Enrolled_Course	VARCHAR	8	Students enrolled courses in different semester. E.g: “Summer”
Enrolled_Section	VARCHAR	2	Students enrolled courses in different section. E.g: “1,2,3,15”
Obtained_Percantages	VARCHAR	4	Students overall performances percentage and individual CO's percentages. E.g: “9%,40%,100%”

CHAPTER-4 PHYSICAL SYSTEM DESIGN

A. INPUT FORMS:



SPMS 4.0 Dashboard Exam Create Course Outline View Course Outline Logout

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

- 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.
- 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.
- 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.
- 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.
- 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.
- f. Class participation is vital for a better understanding of the topics of this course. Students are invited to raise questions.
- g. Students should take tutorials with the instructor during office hours. Prior appointment is required.
- h. Students must maintain the IUB code of conduct and ethical guidelines offered by the school of computer science and engineering.
- i. No working mobile phones are allowed in class. Using one for any purpose will result in serious consequences.

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 DISCRIMINATION 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.

Enter Assessment Type	Enter	Enter	Enter
Enter Assessment Type	Enter	Enter	Enter
Enter Assessment Type	Enter	Enter	Enter
Enter Assessment Type	Enter	Enter	Enter
			Enter

Total : 100%

The following chart will be followed for grading. Please note that for each category.
* Numbers are inclusive

A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F
80-100	85-89	80-84	75-79	70-74	65-69	60-64	55-59	50-54	45-49	40-44	

REFERENCE BOOK AND ADDITIONAL MATERIALS

Enter Reference Materials

```
createCourseOutlinePage1.php
1 <?php
2
3 include 'connect.php';
4
5 ?>
6
7 <!DOCTYPE html>
8 <html>
9 <head>
10 <title>Exam Form</title>
11 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.4.1/dist/css/bootstrap.min.css"
12 integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtu6ug5T0eNV6gBiFeWPGFN9MuhOf23Q9Ifjh" crossorigin="anonymous">
13 <link rel="stylesheet" href="style.css">
14 <link rel="stylesheet" href="questionform.css">
15 <style>
16 body {
17   display: flex;
18   flex-direction: column;
19   align-items: center;
20   justify-content: top;
21   height: 50vh;
22   margin: 0;
23   padding: 0;
24   font-family: Arial, sans-serif;
25   background-color: #71afee;
26 }
27
28 form {
29   display: flex;
30   flex-wrap: wrap;
31   justify-content: center;
32   align-items: center;
33   background-color: #f5f5f5;
34   border: 2px solid #ddd;
35   padding: 15px;
36   box-shadow: 0px 0px 10px #ddd;
37 }
38
39 label {
40   margin: 10px;
41   font-weight: bold;
42   width: 100px;
43   text-align: right;
44   display: block;
45 }
46
47 input, select {
48   padding: 5px;
49   margin: 5px;
50   border-radius: 5px;
51   border: 1px solid #ddd;
52   width: 100px;
53   font-size: 16px;
54   background-color: #fff;
```

```
        input[type="submit"] {
            background-color: #0077cc;
            color: #fff;
            border: none;
            cursor: pointer;
            transition: background-color 0.3s;
        }

        input[type="submit"]:hover {
            background-color: #005299;
        }
    
```

```
</style>
</head>
<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>
```

```
<div class="nav-links">
    <ul>
        <li><a href="employee_dashboard.php" target="_self">Dashboard</a></li>
        <li><a href="addExam.php" target="_self">Add Exam</a></li>
        <li><a href="viewExam.php" target="_self">View Exam</a></li>
        <li><a href="logout.php" target="_self">Logout</a></li>
    </ul>
</div>
</div>
</form>

<label for="courseID">Course:</label>
<select id="courseID" name="courseID">
    <option disabled selected>Course</option>
    <option value="CSC101">CSC101</option>
    <option value="CSC303">CSC303</option>
    <option value="MIS430">MIS430</option>
</select>

<label for="semester">Semester:</label>
<select id="semester" name="semester">
    <option disabled selected>Semester</option>
    <option value="spring">spring</option>
    <option value="summer">summer</option>
    <option value="autumn">autumn</option>
</select>

<label for="sectionNum">Section:</label>
<select id="sectionNum" name="sectionNum">
    <option disabled selected>Section</option>
    <option value="1">section-1</option>
    <option value="2">section-2</option>
    <option value="3">section-3</option>
</select>

<label for="year">Year:</label>
<select id="year" name="year">
    <option disabled selected>year</option>
    <option value="2020">2020</option>
    <option value="2021">2021</option>
    <option value="2022">2022</option>
    <option value="2023">2023</option>
</select>

<input type="submit" value="Submit">
</form>
</body>
```

```

<?php

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

    session_start();
    $year=$_POST['year'];
    $semester=$_POST['semester'];
    $sectionNum=$_POST['sectionNum'];
    $courseID=$_POST['courseID'];

    //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:createCourseOutline.php');

}?>
</body>
</html>

```

```

<?php

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

    session_start();
    $year=$_POST['year'];
    $semester=$_POST['semester'];
    $sectionNum=$_POST['sectionNum'];
    $courseID=$_POST['courseID'];

    //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:createCourseOutline.php');

}?

```

B. OUTPUT FORMS:

For CSV file:

```

<?php
// Include database configuration file
// require_once(__DIR__ . "/testfunction.php");
// require_once(__DIR__ . "/connect.php");
// require_once(__DIR__ . "/user_header.php");

if(isset($_POST['importsubmit'])){

    // Allowed mime types
    $mime_types = array('text/x-comma-separated-values', 'text/comma-separated-values', 'application/octet-stream', 'application/vnd.ms-excel', 'application/x-csv', 'text/x-csv', 'text/csv', 'application/csv', 'application/excel', 'application/vnd.msexcel', 'text/plain');

    // Validate whether selected file is csv formatted or not
    if(!empty($_FILES['file']['name']) && in_array($_FILES['file']['type'], $mime_types)) {

        // If the file is uploaded successfully
        if(is_uploaded_file($_FILES['file']['tmp_name'])){

            // open uploaded csv file with read-only mode
            $csvFile = fopen($_FILES['file']['tmp_name'], 'r');

            // skip the first line (header)
            fgetcsv($csvFile);

            // Loop through the file line by line
            while(($line = fgetcsv($csvFile)) != FALSE){
                $studentID = $line[0];
                $sectionNum = $line[1];
                $semester = isset($line[2]) ? $line[2] : '';
                $courseID = isset($line[3]) ? $line[3] : '';
                $year = isset($line[4]) ? $line[4] : '';
                $obtainGrade = isset($line[5]) ? $line[5] : '';

                // Check whether member already exists in the database with the same ID
                $prevQuery = "SELECT ID FROM members WHERE id = '".$line[0]."'";
                $prevResult = $db->query($prevQuery);

                if($prevResult->num_rows > 0){
                    // Update member data in the database
                    $db->query("UPDATE section_t SET Name = '".$Name."', CourseID = '".$CourseID."', Section = '".$Section."', Mid = '".$Mid."', Final = '".$Final."', Project = '".$Project."', Total = '".$Total."', CoTotal = '".$CoTotal."', Grade = '".$Grade."' WHERE Id = '".$Id."'");
                }else{
                    // Insert member data in the database
                    $db->query("INSERT INTO section_t (Id, Name, CourseID, Section, Mid, Final, Project, Total, CoTotal, Grade) VALUES ('".$Id."', '".$Name."', '".$CourseID."', '".$Section."', '".$Mid."', '".$Final."', '".$Project."', '".$Total."', '".$CoTotal."', '".$Grade."')"); //NOW(), NOW())
                }
            }

            // Close opened csv file
            fclose($csvFile);

            $qstring = '?status=success';
        } else {
            $qstring = '?status=error';
        }
    } else {
        $qstring = '?status=invalid_file';
    }
}

// Redirect to the listing page
header("Location: index.php".$qstring);
?>

```

```

<?php

require_once "connect.php";
if (isset($_POST['submit'])) {
    $isPageReloaded = $_POST['isPageReloaded'];
    if ($isPageReloaded === "false") {
        $studentID = $_POST['studentID'];
        // Display the student's information
    } else {
        // Do nothing or display a message asking the user to submit the form again
    }
}
// Get status message
if(!empty($_GET['status'])){
    switch($_GET['status']){
        case 'success':
            $statusType = 'alert-success';
            $statusMsg = 'Data has been imported successfully.';
            break;
        case 'error':
            $statusType = 'alert-danger';
            $statusMsg = 'Some problem occurred, please try again.';
            break;
        case 'invalid_file':
            $statusType = 'alert-danger';
            $statusMsg = 'Please upload a valid CSV file.';
            break;
        default:
            $statusType = '';
            $statusMsg = '';
    }
}

```

```

// Import data from CSV file
if(isset($_POST['importSubmit'])){
    // Allowed mime types
    $mime_types = array('text/x-comma-separated-values', 'text/comma-separated-values', 'application/octet-stream', 'application/vnd.ms-excel', 'application/x-csv', 'text/x-csv', 'text/csv', 'application/csv', 'application/excel', 'application/vnd.ms-excel', 'text/plain');

    // Validate whether selected file is CSV formatted or not
    if(empty($_FILES['file']['name']) && in_array($_FILES['file']['type'], $mime_types)){
        // If the file is uploaded successfully
        if(is_uploaded_file($_FILES['file']['tmp_name'])){
            // Open uploaded CSV file with read-only mode
            $csvFile = fopen($_FILES['file']['tmp_name'], 'r');

            // Skip the first line (header)
            fgetcsv($csvFile);

            // Loop through the file line by line
            while (($line = fgetcsv($csvFile)) !== FALSE) {
                $studentID = $line[0];
                $sectionNum = $line[1];
                $semester = isset($line[2]) ? $line[2] : '';
                $courseID = isset($line[3]) ? $line[3] : '';
                $year = isset($line[4]) ? $line[4] : '';
                $obtainGrade = isset($line[5]) ? $line[5] : '';
            }
        }
    }
}

```

```

    // Check whether section already exists in the database with the same sectionNum, semester, courseID, and year
    $prevQuery = "SELECT sectionID FROM section_t WHERE sectionNum = '$sectionNum' AND semester = '$semester' AND courseID = '$courseID' AND year = '$year'";
    $prevResult = $con->query($prevQuery);

    if ($prevResult->num_rows > 0) {
        // Get sectionID
        $row = $prevResult->fetch_assoc();
        $sectionID = $row['sectionID'];
    } else {
        // Insert section data into section_t table
        $con->query("INSERT INTO section_t (sectionNum, semester, courseID, year) VALUES ('$sectionNum', '$semester', '$courseID', '$year')");

        // Get sectionID
        $sectionID = $con->insert_id;
    }
    // Insert registration data into registration_t table
    $con->query("INSERT INTO registration_t (sectionID, studentID) VALUES ('".$sectionID."', '".$studentID."')");

    // Get registrationID
    $registrationID = $con->insert_id;

    // Insert student course performance data into student_course_performance_t table
    $con->query("INSERT INTO student_course_performance_t (registrationID, obtainGrade) VALUES ('$registrationID', '$obtainGrade')");
}

```

```

    // Close opened CSV file
    fclose($csvFile);

    $qstring = '?status=success';
} else {
    $qstring = '?status=error';
}
} else {
    $qstring = '?status=invalid_file';
}

}

```

```

$sections = array();
$result = $con->query("SELECT registration_t.studentID, section_t.sectionNum, section_t.semester, section_t.courseID, section_t.year, student_course_performance_t.obtainGrade,
CASE
    WHEN student_course_performance_t.obtainGrade = 'A' THEN 90
    WHEN student_course_performance_t.obtainGrade = 'A-' THEN 85
    WHEN student_course_performance_t.obtainGrade = 'B+' THEN 80
    WHEN student_course_performance_t.obtainGrade = 'B' THEN 75
    WHEN student_course_performance_t.obtainGrade = 'B-' THEN 70
    WHEN student_course_performance_t.obtainGrade = 'C+' THEN 65
    WHEN student_course_performance_t.obtainGrade = 'C' THEN 60
    WHEN student_course_performance_t.obtainGrade = 'C-' THEN 55
    WHEN student_course_performance_t.obtainGrade = 'D+' THEN 50
    WHEN student_course_performance_t.obtainGrade = 'D' THEN 45
    ELSE 43
END as co
FROM registration_t
JOIN section_t ON registration_t.sectionID = section_t.sectionID
JOIN student_course_performance_t ON registration_t.registrationID = student_course_performance_t.registrationID");

if($result->num_rows > 0){
    while($row = $result->fetch_assoc()){

        $sectionKey = $row['sectionNum'].'-'.$row['semester'].'-'.$row['courseID'].'-'.$row['year'];

        if(!isset($sections[$sectionKey])){
            $sections[$sectionKey] = array(
                'studentID' => $row['studentID'],
                'sectionNum' => $row['sectionNum'],
                'semester' => $row['semester'],
                'courseID' => $row['courseID'],
                'year' => $row['year'],
                'obtainGrade' => $row['obtainGrade'],
                'students' => array()
            );
        }
    }
}

```

```

    $sections[$sectionKey]['students'][] = array(
        'studentID' => $row['studentID'],
        'sectionNum' => $row['sectionNum'],
        'semester' => $row['semester'],
        'courseID' => $row['courseID'],
        'year' => $row['year'],
        'obtainGrade' => $row['obtainGrade'],
        'co' => $row['co']
    );
}

?>

<?php
// require_once(__DIR__."/testfunction.php");
// require_once(__DIR__."/connect.php");
// require_once(__DIR__."/user_header.php");
require_once "connect.php";
//echo pre($_SESSION);

if (isset($_POST['submit'])) {
    $studentID = $_POST['studentID'];
} elseif (isset($_SESSION['ID'])) {
    $studentID = $_SESSION['ID'];
}

// Fetch student data
?>

```

```

<!DOCTYPE html>
<!-- Coding by CodingLab | www.codinglabweb.com -->
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <!-- Bootstrap CSS -->
    <!-- <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.4.1/dist/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtuG5ZToeNV6gBiFeWPGFN9Mu0f23Q9Ifjh" crossorigin="anonymous"> -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-K94CHFLLe+nY2dmcWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJ0Z" crossorigin="anonymous">

    <!-- Bootstrap CSS -->
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.4.1/dist/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGsO3+Hhxv8T/Q5PaXtkKtuG5ZToeNV6gBiFeWPGFN9Mu0f23Q9Ifjh" crossorigin="anonymous">
    <title>Employee Dashboard</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" -->
    <!-- ===== CSS ===== -->
    <link rel="stylesheet" href="styleplo.css">
    <!-- <link rel="stylesheet" href="style3.css" -->

    <!--===== Boxicons CSS ===== -->
    <link href="https://unpkg.com/boxicons@2.1.1/css/boxicons.min.css" rel="stylesheet">

    <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
    <script type="text/javascript"></script>

```

```
<style>
  body{
    background-color:#71afee;
    /* background-color:#155977; */
  }

  ::placeholder{
    color:white;
  }

  ::-ms-input-placeholder{
    color:white;
  }

  :-ms-input-placeholder{
    color:white;
  }

</style>

</head>

<body>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha2/dist/js/bootstrap.bundle.min.js" integrity="sha384-qKXV1j0HvMUECBQ+QVp7JcfG1760yU08IQ+GpUo5hlbpg51QRiuqHAJz8+BxE/N" crossorigin="anonymous"></script>

<!-- Navbar start -->
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container">

    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"
      aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav ms-auto mb-2 mb-lg-0">

        <li class="nav-item">
          <a class="btn btn-primary" data-bs-toggle="offcanvas" href="#offcanvasExample" role="button" aria-controls="offcanvasExample">
            Menu
          </a>
        </li>
      </ul>
    </div>
  </div>
</nav>
<!-- Navbar End -->
```

```

<!-- off canvas start -->



<div class="offcanvas-header">
    <h5 class="offcanvas-title" id="offcanvasExampleLabel">SPMS 4.0</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas" aria-label="Close"></button>
  </div>
  <div class="offcanvas-body">
    <ul class="navbar-nav">
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
          Course Performance
        </a>
        <ul class="dropdown-menu">
          <li><a class="dropdown-item" href="school_department_program_stats.php">School/Department/Program-wise</a></li>
          <li><a class="dropdown-item" href="courseWisePerformance.php">Course-wise</a></li>
          <li><a class="dropdown-item" href="instructorWisePerformance.php">Instructor-wise</a></li>
          <li><a class="dropdown-item" href="instructorWiseChosenCourse.php">Instructor-wise(Chosen Course)</a></li>
          <li><a class="dropdown-item" href="school_department_program_stats.php">School-Department-Program Stats</a></li>
        </ul>
      </li>
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
          Exam
        </a>
        <ul class="dropdown-menu">
          <li><a class="dropdown-item" href="addExam.php">Add Exam</a></li>
          <li><a class="dropdown-item" href="viewExam.php">View Exam</a></li>
          <li><a class="dropdown-item" href="viewStudentAnswerScript.php">Evaluate Exam Script</a></li>
          <li><a class="dropdown-item" href="answerScriptGrading.php">Answer Script Grading</a></li>
        </ul>
      </li>
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
          Course Outline
        </a>
      </li>
    </ul>
  </div>


```

```

<ul class="dropdown-menu">
  <li><a class="dropdown-item" href="createCourseOutline.php">Create Course Outline</a></li>
  <li><a class="dropdown-item" href="viewCourseOutline.php">View Course Outline</a></li>
  <li><a class="dropdown-item" href="createCourseOutlinePage1.php">Create Course Outline Page1</a></li>
  <li><a class="dropdown-item" href="createpdf.php">Create PDF</a></li>
</ul>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
    Enrollment Statistics
  </a>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="enrollmentStatistics.php">Student Enrollment Statistics</a></li>
    <li><a class="dropdown-item" href="ploAnalysis.php">PLO Analysis</a></li>
    <li><a class="dropdown-item" href="ploAchieveStats.php">PLO Achievement Statistics</a></li>
    <li><a class="dropdown-item" href="performanceStats.php">Student Performance Stats</a></li>
    <li><a class="dropdown-item" href="spiderChart.php">Spider Chart</a></li>
  </ul>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
    Overall Analysis
  </a>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="ploAnalysisOverall.php">PLO Analysis (Overall, CO Wise, Course Wise)</a></li>
    <li><a class="dropdown-item" href="ploAnalysisDepartmentProgramSchoolAverage.php">PLO Analysis With Department/Program/School Average</a></li>
  </ul>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
    PLO Comparison
  </a>

```

```


- <a class="dropdown-item" href="ploComparisonProgram.php">PLO Comparison(Program)</a></li>
- <a class="dropdown-item" href="ploComparisonCourse.php">PLO Comparison (Course)</a></li>
- <a class="dropdown-item" href="ploComparisonStudent.php">PLO Comparison(Student)</a></li>
- <a class="dropdown-item" href="ploComparisonSchool.php">PLO Comparison(School)</a></li>
- <a class="dropdown-item" href="ploComparisonDepartment.php">PLO Comparison(Department)</a></li>


</ul>
</li>
<li class="nav-item dropdown">
<a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
    Answer Script
</a>
<ul class="dropdown-menu">
    <li><a class="dropdown-item" href="submitAnswerScript.php">Submit Answer Script</a></li>
    <li><a class="dropdown-item" href="viewStudentAnswerScript.php">View Student Answer Script</a></li>
</ul>
</li>
<li class="nav-item">
    <a class="nav-link" href="percentageChart.php">Percentage Chart</a>
</li>

<li class="nav-item">
    <a class="nav-link" href="logout.php">Log Out</a>
</li>
</ul>
</div>
</div>

```

```

<div style="margin-top: 100px;" class="container">
    <h2>Student Course Performance</h2> <br>
    <!-- Display status message -->
    <?php if(!empty($statusMsg)){ ?>
        <div class="col-xs-12">
            <div class="alert" <?php echo $statusType; ?>><?php echo $statusMsg; ?></div>
        <?php } ?>

    <!-- Import Link -->
    <div class="col-md-12 head">
        <div class="float-right">
            <a href="javascript:void(0);</a> class="btn btn-success" onclick="formToggle('importFrm');"><i class="plus"></i> IMPORT</a> <br>
        </div>
        <br>
        <br>

    <!-- CSV file upload form -->
    <div class="col-md-12" id="importFrm" style="display: none;">
        <form action="importStudentdata.php" method="post" enctype="multipart/form-data">
            <input type="file" name="file" />
            <input type="submit" class="btn btn-primary" name="importSubmit" value="IMPORT">
        </form>
    </div>
</div>

```

```

    <!-- Display data -->
    <!-- Display data -->
<div class="row">
    <?php if(empty($sections)){ ?>
        <div class="col-xs-12">
            <div class="alert alert-info">No data found.</div>
        </div>
    <?php } else{ ?>

        <div class="col-xs-12"> <br>
            <!-- <?php echo $sections[array_key_first($sections)][['studentID']].' - '.$sections[array_key_first($sections)][['sectionNum']].' - '.$sections[array_key_first($sections)][['semester']].' - '.$sections[array_key_first($sections)][['courseID']].' - '.$sections[array_key_first($sections)][['year']].' - '.$sections[array_key_first($sections)][['obtainGrade']]; ?></h3> -->
            <table class="table table-striped table-bordered">
                <thead class="thead-dark">
                    <tr>
                        <input type="hidden" id="isPageReloaded" name="isPageReloaded" value="false">

```

```

        <th>studentID</th>
        <th>sectionNum</th>
        <th>semester</th>
        <th>courseID</th>
        <th>year</th>
        <th>obtainGrade</th>
        <th>co</th>
        <th>co1</th>
        <th>co2</th>
        <th>co3</th>
        <th>co4</th>

    </tr>
</thead>
<tbody>
<?php foreach($sections as $section){ ?>
    <?php foreach($section['students'] as $student){ ?>
        <tr>
            <td><?php echo $student['studentID']; ?></td>
            <td><?php echo $student['sectionNum']; ?></td>
            <td><?php echo $student['semester']; ?></td>
            <td><?php echo $student['courseID']; ?></td>
            <td><?php echo $student['year']; ?></td>
            <td><?php echo $student['obtainGrade']; ?></td>
            <td><?php echo $student['co']; ?>%</td>
            <td><?php echo $student['co']; ?>%</td>
            <td><?php echo $student['co']; ?>%</td>
            <td><?php echo $student['co']; ?>%</td>
            <td><?php echo $student['co']; ?>%</td>
        </tr>
        <?php } ?>
    </tbody>
</table>
</div>
<?php } ?>
</div>

```

```

<!-- Overall plo -->
<script>
    window.onload = function() {
        if (performance.navigation.type === 1) { // If the page is reloaded
            document.getElementById("isPageReloaded").value = "true";
            document.getElementById("yourFormId").submit(); // Replace 'yourFormId' with the actual ID of your form
        }
    }
</script>

<!-- Co wise plo -->
<script>
function coWisePlo(){
    ...
}
</script>

<!-- course wise plo -->

<script>
function courseWisePlo(){
    ...
}
</script>

</body>
</html>

<!-- Bootstrap library -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@3.3.7/dist/js/bootstrap.min.js"
integrity="sha384-Tc5IQib02qvyjSMfHjOMaLkfuWxZxUPnCJA7l2mCWNIpG9mGCD8wGNicPD7Txa" crossorigin="anonymous"></script>

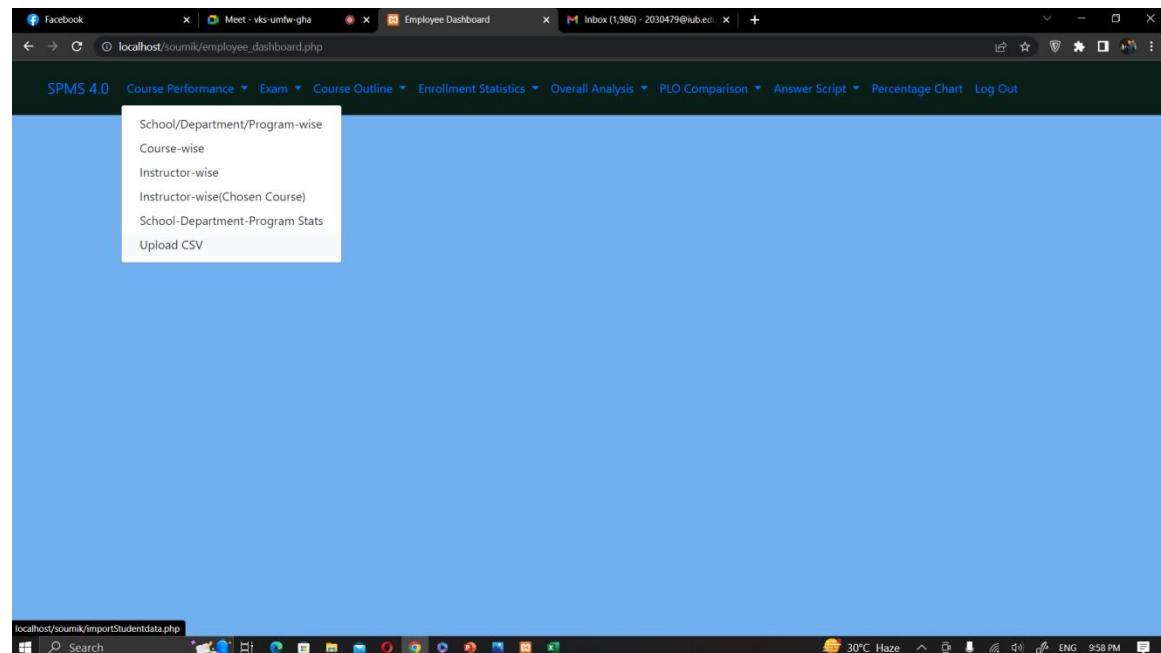
```

```

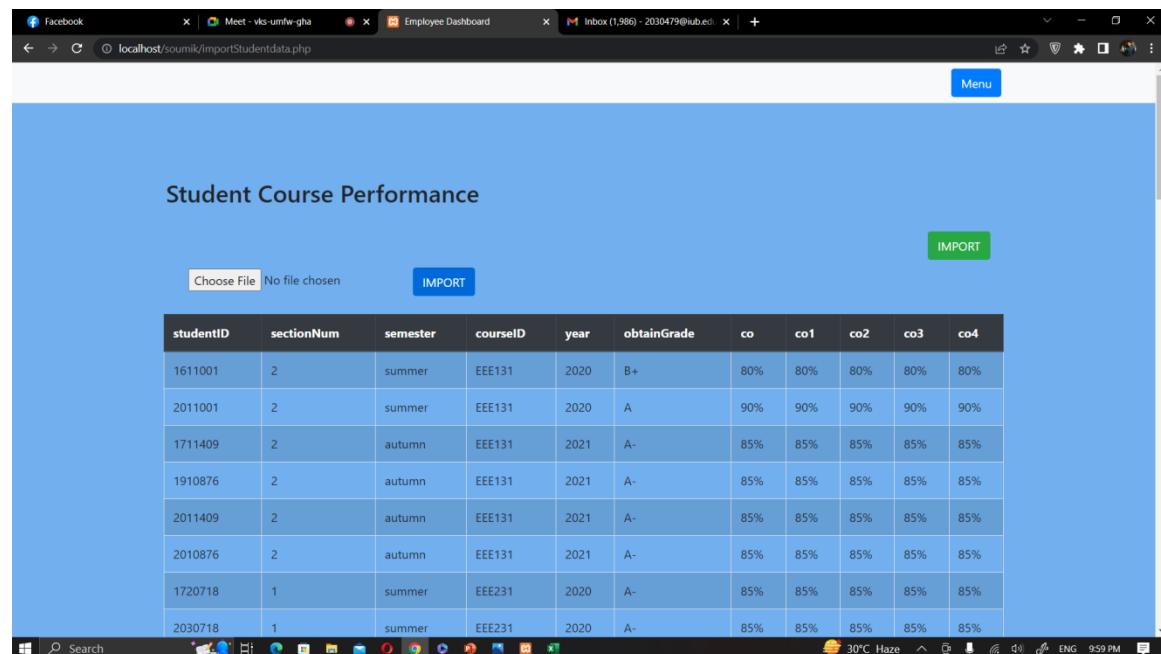
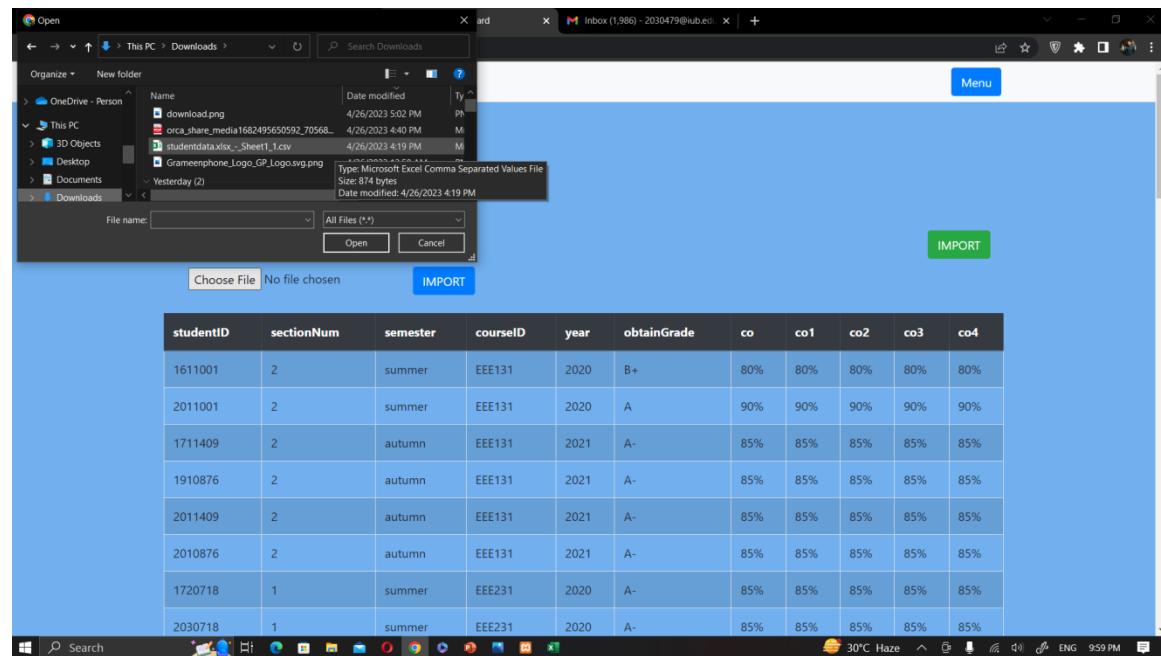
<!-- jquery library -->
<script src="https://code.jquery.com/jquery-3.6.0.min.js" integrity="sha384-Dfxdz2htPH01ssss5nCTpuj/zY4C+OGpamoFVY38MVBN+E+bVVUEw+ORxARkfj" crossorigin="anonymous"></script>

<script>
    function formToggle(ID){
        var element = document.getElementById(ID);
        if(element.style.display === "none"){
            element.style.display = "block";
        }else{
            element.style.display = "none";
        }
    }
</script>
</body>
</html>

```



studentID	sectionNum	semester	courseID	year	obtainGrade	co	co1	co2	co3	co4
1611001	2	summer	EEE131	2020	B+	80%	80%	80%	80%	80%
2011001	2	summer	EEE131	2020	A	90%	90%	90%	90%	90%
1711409	2	autumn	EEE131	2021	A-	85%	85%	85%	85%	85%
1910876	2	autumn	EEE131	2021	A-	85%	85%	85%	85%	85%
2011409	2	autumn	EEE131	2021	A-	85%	85%	85%	85%	85%
2010876	2	autumn	EEE131	2021	A-	85%	85%	85%	85%	85%
1720718	1	summer	EEE231	2020	A-	85%	85%	85%	85%	85%
2030718	1	summer	EEE231	2020	A-	85%	85%	85%	85%	85%



```
<?php

// Include database connection and header files
// require_once(_DIR_."/testfunction.php");
// require_once(_DIR_ . "/connect.php");
require_once "connect.php";
// require_once(_DIR_ . "/user_header.php");
$result = false;
$grade_mapping = [
    "A" => 90,
    "A-" => 85,
    "B+" => 80,
    "B" => 75,
    "B-" => 70,
    "C+" => 65,
    "C" => 60,
    "C-" => 55,
    "D+" => 50,
    "D" => 45,
    "F" => 43,
];
$grade_point_mapping = [
    "A" => 4.0,
    "A-" => 3.7,
    "B+" => 3.3,
    "B" => 3.0,
    "B-" => 2.7,
    "C+" => 2.3,
    "C" => 2.0,
    "C-" => 1.7,
    "D+" => 1.3,
    "D" => 1.0,
    "F" => 0.0,
];

```

```
37 if ($_SERVER["REQUEST_METHOD"] == "POST") {
38     if(isset($_POST["studentID"]) && $_POST["sectionNum"] && isset($_POST["semester"]) && isset($_POST["courseID"]) && isset($_POST
39     ["year"]) && isset($_POST["obtainGrade"])) {
40         // && isset($_POST["coNum"])
41
42         $studentID = $_POST["studentID"];
43         $sectionNum = $_POST["sectionNum"];
44         $semester = $_POST["semester"];
45         $courseID = $_POST["courseID"];
46         $year = $_POST["year"];
47         $obtainGrade = $_POST["obtainGrade"];
48
49
50         // Check if section already exists, if not insert it
51         $sql = "INSERT INTO section_t (sectionNum, semester, courseID, year)
52             SELECT * FROM (SELECT '$sectionNum', '$semester', '$courseID', '$year') AS tmp
53             WHERE NOT EXISTS (
54                 SELECT sectionID FROM section_t WHERE sectionNum='$sectionNum' AND semester='$semester' AND courseID='$courseID' AND
55                 year='$year'
56             ) LIMIT 1";
57
58         if ($con->query($sql) === TRUE) {
59             echo "New section added successfully.";
60         } else {
61             echo "Error: " . $sql . "<br>" . $con->error;
62         }
63
64         // Get the sectionID for the inserted section
65         $sql = "SELECT sectionID FROM section_t WHERE sectionNum='$sectionNum' AND semester='$semester' AND courseID='$courseID' AND
66         year='$year'";
67         $result = $con->query($sql);
68         $row = $result->fetch_assoc();
69         $sectionID = $row['sectionID'];
70
71         // Insert into registration_t table
72
73 }
```

```

// Insert into registration_t table
$sql = "INSERT INTO registration_t (sectionID, studentID) VALUES ('$sectionID', '$studentID')";

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

// Get the registrationID for the inserted registration
$sql = "SELECT registrationID FROM registration_t WHERE sectionID='$sectionID' AND studentID='$studentID'";
$result = $con->query($sql);
$row = $result->fetch_assoc();
$registrationID = $row['registrationID'];

// Insert into student_course_performance_t table
// $sql = "INSERT INTO student_course_performance_t (registrationID, obtainGrade) VALUES ('$registrationID', '$obtainGrade')";

// if ($con->query($sql) === TRUE) {
//     echo "New student_course_performance added successfully.";
// } else {
//     echo "Error: " . $sql . "<br>" . $con->error;
// }
$markObtained = $grade_mapping[$obtainGrade];
$totalMarksObtained = $grade_mapping[$obtainGrade];
$gradePoint = $grade_point_mapping[$obtainGrade];

$sql = "INSERT INTO student_course_performance_t (registrationID, totalMarksObtained, gradePoint, obtainGrade) VALUES
('$registrationID', '$totalMarksObtained', '$gradePoint', '$obtainGrade')";

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

```

```

$questionID = $con->insert_id;
$coNum = $con->insert_id;
// Insert data into question_t
$markPerQuestion = 100;
$sql = "INSERT INTO question_t (markPerQuestion,coNum) VALUES ( '$markPerQuestion',$coNum)";

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

```

```

144 $sql = "INSERT INTO answer_t (markobtained, registrationID, questionID) VALUES ('$markobtained', '$registrationID', '$questionID')";
145
146 if ($con->query($sql) === TRUE) {
147     echo "New answer added successfully.";
148 } else {
149     echo "Error: " . $sql . "<br>" . $con->error;
150 }
151
152 // Fetch the data to display in the HTML table
153 $sql = "SELECT r.studentID, s.sectionNum, s.semester, s.courseID, s.year, scp.obtainGrade
154 FROM student_course_performance_t scp
155 JOIN registration_t r ON scp.registrationID = r.registrationID
156 JOIN section_t s ON r.sectionID = s.sectionID
157 WHERE r.studentID= '$studentID'";
158
159 $result = $con->query($sql);
160
161 }
162
163 $con->close();
164
165 ?>

```

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ" crossorigin="anonymous">

    <!-- Bootstrap CSS -->
    <!-- <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.1/dist/css/bootstrap.min.css" integrity="sha384-Vkoo8x4CGs03+Hhxv8T/Q5PaXtkKtu6ug5ToeNV6gBiFeWPGFN9Mu0f23Q9Ifjh" crossorigin="anonymous"> -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYe+M/SXH301p5ILy+dN9+nJOZ" crossorigin="anonymous">

    <title>Employee Dashboard</title>
    <link rel="stylesheet" href="navbar.css">

    <!--Google Font-->
    <link href="https://fonts.googleapis.com/css2?family=Poppins:wght@400;600&display=swap" rel="stylesheet">

    <!-- <link rel="stylesheet" href="style.css" > -->
    <!-- ===== CSS ===== -->
    <link rel="stylesheet" href="styleplo.css">
    <!-- <link rel="stylesheet" href="style3.css" > -->

    <!-- ===== Boxicons CSS ===== -->
    <link href="https://unpkg.com/boxicons@2.1.1/css/boxicons.min.css" rel='stylesheet'>

```

```

<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript"></script>

<style>
    body{
        background-color:#71afee;
        /* background-color:#155977; */
    }

    ::placeholder{
        color:white;
    }

    ::-ms-input-placeholder{
        color:white;
    }

    :-ms-input-placeholder{
        color:white;
    }

    </style>
</head>
<body>

```

```

<!-- Navbar start -->
<nav class="navbar navbar-expand-lg bg-body-tertiary">
  <div class="container">

    <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarSupportedContent"
      aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
      <span class="navbar-toggler-icon"></span>
    </button>
    <div class="collapse navbar-collapse" id="navbarSupportedContent">
      <ul class="navbar-nav ms-auto mb-2 mb-lg-0">

        <li class="nav-item">
          <a class="btn btn-primary" data-bs-toggle="offcanvas" href="#offcanvasExample" role="button" aria-controls="offcanvasExample">
            | Menu
          </a>
        </li>

      </ul>
    </div>
  </div>
</nav>
<!-- Navbar End -->

```

```

<!-- off canvas start -->

<div class="offcanvas offcanvas-start" tabindex="-1" id="offcanvasExample" aria-labelledby="offcanvasExampleLabel">
  <div class="offcanvas-header">
    <h5 class="offcanvas-title" id="offcanvasExampleLabel">SPMS 4.0</h5>
    <button type="button" class="btn-close" data-bs-dismiss="offcanvas" aria-label="Close"></button>
  </div>
  <div class="offcanvas-body">
    <ul class="navbar-nav">
      <li class="nav-item dropdown">
        <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
          Course Performance
        </a>
        <ul class="dropdown-menu">
          <li><a class="dropdown-item" href="school_department_program_stats.php">School/Department/Program-wise</a></li>
          <li><a class="dropdown-item" href="coursewisePerformance.php">Course-wise</a></li>
          <li><a class="dropdown-item" href="instructorwisePerformance.php">Instructor-wise</a></li>
          <li><a class="dropdown-item" href="instructorWiseChosenCourse.php">Instructor-wise(Chosen Course)</a></li>
          <li><a class="dropdown-item" href="school_department_program_stats.php">School-Department-Program Stats</a></li>
        </ul>
      </li>
    </ul>
  </div>
</div>

```

```

</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
    Exam
  </a>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="addExam.php">Add Exam</a></li>
    <li><a class="dropdown-item" href="viewExam.php">View Exam</a></li>
    <li><a class="dropdown-item" href="viewStudentAnswerScript.php">Evaluate Exam Script</a></li>
    <li><a class="dropdown-item" href="answerScriptGrading.php">Answer Script Grading</a></li>
  </ul>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
    Course Outline
  </a>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="createCourseOutline.php">Create Course Outline</a></li>
    <li><a class="dropdown-item" href="viewCourseOutline.php">View Course Outline</a></li>
    <li><a class="dropdown-item" href="createCourseOutlinePage1.php">Create Course Outline Page1</a></li>
    <li><a class="dropdown-item" href="createpdf.php">Create PDF</a></li>
  </ul>
</li>
<li class="nav-item dropdown">
  <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
    Enrollment Statistics
  </a>
  <ul class="dropdown-menu">
    <li><a class="dropdown-item" href="enrollmentStatistics.php">Student Enrollment Statistics</a></li>
    <li><a class="dropdown-item" href="ploAnalysis.php">PLO Analysis</a></li>
    <li><a class="dropdown-item" href="ploAchieveStats.php">PLO Achievement Statistics</a></li>
    <li><a class="dropdown-item" href="performanceStats.php">Student Performance Stats</a></li>
    <li><a class="dropdown-item" href="spiderChart.php">Spider Chart</a></li>
  </ul>
</li>

```

```

<li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
        Overall Analysis
    </a>
    <ul class="dropdown-menu">
        <li><a class="dropdown-item" href="ploAnalysisOverall.php">PLO Analysis (Overall, CO Wise, Course Wise)</a></li>
        <li><a class="dropdown-item" href="ploAnalysisDepartmentProgramSchoolAverage.php">PLO Analysis With Department/Program/School Average</a></li>
    </ul>
</li>
<li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
        PLO Comparison
    </a>
    <ul class="dropdown-menu">
        <li><a class="dropdown-item" href="ploComparisonProgram.php">PLO Comparison(Program)</a></li>
        <li><a class="dropdown-item" href="ploComparisonCourse.php">PLO Comparison (Course)</a></li>
        <li><a class="dropdown-item" href="ploComparisonStudent.php">PLO Comparison(Student)</a></li>
        <li><a class="dropdown-item" href="ploComparisonSchool.php">PLO Comparison(School)</a></li>
        <li><a class="dropdown-item" href="ploComparisonDepartment.php">PLO Comparison(Department)</a></li>
    </ul>
</li>
<li class="nav-item dropdown">
    <a class="nav-link dropdown-toggle" href="#" role="button" data-bs-toggle="dropdown" aria-expanded="false">
        Answer Script
    </a>
    <ul class="dropdown-menu">
        <li><a class="dropdown-item" href="submitAnswerScript.php">Submit Answer Script</a></li>
        <li><a class="dropdown-item" href="viewStudentAnswerScript.php">View Student Answer Script</a></li>
    </ul>
</li>
<li class="nav-item">
    <a class="nav-link" href="percentageChart.php">Percentage Chart</a>
</li>

```

Lo 150 Col 37 Esc 2 L1E 8 GBL DUD ⌂ Go Live

```

<li class="nav-item">
    <a class="nav-link" href="logout.php">Log Out</a>
</li>
</ul>
</div>
</div>
</div>

<nav class="sidebar close">

<header>
    <div class="image-text">
        </div>
        <i class='bx bx-chevron-right toggle'></i>
</header>

</nav>

```

```
<div style="margin-top: 50px;" class="container">
<h2>Student Course Performance</h2>

<div class="col-xs-12">

    <form action="percentageForm.php" method="post">
        <div class="form-group">
            <label for="input1"><h4>studentID:</h4></label>
            <input type="text" class="form-control" id="input1" placeholder="Enter studentID" name="studentID" >
        </div>
        <div class="form-group">
            <label for="input2"><h4>sectionNum:</h4></label>
            <input type="text" class="form-control" id="input2" placeholder="Enter sectionNum" name="sectionNum" >
        </div>
        <div class="form-group">
            <label for="input3"><h4>semester:</h4></label>
            <input type="text" class="form-control" id="input3" placeholder="Enter semester" name="semester" >
        </div>
        <div class="form-group">
            <label for="input4"><h4>courseID:</h4></label>
            <input type="text" class="form-control" id="input4" placeholder="Enter courseID" name="courseID" >
        </div>
        <div class="form-group">
            <label for="input5"><h4>year:</h4></label>
            <input type="text" class="form-control" id="input5" placeholder="Enter year" name="year" >
        </div>
        <div class="form-group">
            <label for="input6"><h4>obtainGrade:</h4></label>
            <input type="text" class="form-control" id="input6" placeholder="Enter obtainGrade" name="obtainGrade" >
        </div>
        <br>
        <button type="submit" class="btn btn-primary" value="submit2" onclick = "percentageForm.php" >Submit</button>
    </form>
    <br>
</div>
```

```

<?php
if($result != false){
    if($result->num_rows > 0){

        while ($row = $result->fetch_assoc()) {
            echo "<tr>
                <td>" . $row['studentID'] . "</td>
                <td>" . $row['sectionNum'] . "</td>
                <td>" . $row['semester'] . "</td>
                <td>" . $row['courseID'] . "</td>
                <td>" . $row['year'] . "</td>
                <td>" . $row['obtainGrade'] . "</td>
                <td>" . $grade_mapping[$row['obtainGrade']] . "%</td>
                <td>" . $grade_mapping[$row['obtainGrade']] . "%</td>

            </tr>";
        }
    }
}

?>

```

```

</div>
</div>
</div>

```

```

<!-- Overall plot -->
<script>
</script>

<!-- Co wise plot -->
<script>
function coWisePlot(){
    ...
}
</script>

<!-- course wise plot -->
<script>
function courseWisePlot(){
    ...
}
</script>

</body>
</html>

<!-- Bootstrap Library -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@3.3.7/dist/js/bootstrap.min.js"
integrity="sha384-Tc5Qib027qvyjSMfHj0MalkfuWxZxUPnCJA7l2mCWNIpG9mGCD8wGNICPD7Txa"
crossorigin="anonymous"></script>

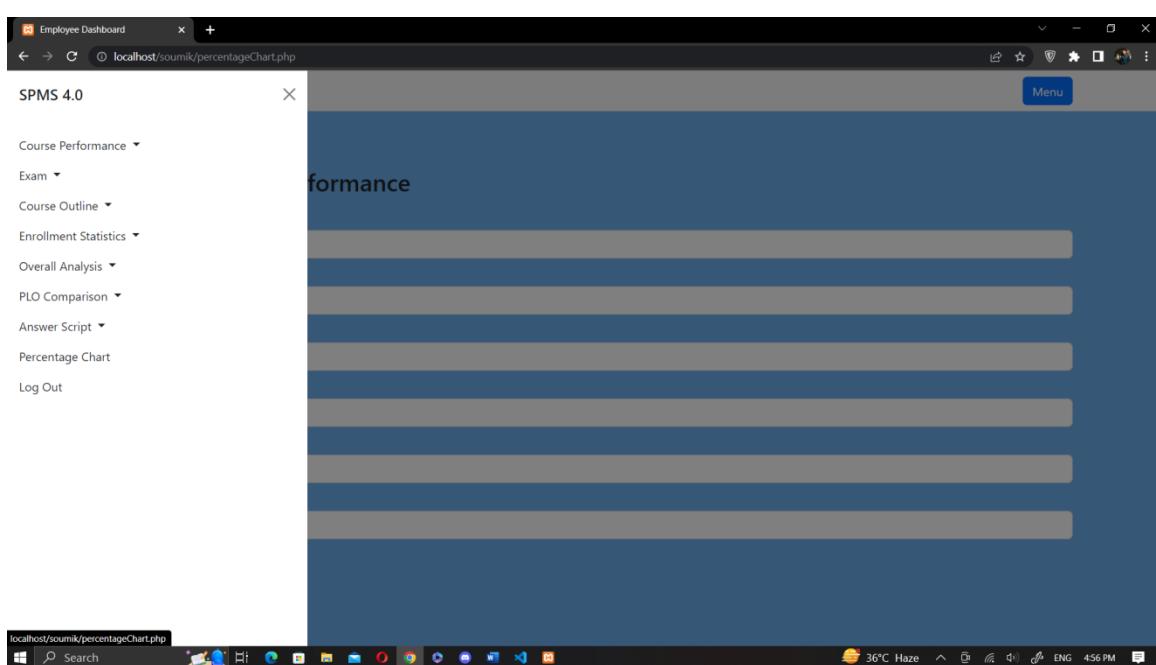
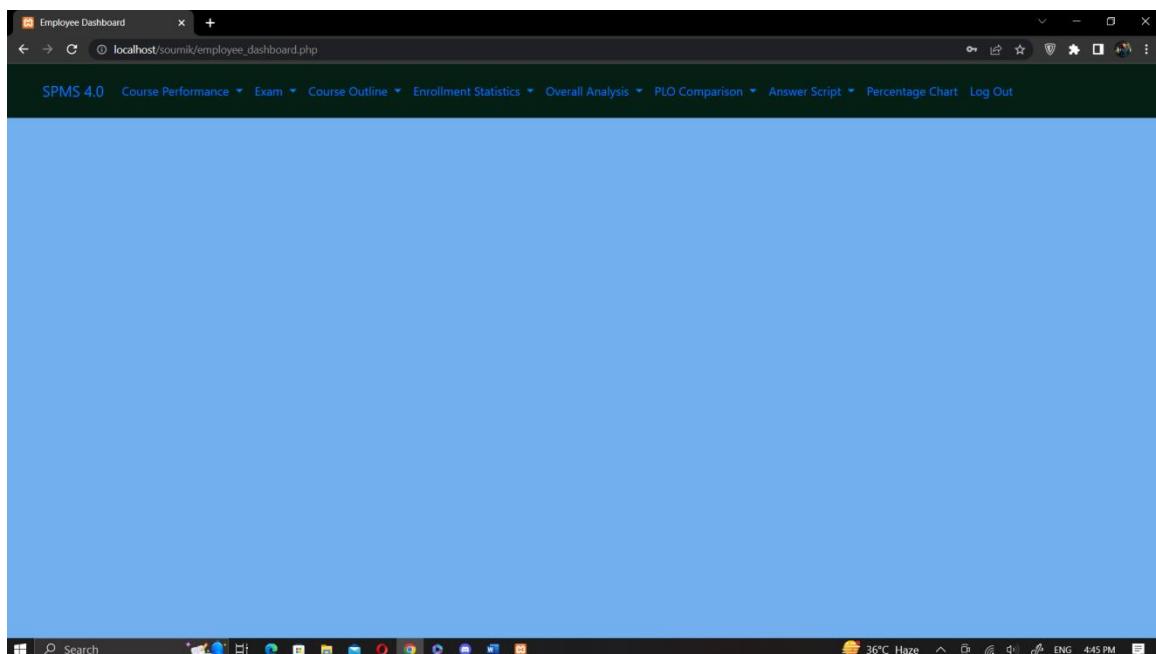
<!-- jQuery Library -->
<script src="https://code.jquery.com/jquery-3.6.0.min.js" integrity="sha384-DfXdz2htPH0lsSS5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+OrCxarKfj"
crossorigin="anonymous"></script>

```

```
<script>
    function formToggle(ID){
        var element = document.getElementById(ID);
        if(element.style.display === "none"){
            element.style.display = "block";
        }else{
            element.style.display = "none";
        }
    }
</script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/js/bootstrap.bundle.min.js"
integrity="sha384-ENjd04Dr2bkBIFxQpeoTz1HICje39Wm4j0Kdf19U8gt4ddQ3GYNs7NTkfAdVQSze" crossorigin="anonymous"></script>

</body>
</html>
```



Employee Dashboard

localhost/soumik/percentageChart.php

Menu

Student Course Performance

studentID: 1531176

sectionNum: 2

semester: Autumn

courseID: CSE303

year: 2023

obtainGrade: A

Submit

```
1 <?php
2
3 // Include database connection and header files
4 // require_once(_DIR_."/testfunction.php");
5 // require_once(_DIR_ . "/connect.php");
6 require_once "connect.php";
7 // require_once(_DIR_ . "/user_header.php");
8 $result = false;
9 $grade_mapping = [
10     "A" => 90,
11     "A-" => 85,
12     "B+" => 80,
13     "B" => 75,
14     "B-" => 70,
15     "C+" => 65,
16     "C" => 60,
17     "C-" => 55,
18     "D+" => 50,
19     "D" => 45,
20     "F" => 43,
21 ];
22 $grade_point_mapping = [
23     "A" => 4.0,
24     "A-" => 3.7,
25     "B+" => 3.3,
26     "B" => 3.0,
27     "B-" => 2.7,
28     "C+" => 2.3,
29     "C" => 2.0,
30     "C-" => 1.7,
31     "D+" => 1.3,
32     "D" => 1.0,
33     "F" => 0.0,
34 ];
35
```

```

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if(isset($_POST["studentID"]) && $_POST["sectionNum"] && isset($_POST["semester"]) && isset($_POST["courseID"]) && isset($_POST
    ["year"]) && isset($_POST["obtainGrade"])){
        // && isset($_POST["coNum"])

        $studentID = $_POST["studentID"];
        $sectionNum = $_POST["sectionNum"];
        $semester = $_POST["semester"];
        $courseID = $_POST["courseID"];
        $year = $_POST["year"];
        $obtainGrade = $_POST["obtainGrade"];

        // Check if section already exists, if not insert it
        $sql = "INSERT INTO section_t (sectionNum, semester, courseID, year)
                SELECT * FROM (SELECT '$sectionNum', '$semester', '$courseID', '$year') AS tmp
                WHERE NOT EXISTS (
                    SELECT sectionID FROM section_t WHERE sectionNum='$sectionNum' AND semester='$semester' AND courseID='$courseID' AND
                    year='$year'
                ) LIMIT 1";
        if ($con->query($sql) === TRUE) {
            echo "New section added successfully.";
        } else {
            echo "Error: " . $sql . "<br>" . $con->error;
        }

        // Get the sectionID for the inserted section
        $sql = "SELECT sectionID FROM section_t WHERE sectionNum='$sectionNum' AND semester='$semester' AND courseID='$courseID' AND
        year='$year'";
        $result = $con->query($sql);
        $row = $result->fetch_assoc();
        $sectionID = $row['sectionID'];
    }
}

```

```

// Insert into registration_t table
$sql = "INSERT INTO registration_t (sectionID, studentID) VALUES ('$sectionID', '$studentID')";

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

// Get the registrationID for the inserted registration
$sql = "SELECT registrationID FROM registration_t WHERE sectionID='$sectionID' AND studentID='$studentID'";
$result = $con->query($sql);
$row = $result->fetch_assoc();
$registrationID = $row['registrationID'];

// Insert into student_course_performance_t table
// $sql = "INSERT INTO student_course_performance_t (registrationID, obtainGrade) VALUES ('$registrationID', '$obtainGrade')";

// if ($con->query($sql) === TRUE) {
//     echo "New student_course_performance added successfully.";
// } else {
//     echo "Error: " . $sql . "<br>" . $con->error;
// }
$markObtained = $grade_mapping[$obtainGrade];
$totalMarksObtained = $grade_mapping[$obtainGrade];
$gradePoint = $grade_point_mapping[$obtainGrade];

$sql = "INSERT INTO student_course_performance_t (registrationID, totalMarksObtained, gradePoint, obtainGrade) VALUES
        ('$registrationID', '$totalMarksObtained', '$gradePoint', '$obtainGrade')";

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

```

```

$questionID = $con->insert_id;
$coNum = $con->insert_id;
// Insert data into question_t
$markPerQuestion = 100;
$sql = "INSERT INTO question_t (markPerQuestion,coNum) VALUES ( '$markPerQuestion',$coNum)";

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

```

```

$sql = "INSERT INTO answer_t (markObtained, registrationID, questionID) VALUES ('$markObtained', '$registrationID', '$questionID')";

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

// Fetch the data to display in the HTML table
$sql = "SELECT r.studentID, s.sectionNum, s.semester, s.courseID, s.year, scp.obtainGrade
FROM student_course_performance_t scp
JOIN registration_t r ON scp.registrationID = r.registrationID
JOIN section_t s ON r.sectionID = s.sectionID
WHERE r.studentID='$studentID'";

$result = $con->query($sql);

}

$con->close();

?>

```

```

<!DOCTYPE html>

<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <!-- Bootstrap CSS -->
    <!-- <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.4.1/dist/css/bootstrap.min.css" integrity="sha384-Vkoo8XACkqO1X4+Iw/ACqX1+wqK+qKxQJYIv4+QG4lEeBfWVjIvMjN" crossorigin="anonymous"> -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0-alpha3/dist/css/bootstrap.min.css" rel="stylesheet" integrity="sha384-KK94CHFLLe+nY2dmCWGMq91rCGa5gtU4mk92HdvYeM/SXH301p5ILy+dN9+nJOZ" crossorigin="anonymous">

    <title>Employee Dashboard</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" > -->
    <!-- ===== CSS ===== -->
    <link rel="stylesheet" href="styleplo.css">
    <!-- <link rel="stylesheet" href="style3.css" > -->

    <!-- ===== Boxicons CSS ===== -->
    <link href="https://unpkg.com/boxicons@2.1.1/css/boxicons.min.css" rel='stylesheet'>

    <script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
    <script type="text/javascript"></script>

</head>
<body>
    <div class="d-flex justify-content-between">
        <div>
            
            <h1>Employee Dashboard</h1>
        </div>
        <div>
            <button>Logout</button>
        </div>
    </div>
    <div class="row">
        <div class="col-3">
            <div>
                <h3>Employee Information</h3>
                <table border="1">
                    <tr>
                        <td>Name:</td>
                        <td>John Doe</td>
                    </tr>
                    <tr>
                        <td>Position:</td>
                        <td>Manager</td>
                    </tr>
                    <tr>
                        <td>Email:</td>
                        <td>johndoe@example.com</td>
                    </tr>
                </table>
            </div>
        </div>
        <div class="col-9">
            <div>
                <h3>Performance Metrics</h3>
                <table border="1">
                    <thead>
                        <tr>
                            <th>Metric</th>
                            <th>Value</th>
                        </tr>
                    </thead>
                    <tbody>
                        <tr>
                            <td>Attendance Rate</td>
                            <td>95%</td>
                        </tr>
                        <tr>
                            <td>Completion Rate</td>
                            <td>85%</td>
                        </tr>
                        <tr>
                            <td>Employee Satisfaction</td>
                            <td>4.2/5</td>
                        </tr>
                    </tbody>
                </table>
            </div>
        </div>
    </div>
</body>

```

```
<style>
  body{
    background-color:#71afee;
    /* background-color:#155977; */
  }

  ::placeholder{
    color:white;
  }

  ::-ms-input-placeholder{
    color:white;
  }

  :-ms-input-placeholder{
    color:white;
  }

</style>

</head>

<body>

<nav class="sidebar close">

<header>
  <div class="image-text">

  </div>

  <i class='bx bx-chevron-right toggle'></i>
</header>
```

```
<header>
  <div class="image-text">

  </div>

  <i class='bx bx-chevron-right toggle'></i>
</header>

<div class="menu-bar">
  <div class="menu">

    <ul class="menu-links">
      <li class="nav-link">
        <a href="employee_dashboard.php">
          <i class='bx bx-home-alt icon' ></i>
          <span class="text nav-text">Dashboard</span>
        </a>
      </li>
    </ul>
  </div>
```

```
</div>

</nav>

<div style="margin-top: 50px;" class="container">
    <h2>Student Course Performance</h2>
    <div class="col-xs-12">
```

```
<div class="row">
    <div class="col-xs-12">
        <table class="table table-striped table-bordered">
            <thead class="thead-dark">
                <tr>
                    <th>Student ID</th>
                    <th>Section Number</th>
                    <th>Semester</th>
                    <th>Course ID</th>
                    <th>Year</th>
                    <th>Obtain Grade</th>
                    <th>Co</th>
                    <th>Co1</th>
                    <th>Co2</th>
                    <th>Co3</th>
                    <th>Co4</th>
                </tr>
            </thead>
```

```
<?php
if($result !== false){
    if($result->num_rows > 0){

        while ($row = $result->fetch_assoc()) {
            echo "<tr>
                <td>" . $row['studentID'] . "</td>
                <td>" . $row['sectionNum'] . "</td>
                <td>" . $row['semester'] . "</td>
                <td>" . $row['courseID'] . "</td>
                <td>" . $row['year'] . "</td>
                <td>" . $row['obtainGrade'] . "</td>
                <td>" . $grade_mapping[$row['obtainGrade']] . "%</td>
                <td>" . $grade_mapping[$row['obtainGrade']] . "%</td>

            </tr>";
        }
    }
}
?>
```

```
        </table>
    </div>
</div>
</div>
```

```
<!-- overall plot -->
<script>
</script>

<!-- Co wise plot -->
<script>
function coWisePlot(){
    ...
}
</script>

<!-- course wise plot -->
<script>
function courseWisePlot(){
    ...
}
</script>
</body>
</html>

<!-- Bootstrap library -->
<script src="https://cdn.jsdelivr.net/npm/bootstrap@3.3.7/dist/js/bootstrap.min.js"
integrity="sha384-Tc5IQib0ZJiXpuán1UQFt4WxPZlF17FbE7Xn3Z8EN6A2hRocjE1C27zWVNXgq" crossorigin="anonymous"></script>

<!-- jQuery library -->
<script src="https://code.jquery.com/jquery-3.6.0.min.js" integrity="sha384-Dfxz2htPH0lSS5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+0rCxarKfj" crossorigin="anonymous"></script>
```

```
<!-- jquery library -->
<script src="https://code.jquery.com/jquery-3.6.0.min.js" integrity="sha384-Dfxz2htPH0lSS5nCTpuj/zy4C+OGpamoFVy38MVBnE+IbbVYUew+0rCxarKfj" crossorigin="anonymous"></script>

<script>
    function formToggle(ID){
        var element = document.getElementById(ID);
        if(element.style.display === "none"){
            element.style.display = "block";
        }else{
            element.style.display = "none";
        }
    }
</script>
</body>
</html>
```

Employee Dashboard x +

localhost/soumik/percentageForm.php

New section added successfully.New registration added successfully.New student_course_performance added successfully.New question added successfully.New answer added successfully.

> [Dashboard](#)

Student Course Performance

Student ID	Section Number	Semester	Course ID	Year	Obtain Grade	Co	Co1	Co2	Co3	Co4
2030479	2	Autumn	CSE303	2023	A	90%	90%	90%	90%	90%
2030479	2	Autumn	CSE303	2023	A	90%	90%	90%	90%	90%
2030479	5	Spring	CSE303	2023	B+	80%	80%	80%	80%	80%
2030479	9	Spring	CSE303	2023	C	60%	60%	60%	60%	60%
2030479	9	Spring	CSE303	2023	D+	50%	50%	50%	50%	50%
2030479	9	Spring	CSE303	2023	B-	70%	70%	70%	70%	70%

Windows Search Earnings upcoming ENG 4:58 PM

CHAPTER 5 - CONCLUSION:

A. PROBLEM AND SOLUTION:

Analysis Phase

Due to the lack of discrete data, the majority of the work was done based on project SPM developers' assumptions and questions when developing the rich picture and six element analysis of the organization's activities. For a clearer grasp of the situation and to avoid similar confusing situations, reputable academics, and stakeholder interviews were conducted.

Designing Phase

The Relational Schema design also included the retention of created entities at their Respective levels based on descriptive study. The instructor's comments was also highly important and valid in this situation.

Implementation Phase

All the Software System Requirement's (SSR's) reached successfully!

Front-End Developing tools: HTML, CSS, Bootstrap JavaScript, Bootstrap, Chart Js

Back End Developing tools: PHP, JSON

Database-integration: MySQL

B. ADDITIONAL FEATURE AND FUTURE DEVELOPMENT:

Future Developing Purposes:

- As we have made this project as 2-tier Architecture. Plans for the project is, to add new feature such as 3-tier Architecture which can this project more scalable, maintainable, and flexible enough to adapt to changing requirements.
- Deployment.

References

- [1] <http://www.iub.edu.bd/AboutIUB/ataglance>. [Online].
- [2] <http://www.cse.iub.edu.bd/degrees/1>. [Online].
- [3] <https://mspguide.org/2022/03/18/rich-picture/#:~:text=What%20is%20a%20Rich%20Picture>. [Online].

[4] <https://www.trisotech.com/bpmn-introduction-and-history/#:~:text=BPMN%20was%20originally%20developed%20by>. [Online].