

Database Management Project

Final Report

Group 12

|  |  |
| --- | --- |
| NAME | ID |
| ASIFUR RAHMAN SHIKDAR | 1921770 |
| NIHAD TANFIZ | 1920301 |
| SADNAN HAQUE | 1920370 |
| FUAD HOSSAIN FAHIM | 2010346 |
| SYED SOWAD EPHRAIM ALI | 1911034 |
| ABDULLAH MOHAMMAD AL MATIN | 1910980 |

Table of Contents

[CHAPTER-1 INTRODUCTION](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.gjdgxs)

1. [BACKGROUND OF THE ORGANIZATION - IUB](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.30j0zll)
2. [BACKGROUND OF THE PROJECT - SPMS 4.0:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.1fob9te)
3. [OBJECTIVE OF THE PROJECT - SPMS 4.0](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.3znysh7)
4. [SCOPE OF THE PROJECT:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.2et92p0)

[CHAPTER-2 REQUIREMENT ANALYSIS](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.tyjcwt)

1. METHODOLOGY
2. [RICH PICTURE – EXISTING SYSTEM (SPMS 3.0)](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.3dy6vkm)
3. [SIX ELEMENT ANALYSIS – EXISTING SYSTEM (SPMS 3.0)](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.1t3h5sf)
4. [PROBLEM ANALYSIS – EXISTING SYSTEM (SPMS 3.0)](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.3j2qqm3)
5. [RICH PICTURE – PROPOSED SYSTEM (SPMS 4.0)](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.1y810tw)
6. [SIX ELEMENT ANALYSIS – PROPOSED SYSTEM (SPMS 4.0)](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.4i7ojhp)

[CHAPTER-3 LOGICAL SYSTEM DESIGN](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.32hioqz)

1. [BUSINESS RULES – SPMS 4.0](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.1hmsyys)
2. [ENTITY RELATIONSHIP DIAGRAM](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.41mghml)
3. [ENTITY RELATIONSHIP DIAGRAM TO RELATIONAL SCHEMA](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.2grqrue)
4. [NORMALIZATION](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.vx1227)

[1NF:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.3fwokq0)

[2NF:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.1v1yuxt)

[3NF:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.4f1mdlm)

1. [DATA DICTIONARY](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.2u6wntf)

[CHAPTER-4 PHYSICAL SYSTEM DESIGN](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.19c6y18)

1. [INPUT FORMS](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.3tbugp1)
2. [OUTPUT FORMS](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.28h4qwu)

[CHAPTER-5 CONCLUSION](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.nmf14n)

1. [A PROBLEM AND SOLUTION](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.37m2jsg)
2. [Analysis Phase:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.1mrcu09)
3. [Designing Phase:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.46r0co2)
4. [Implementation Phase:](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.2lwamvv)
5. [Additional Features and Future Development](https://docs.google.com/document/d/1AmLniEBfdSZCbz6Y__AUFohbUU_wOilB/edit#heading=h.111kx3o)

CHAPTER-1 INTRODUCTION

1. BACKGROUND OF THE ORGANIZATION - IUB:

IUB is a private university in Bangladesh which was established in 1993. It has a current enrollment of around 10 000 at undergraduate and graduate levels [1]. More than 450 highly qualified and highly-skilled faculty members who are exceptionally good at their department of teaching and where at least half of them are PhD degree holders [2]. IUB also has an alumni strength of nearly 14000 people. There are currently 5 academic schools in IUB [1]

1. School of Business & Entrepreneurship
2. School of Engineering, Technology and Sciences
3. School of Environment & Life Sciences
4. School of Liberal Arts & Social Sciences
5. School of Pharmacy and Public Health

1. BACKGROUND OF THE PROJECT - SPMS 4.0:

Student Performance Monitoring System (SPMS 4.0) is a framework for Outcome-Based Education (OBE). It evaluates the performance of students, course instructors, schools, departments and programs and helps the Higher Authorities of the education institution to make strategies for improvements. It allowed us to provide a system to Monitor a students Course Outcomes and Course Learning Outcomes with their Grades published in order to show a students academic profile.

1. OBJECTIVE OF THE PROJECT - SPMS 4.0:

SPMS 4.0 monitors and analyzes the performance of its stakeholders such as Students, Course instructors, Departments, Schools, Programs through the database of assessments such as quizzes, midterm exams, final term exams etc. In order to evaluate the performance of the stakeholders, SPMS 4.0 stores necessary documents and data in the database such as all the exam question papers, answer scripts, course outlines and marks of the exams and assessments with respect to their Course Outcomes (CO), Program Learning Outcomes (PLO) and Program Outcomes (PO) achieved by the students. Hence, students can statistically monitor their own performance. SPMS 4.0 also creates opportunities for Higher Authorities to draw conclusions and make further

improvements by providing them with a wide range of analytical reports based on the performance of students, course instructors, departments, schools, and programs.

1. SCOPE OF THE PROJECT:

We have done a complete analysis of the existing system (SPMS 3.0) and identified some issues in the business processes which can cause the process to become slow, inefficient and cause lapses in communication.

The proposed solution to overcome those issues is to create a more improved version of the system called SPMS 4.0 (Student Performance Monitoring System 4.0) which uses a Relational Database Management System (RDBMS) to store, update and retrieve necessary documents such as Course Outlines, Exam Question Papers and Answer Scripts as well as other necessary data required to monitor student performance and produce other OBE (outcome-Based Education) reports.

We have identified all the users of the system (SPMS 4.0), how they would be accessing the necessary data and information and how they would interact with each other etc.

We want to build interfaces for all the users to be able to access their required data and generate, view, and download their desired reports and documents using the system.

We also want to build an interface for the course instructors/faculties to be able to collaborate with each other on developing course outlines, exam papers, marksheets etc.

# CHAPTER-2 REQUIREMENT ANALYSIS

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

METHODOLOGY:

This is how our STUDENT PERFORMANCE MONITORING SYSTEM database management system operates:

1. Specify the project's requirements and scope:

a. Determine the types and sources of the data needed for the analysis.

b. Establish the method for calculating the students' performance score.

c. Describe all of our user needs, such as creating reports.

2. Create a schema for the database:

a. List the entities on which to base the ERD.

b. Pretend that there are relationships among the entities, such as One to

Many or Many to Many.

c. Describe the characteristics of each entity.

3. Create a database:

a. Build the database table based on the schema, then load the data into

it.

b. List the data restrictions, such as primary keys and foreign keys.

4. Put data processing and analysis into practice by creating queries to extract and process the necessary data, using statistical analysis to produce performance metrics, and creating reports.

5. Validate the results with users to make sure the project satisfies the requirements:

a. Validate the results with users to make sure the project satisfies the

requirements.

b. Refine the database schema and analysis as necessary based on

testing results.

6. Deploy the database and analysis. This includes:

a. deploying the database and analysis to the production environment;

b. educating end users on how to use the database and analysis;

c. providing support and maintenance to the project to ensure its

success.

## RICH PICTURE – EXISTING SYSTEM (SPMS 3.0):

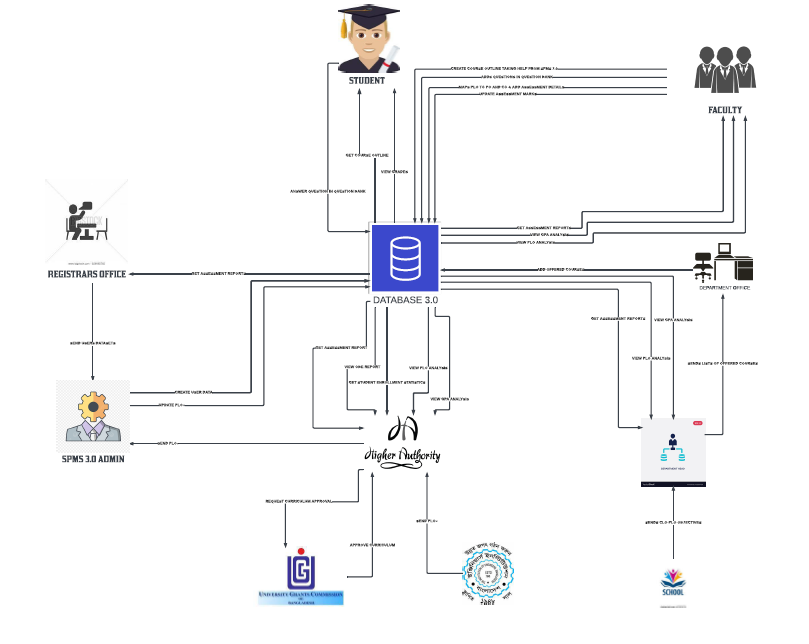
A rich picture is a way to demonstrate processes in a system which is easier to understand for everyone. It consists of pictures text, symbols and icons which are all used to illustrate graphically the situation. [3] A rich picture helps us to see relationships and connections that we may otherwise miss [3]. It helps identifying one or more themes participants may want to further explore and address. Rich pictures are therefore always used in the pre-analysis phase [3].

In this rich picture the stakeholders are:

1. UGC
2. IEB
3. Higher Authority (VC, Dean etc)
4. Department Head
5. Department Office
6. SPMSV3.0 Admin (SPMS Manager)
7. Registers Office
8. Faculty
9. Student
10. School

The Main Storages are

1. SPMS V3.0



## SIX ELEMENT ANALYSIS – EXISTING SYSTEM (SPMS 3.0)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| PROCESS | HUMAN | NON-COMPUTING  HARDWARE | COMPUTING HARDWARE | SOFTWARE | DATBASE | NETWORKING &  COMMUNICATION |
| Trend in academic success based on grade point average. | Department Head:  1) Enters their ID and password to log in to the system.  2) Enter the viewing time frame and course ID.  3) Examine student development using a graph created after analysis and the GPAs attained by the highest/lowest/average number of students.  Faculty:  1) Enters their ID and password to log in to the system.  2) Use the course ID and time period to find the subject they are teaching and view the students' progress in that course.  Student:  1) Enters their ID and password to log in to the system.  2) Use the course ID to find the subject and view the student's progress in it as well as their grade point average.  Dean/VC:  1) Enters their ID and password to log in to the server.  2) Use the course ID and time period to search for the course, then view the students' progress in that subject. | Paper and a Pen:  1) Possibility of developing rough assessment queries.  2) Employed to respond to evaluation inquiries. | 1) The laptop/computer used to sign into SPMS 3.0.  2) Printers used to make a hard copy of the students' progress for the current semester so that they can compare it to the progress of the students who took that course the semester before. | 1) Used to create performance analysis graphs using database data or store pupil data in the database, SPMS3.0 | 1) SPMS3.0 Database All current information is kept here and can be changed by SPMS3.0 administrators. | INTERNET:  1)used to sign into SPMS 3.0 via Internet. |
| GPA-based student achievement as determined by the faculty | Faculty:  1) Enters their ID and password to log in to the server.  2) Track the Development of the Learners under their Instruction.  Department Head: 1) Enters their ID and password to log in to the system.  2) Use the school's name to look up a school that needs to be evaluated.  3) Based on the GPAs that the students have achieved, view the progress of the students who are being taught by that faculty.  Dean/VC: 1) Enters their ID and password to log in to the server.  2) Use the Faculty name and Department ID to find a faculty member who needs to be evaluated.  3) Based on the GPAs that the students have achieved, view the progress of the students who are being taught by that faculty. |  | 1) The laptop/computer used to sign into SPMS 3.0.  2) Printers used to make a hard copy of the students' progress for the current semester so that they can compare it to the progress of the students who took that course the semester before. | 1) Used to create performance analysis graphs using database data or store pupil data in the database, SPMS3.0 | 1) SPMS2.0 Database All current information is kept here and can be changed by SPMS3.0 administrators. | INTERNET:  1)used to sign into SPMS 3.0 via Internet. |
| Course-wise PLO achievement of a student | VC/Dean: 1) Enters their ID and password to log in to the server.  2) Click the PLO award tab and enter the course ID.  3) Review the student's PLO accomplishments.  Department Head: 1) Enters their ID and password to log in to the system.  2) Click the PLO award tab and enter the course ID.  3. Examine the PLOs that the pupils met.  Faculty: 1) Enters their ID and password to log in to the system.  2) Click the PLO award tab and enter the course ID.  3) Check out the PLOs that the pupils in a course have met.  Student: 1) Enters their ID and password to log in to the server.  2) Review the PLOs they have already met and the number they still need to reach to pass the training. |  | 1) The laptop/computer used to sign into SPMS 3.0.  2) Printers used to make a hard copy of the students' progress for the current semester so that they can compare it to the progress of the students who took that course the semester before. | 1)It is utilized to keep data and produce PLO automatically based on the CO given. | 1) SPMS2.0 Database All current information is kept here and can be changed by SPMS3.0 administrators. | INTERNET:  1)used to sign into SPMS 3.0 via Internet. |
| Trend in student success under the VC, Dean, and Department Head | Dean: 1) Enters their ID and password to log in to the server.  2) Using their Name and Department ID, look up the Department Head to be verified.  3) Track students' development beneath them or them.  1) User logs in to server with ID and password.  2) Using their name and either their school ID or department ID, search for a dean or department head to be verified.  3) Track pupil development beneath them.  Department Head: 1) Enters their ID and password to log in to the system.  2) Track students' development beneath them. |  | 1) The laptop/computer used to sign into SPMS 3.0.  2) Printers used to make a hard copy of the students' progress for the current semester so that they can compare it to the progress of the students who took that course the semester before. | 1) Used to create performance analysis graphs using database data or store pupil data in the database, SPMS3.0 | 1) SPMS3.0 Database All current information is kept here and can be changed by SPMS3.0 administrators. | INTERNET:  1)used to sign into SPMS 3.0 via Internet. |
| Statistics of Course, Program, department, schools CLO-PLO | Dean/VC:  1) Enters their ID and password to log in to the server.  2) Check out the CLO-PLO mapped student achievement data.  Department Head: 1) Enters their ID and password to log in to the system.  2) Check out the CLO-PLO mapped student achievement data.  Faculty: 1) Enters their ID and password to log in to the system.  2) Check out the CLO-PLO mapped student achievement data.  Student: 1) Enters their ID and password to log in to the server.  2) Examine the CLO-PLO mapped statistics that they and other students accomplished. |  | 1) The laptop/computer used to sign into SPMS 3.0.  2) Printers used to make a hard copy of the students' progress for the current semester so that they can compare it to the progress of the students who took that course the semester before. | SPMS2.0 1) Used to create CLO-PLO statistical data or graphs and keep data in databases | 1) SPMS2.0 Database All current information is kept here and can be changed by SPMS3.0 administrators. | INTERNET:  1)used to sign into SPMS 3.0 via Internet. |
| Course, student, and faculty expectations vs. PLO results | Dean/VC:  1) Logs in to the server using their ID and password  2) Compare the expected and achieved PLO results for the students during the period that has been entered.  Department Head:  1) Log in to the system using your ID and password,  2) Compare the expected and achieved PLO results for the students during the period that has been entered.  Faculty:  1) Log in to the system with your ID and password,  2) Compare the expected and achieved PLO results for the students during the period that has been entered.  Student:  1) Log in to the system with your ID and password.  2) Compare the expected and achieved PLO results for the students during the period that has been entered. |  | 1) The laptop/computer used to sign into SPMS 3.0.  2) Printers used to make a hard copy of the students' progress for the current semester so that they can compare it to the progress of the students who took that course the semester before. | SPMS2.0 1) Used to create CLO-PLO statistical data or graphs and keep data in databases | 1) SPMS2.0 Database All current information is kept here and can be changed by SPMS2.0 administrators. | INTERNET:  1)used to sign into SPMS 3.0 via Internet. |
| Department average of total PLO results and students attempted | Dean/VC:  1) Enter your ID and password to log in.  2) Select the semester's desired viewing time frame.  3) View the overall PLO average for the department as well as the number of students who tried.  Department Head:  1) Log in to the system using your ID and password.  2) Select the semester's desired viewing time frame. 3) View the overall PLO average for the department as well as the number of students who tried.  FACULTY:  1) Log in to the system with your ID and password.2) Check out the pupils' PLO results overall department average.  Student:  1) Log in to the system with your ID and password.2) See the pupils' overall departmental average of PLO results. |  | 1) The laptop/computer used to sign into SPMS 3.0.  2) Printers used to make a hard copy of the students' progress for the current semester so that they can compare it to the progress of the students who took that course the semester before. | SPMS2.0 1) Used to create CLO-PLO statistical data or graphs and keep data in databases | 1) SPMS2.0 Database All current information is kept here and can be changed by SPMS3.0 administrators. | INTERNET:  1)used to sign into SPMS 3.0 via Internet. |
| Assessments  And grading | Faculty:  1) Create Assessment with respect to course outline.  2) Give Assessment to students.  3) Grade the Answer Script papers returned by students.  4) Store Grades to SPMSV2.0  Student:  1) Receive Assessment from Faculty.  2) Complete and Return the Answer Scripts to the respective Faculty.  3)Get Grades from SPMSV3.0 when available | Paper and Pen:  1) Possibility of developing rough assessment queries.  2) Employed to respond to evaluation inquiries. | 1)Computer/Laptop: To Print Using MS Office  2)Printer: Used, if necessary, to make hard copies of the assessments | MS OFFICE was used to 1) construct the assessment and 2) store the grades in a spreadsheet. | used to store all of the student-returned response scripts | Internet:  1) Used for Google Classroom sign-ins when contact is necessary.  2) Used for SPMS 3.0 Sign-In |
| Statistics on student enrollment by VC, Dean, and Department Head. | VC/Dean:  1) Enter your ID and password to log in.  2) Under the Student Enrollment Statistics option, choose Year and Semester.  3) Check out the student enrollment figures for that semester and year.  Head of Department:  1) Enter your ID and password to log in.  2) Under the Student Enrollment Statistics option, choose Year and Semester.  3) Check out the student enrollment figures for that semester and year. |  | Computer/Laptop:  1) Used to sign into SPMS 3.0  Printer:  1) Used to print the hard duplicate of enrollment statistics for students, if necessary. | SPMS2.0 1) Used to create graphs of student enrollment statistics and keep data in the database. | 1) SPMS2.0 Database All current information is kept here and can be changed by SPMS3.0 administrators. | INTERNET:  Used for SPMS 3.0 Sign-In |
| Making and preserving the course outline | Faculty:  1) Contact the department office to request course materials from the previous semester.  2) Produce a course syllabus  3) Keep the course material in actual storage.  4) Hand the students the course outline.  Department Office:  1) Distribute course materials to students from the prior  Semester.  Students:  1) Receive the faculty's course  outline | Pen & Paper:  1) Used to draft an approximate course outline Overview | **Computer/Laptop:**  2)To Use MS OFFICE  **Printer:**  1) Used to print Hard Copy Of Course Outline | MS OFFICE was used to  1) construct the assessment and 2) store the grades in a spreadsheet. | Physical Storages:  For storing Course Outlines. | INTERNET:  Used for SPMS 3.0 Sign-In |

## PROBLEM ANALYSIS – EXISTING SYSTEM (SPMS 3.0)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process name | Stake holders | Concerns (probs) | Analysis (reasons of probs) | Proposed solution |
| Assessments & Grading | Faculty:  Student: | 1) The state of the question paper and answer script  2) Process of Giving and Receiving  3) Insecure Storage  4) Lack of awareness of Question Difficulty and Domain of Learning | 1) The question papers and answer scripts that are kept in tangible storage could be misplaced or damaged.  2) The process for finishing the assessment and handing it in person to the instructor is cumbersome.  3) Physical storage may rapidly run out of room as more answer scripts are submitted.  4) It requires a lot of effort to manually determine the question's difficulty and learning domain. | There are no issues with physical storage room or the condition of the paper because the answer scripts and question paper are both stored in the database. A question's difficulty level and learning topic are automatically given based on the bloom's taxonomy once it is added to the question bank. Time is saved by submitting assessments online rather than in person because there is no need to submit a hard duplicate. |
| Storing & Preparing course outline: | Department Office:  Faculty:  Student: | 1) A delay in getting the required resources  2) Creating a Course Map  3) The State of the Curriculum  4) The Process of Giving and Receiving | 1) Faculty must submit a request to the department office and wait for them to send the course materials from the previous term.  2) Manually creating a course outline requires a lot of time.  3) The physical storage where the course outlines are kept could become misplaced or damaged.  4) The distribution of course outlines to students requires time. | The Faculty can add items to their course outline, and SPMSV3 can instantly create course outlines that include those items. There is no issue with physical storage capacity or the condition of the paper because it is saved in the database. The students can obtain a course outline as soon as it is created, ensuring immediate access to the outlines. |

## RICH PICTURE – PROPOSED SYSTEM (SPMS 4.0):­­­­­­

## 

## SIX ELEMENT ANALYSIS – PROPOSED SYSTEM (SPMS 4.0)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| PROCESS | HUMAN | NON-COMPUTING  HARDWARE | COMPUTING HARDWARE | SOFTWARE | DATBASE | NETWORKING &  COMMUNICATION |
| Trend in academic success based on grade point average. | Department Head:  1.Enters their unique identification and password to log into the system.  2.Specifies the desired course identification number and viewing period.  3.Reviews students' progress by analyzing the data and using a graphical representation that shows the grade point averages of the top, bottom, and average performers.  Faculty:  1.Logs into the system using their identification and password.  2.Searches for the course they teach by entering the course identification number and reviews the students' progress in that course within the specified time period.  3.Uploads Grades on the database.  Student:  1.Logs into the system using their personal identification and password.  2.Locates the desired course using the course identification number and views their progress as well as their grade point average in the subject.  Dean/VC:  1.Logs into the server using their unique identification and password.  2.Searches for a specific course by entering the course identification number and the desired time period, and reviews the students' progress in that subject. | Paper and Pen:  1.Can be utilized to draft initial assessment questions.  2.Used to answer assessment inquiries. | 1.The device, either a laptop or a computer, used for logging into SPMS 4.0.  2.Printers utilized to produce a physical copy of the students' progress during the current semester, enabling them to compare it with the progress of the previous semester's students who took the same course. | 1.Employed for generating performance analysis charts by utilizing data from the database or storing student information in the SPMS 4.0 database. | 1.The SPMS4.0 database stores all current information, now also includes grades, Co percentage and displaying it, which is editable only by SPMS4.0 administrators. | INTERNET:  1) Employed for logging into SPMS 4.0 through the internet. . |
| GPA-based student achievement as determined by the faculty | Faculty:  1.Logs into the system by entering their identification and password.  2.Monitors the progress of the students under their instruction.  Department Head:  1.Logs into the system using their unique identification and password.  2.Searches for a school requiring evaluation by entering the school's name.  3.Reviews the progress of the students being taught by that faculty based on their GPAs.  Dean/VC:  1.Logs into the server by entering their identification and password.  2.Searches for a faculty member requiring evaluation by entering their name and department ID.  3.Reviews the progress which is grades and CO outcomes of the students being taught by that faculty based on their GPAs. | Paper and Pen:  1.Can be utilized to draft initial assessment questions.  2.Used to answer assessment inquiries. | 1.The device, either a laptop or a computer, utilized for logging into SPMS 4.0.  2.Printers employed for producing a physical copy of the current semester's students' progress, enabling comparison with the progress of the students who took the same course in the previous semester. | 1) The creation of performance analysis graphs using data from the database or storing student information in the SPMS 4.0 database is facilitated by this tool. | 1) The SPMS 4.0 database is a repository for all current information, which can be modified only by SPMS 4.0 administrators. | INTERNET:  1Employed for logging into SPMS 4.0 through the internet. |
| Course-wise PLO achievement of a student | VC/Dean:  1.Login to the server by entering their unique identification and password.  2.Access the PLO award tab and enter the course ID.  3.Review the student's achievements with respect to the Program Learning Outcomes.  Department Head:  1.Login to the system using their identification and password.  2.Access the PLO award tab and enter the course ID.  3.Review the Program Learning Outcomes that the students have met.  Faculty:  1.Login to the system using their identification and password.  2Access the PLO award tab and enter the course ID.  3.Review the Program Learning Outcomes that the students in their course have achieved.  Student:  1.Login to the server using their personal identification and password.  2.Review the Program Learning Outcomes they have already achieved and the number they still need to attain to pass the course. |  | 1.The device utilized to access SPMS 4.0 is a laptop or computer.  2.Printers are utilized to generate a physical copy of students' progress for the current semester, which can be compared to the progress of the students who took that course in the previous semester. | 1.The tool is used to store data and automatically generate Program Learning Outcomes (PLOs) based on the Course Outcomes (COs) provided. | 1) The SPMS4.0 database is where all current information is stored and managed by administrators with appropriate access. This includes data related to students, faculty, courses, and performance analysis, among other important records. Changes to this data can only be made by authorized SPMS4.0 administrators.  . | INTERNET:  1) The system can be accessed via a web browser and login credentials, such as a unique identification and password, are required to sign into SPMS 4.0 over the internet. |
| Trend in student success under the VC, Dean, and Department Head | Dean:  1.The user logs into the server with their login ID and password.  2.They use their name and either their school ID or department ID to search for the Department Head to be verified.  3.They track the students' progress under their department or under the department head.  Department Head:  1.The user logs into the system using their ID and password.  2.They track the students' progress under their department.  3.Students with CGPA 3.9 or above are awardees. |  | 1.The device used for accessing SPMS 4.0 via the internet.  2.The printers used to create physical copies of the current semester's progress reports for students, allowing them to compare their progress to that of students from previous semesters. | 1) This refers to the functionality of the SPMS4.0 system to create graphs that analyze student performance based on data stored in the database. It can also be used to store data on individual students within the system's database. | 1) The SPMS4.0 Database is the central location where all current information is stored and managed by SPMS4.0 administrators. | INTERNET:  1) The internet is used to access and sign into SPMS 4.0. . |
| Statistics of Course, Program, department, schools CLO-PLO | Dean/VC:  1.The Dean/VC logs into the server using their ID and password.  2.The Dean/VC reviews the data on student achievement that is mapped to CLO-PLO.  Department Head:  1.The Department Head enters their login credentials to access the system.  2.The Department Head reviews the data on student achievement that is mapped to CLO-PLO.  Faculty:  1.Faculty members enter their login credentials to access the system.  2.Faculty members review the data on student achievement that is mapped to CLO-PLO.  Student:  1.Students enter their login credentials to access the server.  2.Students examine the statistics on CLO-PLO mapping to see their own and other students' achievements. |  | 1.A computer or laptop is required to access SPMS 4.0.  2.Printers are utilized to generate a physical copy of the students' progress report for the current semester to enable comparison with the previous semester's progress report of students who took the same course. | SPMS2.0 1) The system is used to generate statistical data or graphs for CLO-PLO and store the data in the databases. | 1) The SPMS4.0 database serves as a repository for all the current information and can be modified by authorized SPMS4.0 administrators. | INTERNET:  1) SPMS 4.0 can be accessed and logged into through the internet using a laptop or computer. |
| Course, student, and faculty expectations vs. PLO results | Dean/VC:  1.Accesses the server by logging in with their unique ID and password.  2.Reviews and compares the expected and achieved PLO results for students during the specified period.  Department Head:  1.Accesses the system using their personal ID and password.  1.Reviews and compares the expected and achieved PLO results for students during the specified period.  Faculty:  1.Logs in to the system using their individual ID and password.  2.Reviews and compares the expected and achieved PLO results for students during the specified period.  Student:  1.Logs in to the system using their unique ID and password.  2.Reviews and compares their own expected and achieved PLO results with those of their peers during the specified period.  3.Reviews faculty too under whom the course was completed. |  | 1.The laptop or computer is used to access SPMS 4.0.  2.  Printers are utilized to produce a physical copy of students' progress for the current semester. This allows for a comparison to be made with the progress of students who took the same course in the previous semester. | SPMS2.01) CLO-PLO statistical data or graphs were generated using a tool, and the resulting data was stored in databases. | 1) The SPMS4.0 database contains all current information, and it can only be modified by administrators with access to the system. | INTERNET:  1) Previously, it was possible to access SPMS 4.0 via the internet using a login system. |
| Department average of total PLO results and students attempted | Dean/VC:  1.Use your ID and password to log in.  2.Select the desired semester time frame for viewing.  3.Review the overall PLO average for the department and the number of students who attempted to meet the PLOs.  Department Head:  1.Log in to the system using your personal ID and password.  2.Select the desired semester time frame for viewing.  3Review the overall PLO average for the department and the number of students who attempted to meet the PLOs.  Faculty:  1.Log in to the system using your unique ID and password.  2.Review the overall departmental average for PLO results achieved by students.  Student:  1.Log in to the system using your personal ID and password.  2.Review the overall departmental average for PLO results achieved by students. |  | 1A laptop or computer is used to access SPMS 4.0.  2.Printers are utilized to create a physical copy of students' progress for the current semester. This allows for a comparison to be made with the progress of students who took the same course in the previous semester. | SPMS2.0 1) A tool was used to generate statistical data or graphs for CLO-PLO mapping, and the resulting data was stored in databases. | 1) The SPMS4.0 database stores all the latest information and can be modified by administrators who have access to the system. | INTERNET:  1) Previously, it was possible to access SPMS 4.0 through the internet by logging in with appropriate credentials. |
| Assessments  And grading | Faculty:  1.Use the course outline to create an assessment.  2.Distribute the assessment to students.  3.Grade the answer scripts submitted by students.  4.Store grades in SPMSV2.0.  Student:  1.Receive the assessment from the faculty.  2.Complete the assessment and submit the answer scripts to the faculty.  3.Access grades via SPMSV4.0 when they become available. | Paper and Pen:  1. can be utilized to brainstorm and draft assessment questions.  2.Students can utilize paper and pen to respond to assessment questions during an exam. | 1.A computer or laptop running MS Office can be used to print assessments.  2.A printer can be utilized, if required, to produce hard copies of the assessments. | MS Office was utilized for the following purposes:  1.To create the assessment.  2.To store the grades in a spreadsheet. | A storage system was utilized to store all of the response scripts submitted by students. | Internet:  1) When contact is necessary, Google Classroom sign-ins are utilized.  SPMS 2.0 2. Sign-In is utilized for accessing the SPMS 2.0 platform. Used for Google Classroom sign-ins when contact is necessary.  2) Used for SPMS 2.0 Sign-In |
| Statistics on student enrollment by VC, Dean, and Department Head. | VC/Dean:  1.Use your login credentials to access the system.  2.Navigate to the Student Enrollment Statistics section and select the desired Year and Semester.  3.View the enrollment data for the selected time period.  Head of Department:  1.Use your login credentials to access the system.  2.Navigate to the Student Enrollment Statistics section and select the desired Year and Semester.  3.View the enrollment data for the selected time period. |  | Computer/Laptop:  1. tool allows users to access SPMS 4.0 by logging in through the internet.  Printer 1.can be used to obtain a physical copy of the enrollment statistics for students, which can be generated from the database. | SPMS4.0 1) A data analysis and visualization tool can be used to create graphs and analyze student enrollment statistics. The data is stored in the database for future reference and analysis. . | 1) The SPMS4.0 database stores all the up-to-date information and can be edited by the administrators of SPMS4.0. | INTERNET:  This is the method for logging in to SPMS 4.0. |
| Making and preserving the course outline | Faculty:  1.Request course materials from the department office for the previous semester.  2.Develop a course syllabus and provide it to students.  3.Store course materials as necessary.  Department Office:  1.Provide course materials from the previous semester to faculty for their use.  Students:  2.Receive the course outline from their faculty. | Pen & Paper:  1) This tool is employed to create a rough draft or an initial outline of a course. | **Computer/Laptop:**  1)To Use MS OFFICE  **Printer:**  1) Used to print Hard Copy Of Course Outline | MS OFFICE was used to 1) construct the assessment and 2) store the grades in a spreadsheet. | Physical Storages:  This is used to store and manage course outlines. | INTERNET:  This tool is utilized to log in to the SPMS 2.0 system. |

# CHAPTER-3 LOGICAL SYSTEM DESIGN

## BUSINESS RULES – SPMS 4.0:

* 1. A student must have one department. A STUDENT has StudentID, FirstName, LastName, dateOfBirth, gender, email, phone, address, departmentID, programID, enrollmentYear, enrollmentSemester, password. A department must have one or many Students.

1. Student may perform many registrations. A REGISTRATION includes RegistrationID, sectionID, studentID. A registration must be performed by at least one student.

1. A section mandatorily has many registrations. A registration has at least one section. A section includes sectionID, sectionNum, courseID, facultyID, year.

1. A registration may belong to many EVALUATIONS. An evaluation mandatorily belongs to one registration. An EVALUATION contains evaluationID, examID, registrationID,totalMarks.
2. A CO must map with one PLO. A PLO’s must map with one or many CO’s. PLO includes ploID, ploNum, programID. CO must map with the grade provided by the faculty.

1. 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 course.

1. A program must belong to one department. A department must belong to one or many programs. A DEPARTMENT contains departmentID, departmentName, schoolID.

1. A department must contain one school. A SCHOOL must contain one or many departments. A school includes schoolID, schoolName.

1. An employee has four sub-type( Dean, Department Head, Faculty, VC). An EMPLOYEE includes employeeID,password, firstName, lastName.

1. A school must be run by exactly one. A dean must run exactly one school. A DEAN has schoolID, startDate, endDate.

1. A Department must be run by exactly one Department head. A department head must manage exactly one department. A DEPARTMENTHEAD includes departmentID, startDate, endDate.

1. A Faculty must have exactly one Department. A department must have one or many Faculties. A FACULTY includes departmentID, rank, joinDate. A faculty may teach many sections. A section must be taught by exactly one faculty.

1. A course outline belongs to exactly one section. A section must have exactly one course outline. A COURSE\_OUTLINE includes courseOutlineID ,sectionID, contactHours, courseDescription ,objective ,content ,refMaterials ,courseType, courseTitle ,prerequisiteCode ,creditValue.
2. A Course outline must have exactly one CLO Matrix. A CLO matrix belongs to exactly one course outline. A CLO\_MATRIX includes clo\_MatID, cloNum, coDescription, ploAssessed, correlation, courseOutlineID , c ,p ,a ,s.

1. A Lesson Plan Strategy must have exactly one Evaluation strategy. An Evaluation strategy must have exactly one Lesson Plan Strategy .A LESSON\_PLAN\_STRATEGY includes lPSID ,week ,topic ,learningStrategy , assessmentStrategy, correspondingClo, courseOutlineID.

1. An exam has exactly one evaluation. An Evaluation for an exam is done exactly once. An exam belongs to exactly one section. An EXAM includes examID, examName, sectionID. A section must have one or many exams.

1. An exam must have one or many questions. Every question must belong to exactly one exam. A QUESTION includes questionID , questionDetails , marksPerQuestion , questionNum , difficutltyLevel , examID , coNum. A Question is answered exactly once. An answer has exactly one question.

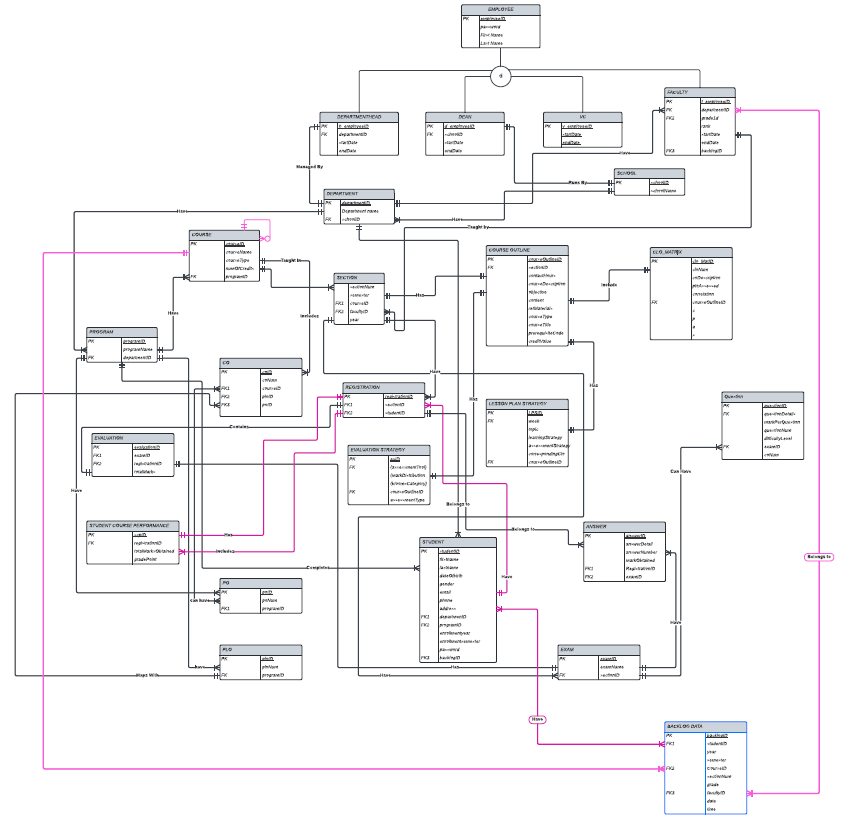
1. A PO belongs to exactly one program A program must have one or many PO.PO includes poID , poNum , programID. A PO must belong to one or many CO. A CO must have exactly one PO.

1. A student course performance evaluation is done for registration exactly once. A registration has student course performance evaluation done exactly once. A registration has exactly one evaluation. An Evaluation has exactly one registration.

20. Many faculties upload many grades. Many courses have many grades. Many students

has many grades. Primary key is GradeID, attribute is grade.

## ENTITY RELATIONSHIP DIAGRAM:



1. ENTITY RELATIONSHIP DIAGRAM TO RELATIONAL SCHEMA:

|  |
| --- |
| ­­­­­­­­­­ |

­­­

BACKLOG\_DATA

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| backlogID | studentID | year | semester | courseID | sectionNum | grade | f\_employeeID | date | time |

1. NORMALIZATION:

|  |  |  |
| --- | --- | --- |
| Employee | Emp\_ID | e1 |
|  | FirstName | e2 |
|  | LastName | e3 |
|  | facultyID | F1 |
|  | Emai | e4 |
|  | Address | e5 |
|  | Emp\_Type | e6 |
|  | Date \_Of\_Birth | e7 |
|  | Gender | e8 |
|  | Phone | e9 |
|  | Faculty\_ID | f1 |
| Student | StudentID | s 1 |
|  | FirstName | s 2 |
|  | LastName | s 3 |
|  | Email | s 4 |
|  | Address | s 5 |
|  | DateOfBirth | s 6 |
|  | Gender | s 7 |
|  | EnrollmentDate | s8 |
| Registration | RegistrationID | r1 |
|  | Semester | r2 |
|  | Year | r3 |
|  | StudentID | s1 |
|  | SectionID | u1 |
| Section | SectionID | u1 |
|  | SectionNum | u2 |
|  | CrouseID | o1 |
|  | FacultyID | f1 |
|  | Semester | b2 |
|  | Year | b1 |
| Evaluation | EvaluationID | m1 |
|  | Obtained \_Marks | m2 |
|  | QuestionID | a1 |
|  | RegistrationID | r1 |
| Program | program \_ID | j1 |
|  | program \_Name | j2 |
|  | Program DepartmentID | d1 |
| Course | CourseID | o1 |
|  | Course \_Name | o2 |
|  | Course \_Type | o3 |
|  | No \_Of\_Credits | o4 |
|  | Course\_Prerequisite | o5 |
|  | Enrollment\_Course b3 | b3 |
|  | StudentID | s 1 |
| Course Outline | cou\_ID | w1 |
|  | cou\_Description | w2 |
|  | Ass\_and\_MarkDistribution | w3 |
|  | Grade\_con\_Schema | w4 |
|  | Req\_Textbook | w5 |
|  | Cou\_policy | w6 |
|  | uni\_regu\_and\_codeOfConduct | w7 |
|  | SectionID | u1 |
| Department | Dep\_ID | d1 |
|  | Dep\_Name | d2 |
|  | Dep\_Ranking | d3 |
| Faculty | FacultyID | f1 |
|  | Fac\_Position | f2 |
|  | Join\_Date | f3 |
|  | Dep\_ID | d1 |
| School | School\_ID | y1 |
|  | School\_Name | y2 |
|  | School\_Ranking | y3 |
|  | Dep\_ID | d1 |
| CO | CO\_ID | c1 |
|  | CO\_Num | c2 |
|  | CourseID | o1 |
|  | PLO\_ID | p1 |
| Question | QuestionID | a1 |
|  | Question\_Name | a2 |
|  | Total\_Mark | a3 |
|  | Weight | a4 |
|  | Question\_Num | a5 |
|  | Question | a6 |
|  | CO\_ID | c1 |
| PLO | PLO\_ID | p1 |
|  | PLO\_No | p2 |
|  | Details | p3 |
|  | Program\_ID | j1 |
| Backlog data | backlogID | x1 |
|  | studentID | s1 |
|  | year | b1 |
|  | Semester | b2 |
|  | courseID | o1 |
|  | sectionNum | b4 |
|  | grade | b5 |
|  | facultyID | f1 |
|  | Date | x2 |
|  | time | x3 |
|  |  |  |

e1→ e2, e3, e4, e5, e6, e7, e8,f1

s1→ s2, s3, s4, s5, s6, s7, s8, j1, r1

r1→ r2, r3, a1, s1,u1

u1→ u2,u3, u4, o1, f1,b1,b2

m1→ m2, a1, r1

j1→ j2, d1

o1→ o2, o3, o4,o5, b3,s1

d1→ d2, d3

f1→ f2, f3, d1

w1→ w2, w3, w4, w5, w6, w7, u1

y1→ y2,y3,d1

c1→ c2, o1, p1

a1→ a2, a3, a4, a5 ,a6, c1

b1→ b2, b3,b4, b5, s1, o1

p1→ p2, p3, j1

x1→s1,,b1,b2,,o1,b4,b5,f1,x2,x3

1NF: A relation that has a primary key and in which there are no repeating groups.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| s1 | s2 | s3 | s4 | a2 | s5 | s6 | s7 | s8 | j1 | j2 | r1 | r2 | r3 | d1 | d2 | | d3 | | a1 | a2 |
| o4 | o5 | f1 | f2 | f3 | b1 | b2 | b3 | b4 | b5 | p1 | p2 | p3 | j1 | j2 |  |  | |

2NF: A relation in first normal form in which every non-key

attribute is fully functionally dependent on the primary key

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| s1 | s2 | s3 | s4 | s5 | s6 | s7 | s8 | j1 | j2 |
| r1 | r 2 | r 3 | d1 | d 2 | d 3 | a1 | a 2 | a 3 | a 4 |
| a 5 | a 6 | u1 | u2 | u3 | u4 | c1 | c2 | o1 | o2 |
| o3 | o4 | o5 | f1 | f2 | f3 | b1 | b2 | b3 | b4 |
| b5 | p1 | p2 | p3 | j1 | j2 | e1 | e2 | e3 | e4 |
| e5 | e6 | e7 | e8 | w1 | w2 | w3 | w4 | w5 | w6 |
| w7 | y1 | y2 | y3 | x1 | x2 | x3 |

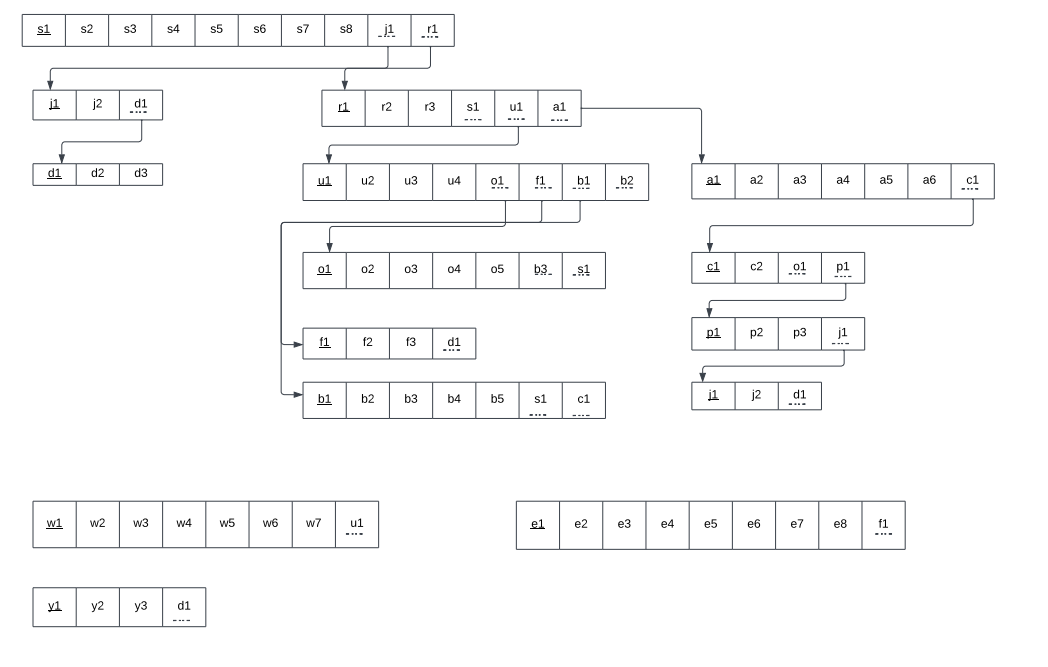
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| e1 | e2 | e3 | e4 | e5 | e6 | e7 | e8 |



|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| w1 | w2 | w3 | w4 | w5 | w6 | w7 |

|  |  |  |
| --- | --- | --- |
| y1 | y2 | y3 |

3NF: A relation that is in second normal form and has no transitive dependencies



BCNF: All determinants are candidate keys. There is no determinant that is not a unique identifier. Here, all the relations already are in BCNF.

E.DATA DICTIONARY:

VC\_T

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name |  | Data Type | Size | Remark |
| v\_employeeID |  | INTEGER | 4 | This is the foreign key from  the Employee  table. E.g:  “4250” |
| startDate |  | DATE | 10 | This is starting  date for the VC.  E.g: “01-03-  2020” |
| endDate |  | DATE | 10 | This is the date  VC retire from  his post. E.g:  “01-03-2024” |

 STUDENT\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| studentID | INTEGER | 7 | This is the primary key  for the Student table.  E.g: “1921834”. |
| firstName | VARCHAR | 40 | This is the first name  of the student. E.g:  “Rakibul”. |
| lastName | VARCHAR | 40 | This is the last name  of the student. E.g:  “Hasan”. |
| dateOfBirth | DATE | 10 | This is the birth date of  the student. E.g: “21- 12-1996”. |
| gender | VARCHAR | 6 | This is the gender of  the student.  E.g: “Female”. |
| email | VARCHAR | 50 | This is the email of the  student.  E.g:   “1921834@iub.edu.bd” |
| phone | VARCHAR | 11 | This is the phone of  the student.  E.g: “01XXXXXXXXX”. |
| address | VARCHAR | 100 | This is the address of  the student. E.g:  “House 1,Road  4,Block D,  Bashundhara RA” |
| departmentID | VARCHAR | 3 | This is the foreign key from the  Department table.  E.g: “CSE” |
| programID | INTEGER | 11 | This is the foreign key from the Program  table.   E.g: “1” |
| enrollmentSemester | VARCHAR | 10 | This is the enrollment semester of the  student. |
| enrollmentYear | VARCHAR | 4 | This is enrollment year of the student. |

 STUDENT\_COURSE\_PERFORMANCE\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| scpID | INTEGER | 11 | This is the primary key for this table |
| registrationID | INTEGER | 11 | This is the foreign key from registration table |
| totalMarksObtained | INTEGER | 3 | This is the total marks obtained by the student |
| gradePoint | FLOAT |  | This is the grade point achieved by the student |

SECTION\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| sectionID | INTEGER | 11 | This is the Primary Key for Section.   E.g: “1” |
| sectionNum | INTEGER | 11 | This is the section number.   E.g: “1” |
| semester | VARCHAR | 6 | This is the semester of the section. E.g: “Summer” |
| courseID | VARCHAR | 6 | This is the foreign key from the Course table.  E.g: “CSE101” |
| facultyID | INTEGER | 11 | This is the foreign key from Faculty table. E.g: “1801” |
| year | YEAR | 4 | This is the year this section of this course was taken by this specific faculty |

SCHOOL\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| schoolID | VARCHAR | 5 | This is the primary key of School. E.g: “SETS” |
| schoolName | VARCHAR | 50 | This is the name of the School.  E.g: “School of  Engineering, |

REGISTRATION\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| registrationID | INTEGER | 11 | This is the Primary Key for Registration.  E.g: “0101010101” |
| sectionID | INTEGER | 11 | This is the foreign key from section table |
| studentID | INTEGER | 11 | This is the foreign key from student table |

QUESTION\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| questionID | INTEGER | 11 | This is the primary key of this table |
| questionDetails | MEDIUMTEXT |  | This is the question |
| markPerQuestion | INTEGER | 11 | This is the mark each question  contains |
| questionNum | INTEGER | 11 | This is the number of the question |
| difficultyLevel | INTEGER | 11 | This is the difficulty level of the question |
| examID | VARCHAR | 20 | This is the foreign key from exam table |
| courseID | VARCHAR | 6 | This is the foreign key from course table |
| coNum | INTEGER | 11 | This is the CO number of the question |

PROGRAM\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| programID | INTEGER | 11 | This is the primary key for a program.  E.g: “1” |
| programName | VARCHAR | 50 | This is the name of the program. E.g: “Bachelor of Science” |
| departmentID | VARCHAR | 4 | This is the foreign key from the  Department table. |

PO\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| poID | VARCHAR | 5 | This is the primary  key for Program  Outcome.   E.g: “PO1” |
| poNum | INTEGER | 11 | This is the PO number. E.g: “1” |
| programID | INTEGER | 11 | This is a foreign key from Program table.  E.g: “1” |

PLO\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| ploID | INTEGER | 11 | This is the primary key for Program Learning Outcome.  E.g: “PLO1” |
| ploNum | INTEGER | 11 | This is the PLO number. E.g: “1” |
| programID | INTEGER | 11 | This is a foreign key from Program table.  E.g: “1” |

LESSON\_PLAN\_STRATEGY\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| lpsID | INTEGER | 11 | This is the primary key of the table |
| week | INTEGER | 11 | This is the week number |
| topic | MEDIUMTEXT |  | This is the topic name |
| learningStrategy | MEDIUMTEXT |  | This is the lesson plan strategy of  that topic |
| assessmentStrategy | VARCHAR | 10 | This is the assessment strategy of that  Topic |
| courseOutlineID | INTEGER | 11 | This is the foreign key from course outline table |

FACULTY\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| f\_employeeID | INTEGER | 11 | This is the foreign  key from the Employee table.   E.g: “4250” |
| departmentID | VARCHAR | 3 | This is the DepartmentID of the department faculty belongs to. E.g: “CSE” |
| rank | VARCHAR | 30 | This is the rank of the faculty. E.g:  “Assistant Professor” |
| joinDate | DATE | 10 | This is starting date.  E.g: “01-03-2020” |

EXAM\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| examID | INTEGER | 11 | This is the primary key for this table |
| examName | VARCHAR | 30 | This is the name of the exam |
| sectionID | INTEGER | 11 | This is the foreign key from exam table |

EVALUATION\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| evaluationID | INTEGER | 11 | This is the primary key for this table |
| examID | VARCHAR | 20 | This is the foreign key from exam table |
| registrationID | INTEGER | 11 | This is the foreign key from registration table |
| totalMarks | INTEGER | 11 | This is the total marks achieved by  the student in a specific exam |

EVALUATION\_STRATEGY\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| eSID | INTEGER | 11 | This is the primary key for this table |
| courseOutlineID | INTEGER | 11 | This is the foreign key from course outline table |

ASSESSMENT\_TOOL\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| asID | INTEGER | 11 | This is the primary key for this table |
| eSID | INTEGER | 11 | This is the foreign key from evaluation strategy table |

MARK\_DISTRIBUTION\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| mdID | INTEGER | 11 | This is the primary key for this table |
| eSID | INTEGER | 11 | This is the foreign key from evaluation strategy table |

BLOOMS\_CATEGORY\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| bcID | INTEGER | 11 | This is the primary key for this table |
| eSID | INTEGER | 11 | This is the foreign key from evaluation strategy table |

EMPLOYEE\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| employeeID | INTEGER | 11 | This is the primary  key for Employee  table.  E.g: “1801” |
| password | VARCHAR | 10 | This is the password of the employee |
| firstName | VARCHAR | 50 | This is the last name of the faculty.  E.g: “Ahmed” |
| lastName | VARCHAR | 50 | This is the last name of the faculty.  E.g: “Ahmed” |

DEPARTMENTHEAD\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| h\_employeeID | INTEGER | 11 | This is the foreign key from the Employee table. |
| departmentID | VARCHAR | 3 | This is the DepartmentID of the department HEAD manages.   E.g: “CSE” |
| startDate | DATE |  | This is starting date.  E.g: “01-03-2020” |
| endDate | DATE |  | This is the date HEAD retire from  his post. E.g: “01- 03-2024” |

DEPARTMENT\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| departmentID | VARCHAR | 3 | This is the primary key for the  Department table.  E.g: “CSE” |
| departmentName | VARCHAR | 50 | 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”. |

DEAN\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| d\_employeeID | INTEGER | 11 | 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 |  | This is starting date.   E.g: “01-03-2020” |
| endDate | DATE |  | This is the date DEAN retire from  his post. E.g: “01- 03-2024” |

COURSE\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| courseID | VARCHAR | 6 | This is the Primary Key for the Course.  E.g: “CSE203” |
| courseName | VARCHAR | 40 | This is the name of the Course.  E.g: “Discreet Mathematics” |
| numOfCredits | INTEGER | 11 | 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 | 11 | This is the foreign key from the program table.  E.g: “1” |

COURSE\_OUTLINE\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| courseOutlineID | INTEGER | 11 | This is the primary key for this table |
| sectionID | INTEGER | 11 | This is the foreign key from the  section table |
| courseDescription | MEDIUMTEXT |  | This is the description of the  course |
| objective | MEDIUMTEXT |  | This is the objective of the course |
| content | MEDIUMTEXT |  | This is the content of the course |
| refMaterials | MEDIUMTEXT |  | This is the reference material |
| courseTitle | VARCHAR | 1000 | This is the title of the course |
| prerequsiteCode | VARCHAR | 6 | This is the prerequisite course code |
| creditValue | INTEGER | 11 | This is the credit value of the course |

CO\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| coID | INTEGER | 11 | This is the primary  key for the CO table. E.g: “CO1”. |
| coNum | INTEGER | 11 | This is the CO number.  E.g: 1,2 etc. |
| courseID | VARCHAR | 6 | 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” |
| poID | VARCHAR | 6 | This is the foreign key from the PLO table. E.g: “PO1” |

CLO\_MATRIX\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| clo\_MatID | INTEGER | 11 | This is the primary key for this table |
| cloNum | INTEGER | 11 | This is the clo number |
| coDescription | MEDIUMTEXT |  | This is the CO Description |
| ploAssessed | VARCHAR | 10 | This is the name of the plo assessed |
| correlation | INTEGER | 11 | This is the correlation value or number |
| courseOutlineID | INTEGER | 11 | This is the foreign key from the course outline table |

ANSWER\_T

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Data Type | Size | Remark |
| answerID | INTEGER | 11 | This is the primary key for this table |
| answerDetails | MEDIUMTEXT |  | This is the answer details |
| answerNum | INTEGER | 11 | This is the number of the answer |
| markObtained | INTEGER | 11 | This is the mark obtained by the student for each answer |
| registrationID | INTEGER | 11 | This is the foreign key from registration table |
| examID | INTEGER | 11 | This is the foreign key from the exam table |

CHAPTER-4 PHYSICAL SYSTEM DESIGN:

Graphical user interface

Description automatically generated

Text

Description automatically generated

Graphical user interface

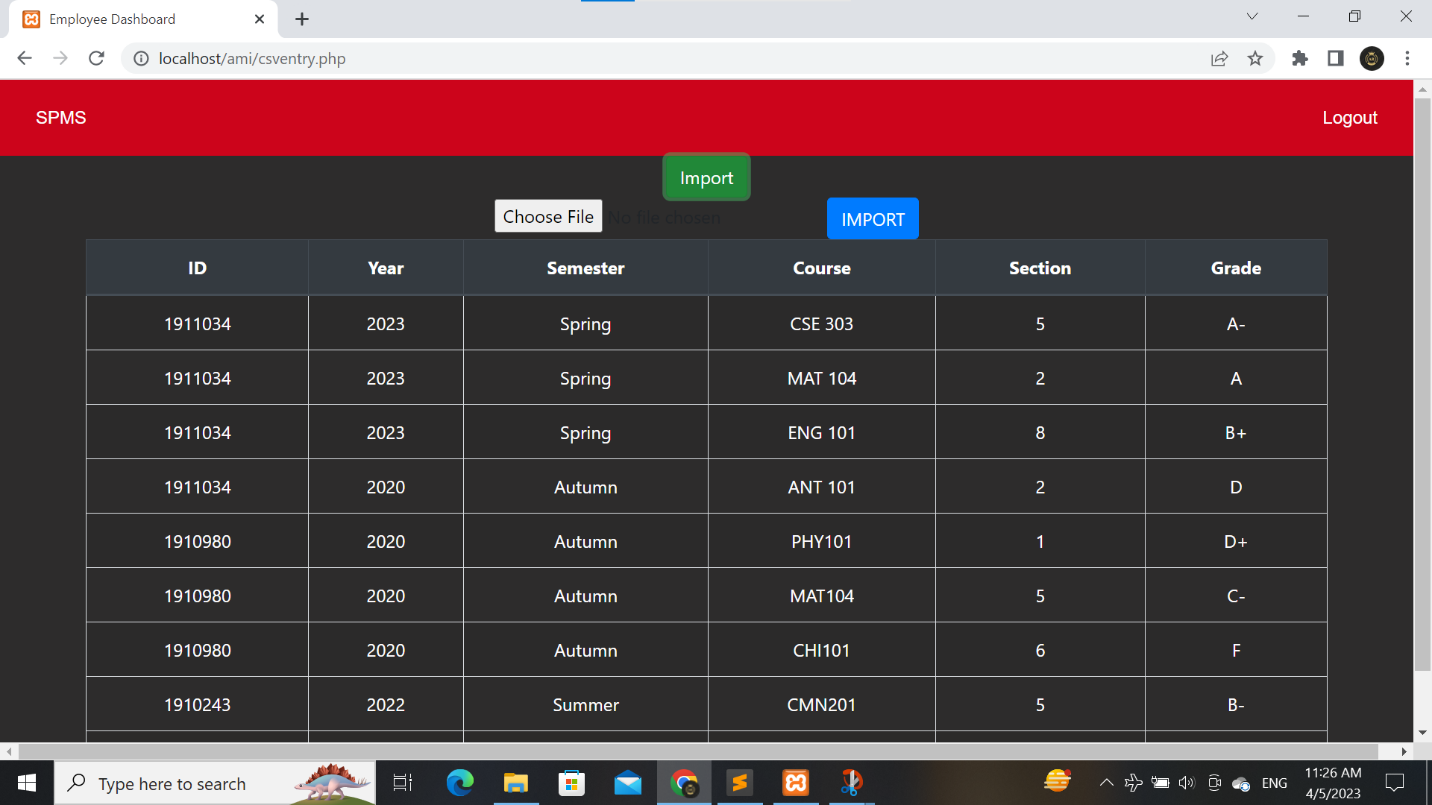
Description automatically generated

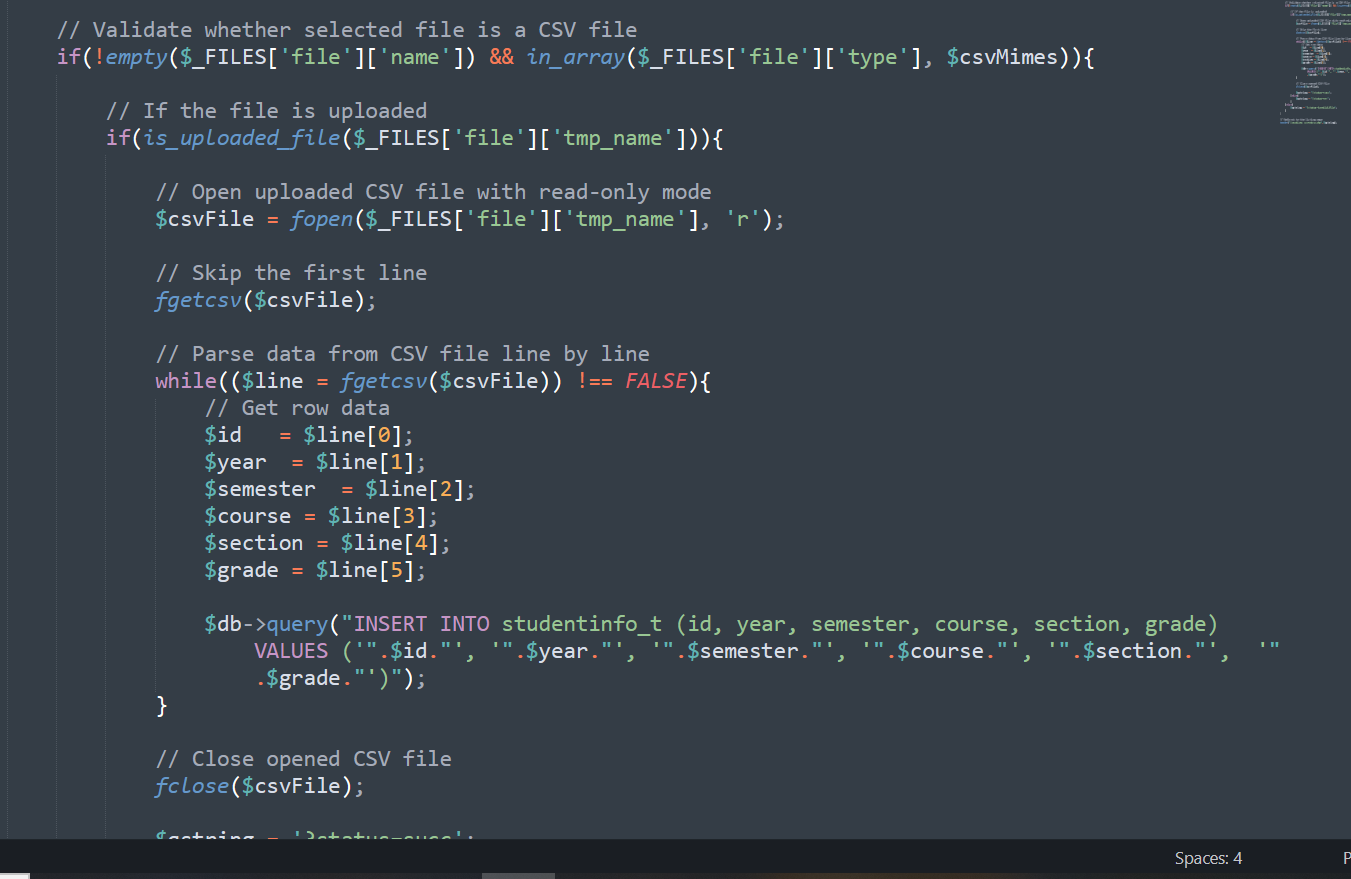
Text

Description automatically generated

Text

Description automatically generated





Chart, radar chart

Description automatically generated

Text

Description automatically generated

CHAPTER 5: CONCLUSION

PROBLEM & SOLUTION:

ANALYSIS PHASE: -

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

DESIGNING PHASE: -

A numerous amount of problems was faced while creating the ERD and Relational Schema during the Design Phase, However, some feedbacks from the faculty were pretty much useful enough to overcome those issues but we are still under a little dilemma about these.

IMPLEMENTATION PHASE: -

All the System Requirements were completed successfully.

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

Back-End Developing tools: PHP, JSON

Database-integration: MySQL

ADDITIONAL FEATURES & FUTURE DEVELOPMENTS: -

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

REFERENCES: -

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