



DATABASE MANAGEMENT

Project Report

GROUP 05

Abu Bakar Siddique (1731237)

Aqib Reza (1731085)

Md Musaraf Hussain (1731374)

Tashdid Zaman (1720718)

Syedur Roshid Emon (1711181)

Rakib Chowdhury Joy (2020255)

Contents

Chapter 1: Introduction	3
Background of the Organization	3
Background of the Project.....	3
Objectives of the Project.....	4
Scope of the Project.....	5
Chapter 2: Requirement Analysis	6
A. Six Elements Analysis - Existing System.....	7
B. Business Process Model and Notation 1.0 (As IS):	10
C. Problem Analysis – Existing System	13
D. Rich Picture - Proposed System	15
E.Six Elements Analysis - Proposed System.....	16
F.Business Process Model and Notation 2.0 – Proposed System.....	22

CHAPTER 1: INTRODUCTION

BACKGROUND OF THE ORGANIZATION

Independent University of Bangladesh (IUB) is one of the country's most prestigious private higher education institutions. IUB, which now has 8,423 students enrolled, 13,745 alumni, and 401 faculty members, aspires to create world-class graduates while also being in the forefront of cutting-edge research in fields ranging from Computer Science to Public Health. The University now includes six academic schools:

- School of Business
- School of Engineering and Computer Science
- School of Environmental Sciences and Management
- School of Liberal Arts and Social Sciences
- School of Life Sciences
- School of Public Health

The School of Architecture and the School of Biotechnology are on their way to become part of its constantly expanding offering. Aside from the wide campus, well-equipped laboratories, and vast library, the institution provides generous scholarships to its students, both academic and need-based, so that they can gain the essential information and expertise to be well-prepared for their future employment.

BACKGROUND OF THE PROJECT

Every semester, an institution the size of IUB generates a massive amount of enrollment data. For the people whose job it is to filter this data into a form that higher authorities can utilize to better guide their decision-making, analyzing this data is a tremendous endeavor. From start to finish, it's a laborious procedure involving a lot of time-consuming and error-prone manual labor.

Introduce yourself to the Student Enrollment Analysis System (SEAS) (SEAS). We want to employ digital automation to eliminate this tedious, boring procedure. We propose that the enrollment data be saved in the

database that comes with the Student Enrollment Analysis System (v1.2). SEAS(v1.2) will process this data without the need for human intervention and provide the many charts and tables that IUB administration and stakeholders require to run the university efficiently on-demand. As a result, staff will be freed up to focus on more pressing and relevant activities, while higher-ups will have fast access to all the data, they require with just a few mouse clicks.

Overall, SEAS will save Independent University, Bangladesh time, money, and resources that can be better spent on the institution's principal purpose of providing the greatest education possible to the students that make it so unique.

OBJECTIVES OF THE PROJECT

The primary goal of our project is to develop software that will generate data such as total student enrollment by school and department, enrollment-based number of section comparison among those schools, IUB unused resources analysis based on enrollment, i.e., classroom requirements, and section analysis by semester for various schools and departments. All this vital information may be acquired without requiring much human intervention in the creation of a usable product. This program will improve efficiency and productivity among those in need of such critical information by allowing easy data access, which will save manual work and lead to fewer human errors.

However, while the cost of developing such a system may be significant at first, it is very low in comparison to the cost of performing the same data retrieval operation manually. Furthermore, as the process gets more automated, individuals will have more time to focus on other manual tasks. Furthermore, because this automated data extraction technique leaves little/no space for human error, the quality of the job generated is improved, reducing possible expenditures.

SCOPE OF THE PROJECT

We uncovered areas in the business processes that can cause major delays in time and communication after a thorough evaluation of the current system, which we will analyze in the following chapter.

Our proposed solution is to develop a Student Enrollment Analysis System (SEAS), which will be a web application with a centralized database that will be powered by the MySQL Relational Database Management System (RDMS). It will be able to store, edit, add, and update data needed to generate accurate reports with charts and tables automatically. The money earned in each semester, as well as a detailed analysis of student enrollment in terms of availability and utilization of IUB resources, will be available to view.

The suggested system will be able to extract newly aggregated data from uploaded tally-sheet spreadsheet files and add it to the required relational tables in a highly ordered manner. The system will then be able to execute all the necessary computations to generate the user's desired charts and tables.

A user interface will be created and connected to the database to serve as the web application's portal, allowing users to log in and see what types of reports are available to read. These reports will be created according to the functional requirements' recommendations.

CHAPTER 2: REQUIREMENT ANALYSIS

Rich Picture (AS IS):

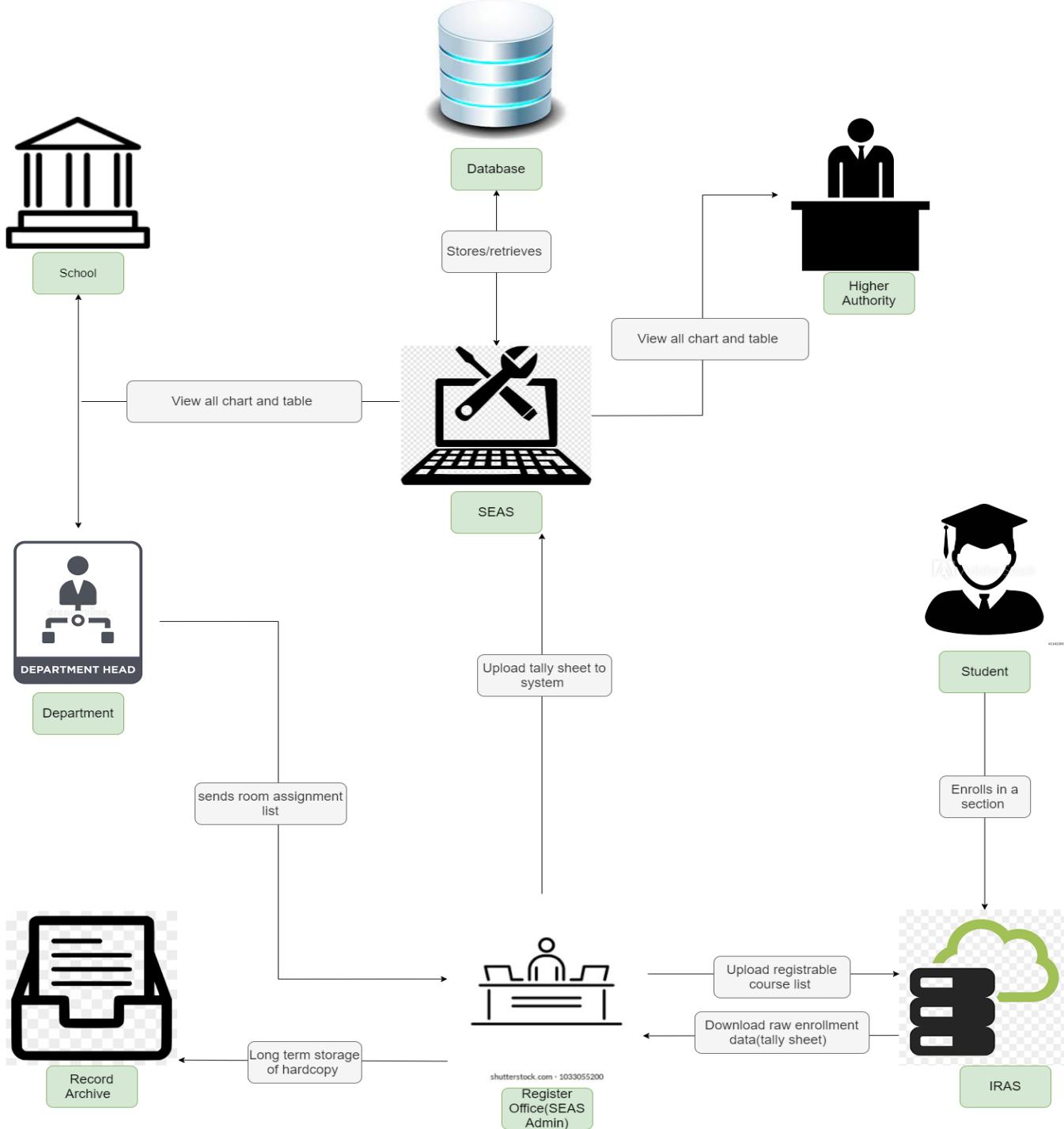


Fig 1: Rich Picture (Existing system)

A. SIX ELEMENTS ANALYSIS - EXISTING SYSTEM

Process	System Roles					
	Human	Non-Computing Hardware	Computer Hardware	Software	Database	Network & Communication
Class Predictions send to registrar	<p>Department:</p> <ul style="list-style-type: none"> 1. Views previous semester data from the proposed system. 2. Make a prediction. 3. Make sections. 4. Send to registrar. 5. Make a copy for storage. 	<p>Pen & paper</p> <ul style="list-style-type: none"> 1. Hard copy of list printed and signed before storing for future reference. 	<p>Computer</p> <ul style="list-style-type: none"> 1. It is used to create sections, and to send a list to the registrar. <p>Printer</p> <ul style="list-style-type: none"> 1. Used to print the hardcopy of the file. 	<p>Proposed System Website:</p> <ul style="list-style-type: none"> 1. For viewing the data for previous semester enrollment data <p>Microsoft Office</p> <ul style="list-style-type: none"> 1. Microsoft Office will be used to make the document. <p>Gmail</p> <ul style="list-style-type: none"> 1. Used to email the files to the registrar. 	<p>Proposed System Database:</p> <ul style="list-style-type: none"> 1. Contains historical enrollment data that the system draws from to show charts and tables. <p>Department Archive:</p> <ul style="list-style-type: none"> Stores the hardcopy/softcopy of previous semester data that is used to make the predictions. 	<p>Internet</p> <ul style="list-style-type: none"> For email communication between offices. <p>Intercom:</p> <ul style="list-style-type: none"> For rapid communication between offices
Registrar uploads tally sheets into the proposed system	<p>Registrar:</p> <ul style="list-style-type: none"> 1. Log in to IRAS. 2. Download the semester tally sheet. 3. Log in to proposed system. 4. Upload tally sheet. 5. Notify department. 		<p>Computer</p> <ul style="list-style-type: none"> 1. For accessing IRAS and the proposed system. 2. For accessing email client to email the department to notify. 	<p>SEAS:</p> <ul style="list-style-type: none"> 1. Tally sheet uploaded here <p>IRAS:</p> <ul style="list-style-type: none"> 1. Generates the tally sheet as students enroll into courses. <p>Gmail</p> <ul style="list-style-type: none"> 1. Used to email the departments 	<p>SEAS Database:</p> <ul style="list-style-type: none"> 1. Tally Sheet scanned to add semester enrollment data to database, 	<p>Internet</p> <ul style="list-style-type: none"> For email communication between offices.

Department Rechecks Room Assignment	<p>Department: 1. Log in to proposed system 2. Rechecks the room assignment based on surplus and deficit 3. Sends revised room assignment to registrar's office</p> <p>Registrar's Office: 1. Registrar logs in to the system 2. Registrar updates the room assignment for students. 3. Registrar updates archive. 4. Registrar updates the tally sheet.</p>	<p>Paper: Needed for printing the hardcopy for archiving.</p>	<p>Computer: 1. Used to access the proposed system. 2. Used to access email clients to send the data to the registrar.</p> <p>Printer: To print the hardcopy of the revised data.</p>	<p>Proposed System Website: 1. For viewing surplus and deficit sections of the current semester.</p> <p>Microsoft Office 1. Microsoft Office will be used to make the document</p> <p>2. To view the revised document.</p> <p>Gmail 1. Used to email the files to the registrar.</p>	<p>Proposed System Database: 1. Enrollment data of the current semester is updated from the new tally sheet.</p> <p>Department Archive: 1. Stores revised section enrollment data.</p>	<p>Internet: For email communication between offices.</p> <p>Intercom: For rapid communication between offices</p>
Higher authority views historical enrollment and revenue charts generated by data	<p>Higher Authority: 1. Logs in to SEAS 2. Select view charts. 3. Selects range of years 4. Can choose appropriate chart from list of charts available 5. Clicks a school name from a list of schools. 5. Can choose appropriate chart from list of charts available 6. Clicks a department name from a list of departments 7. Can choose appropriate chart</p>	<p>Paper: 1. Needed to print if the user wants to</p>	<p>Computer: 1. Used to access the SEAS</p> <p>Printer: 1. Needed to print if the user wants to</p>	<p>Student Enrollment Analysis System: 1. For viewing historical enrollment and revenue related charts</p>	<p>SEAS Database: 1. Contains historical enrollment data and revenue related data that the system draws from to show charts and tables.</p>	<p>Internet For accessing the proposed system website from outside.</p>

	from list of charts available					
Higher Authority views class size distribution data for schools	Higher Authority: 1. Logs in to proposed system 2. Selects a chart/table option from Class Size Distribution column 3 Can choose appropriate chart from list of charts available	Paper: 1. Needed to print if the user wants to	Computer/Mobile: 1. Used to access the proposed system. Printer: 1. Needed to print if the user wants to (after downloading)	Proposed System Website: 1. For viewing semester-wise data	Proposed System Database: 1. Contains historical enrollment data that the system draws from to show charts and tables.	Internet For accessing the proposed system website from outside.
Higher Authority views resource requirement and usage data	Higher Authority: 1. Logs in to proposed system 2. Selects a chart/table option from Resource Requirement and Usage column 3. Can choose appropriate chart from list of charts available	Paper: 1. Needed to print if the user wants to.	Computer/Mobile: 1. Used to access the proposed system. Printer: 1. Needed to print if the user wants to.	Proposed System Website: 1. For viewing resource requirement related charts and tables.	Proposed System Database: 1. Contains historical enrollment and IUB's existing resource capacity data that the system draws from to show charts and tables.	Internet For accessing the proposed system website from outside.

B. BUSINESS PROCESS MODEL AND NOTATION 1.0 (AS IS):

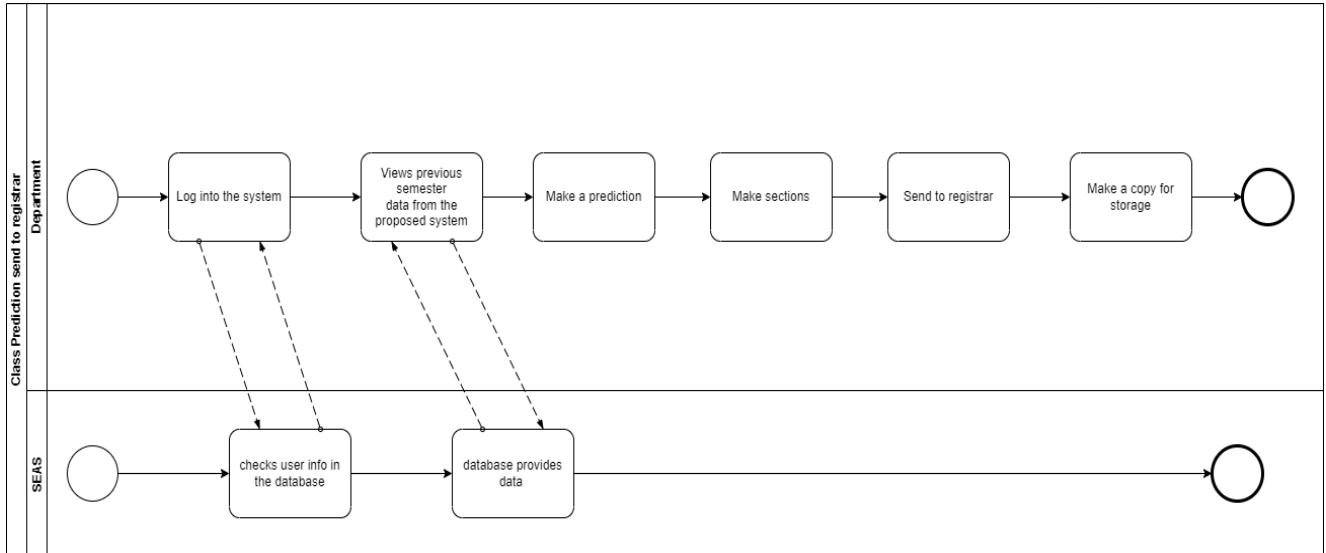


Fig 2: Class Prediction send to register

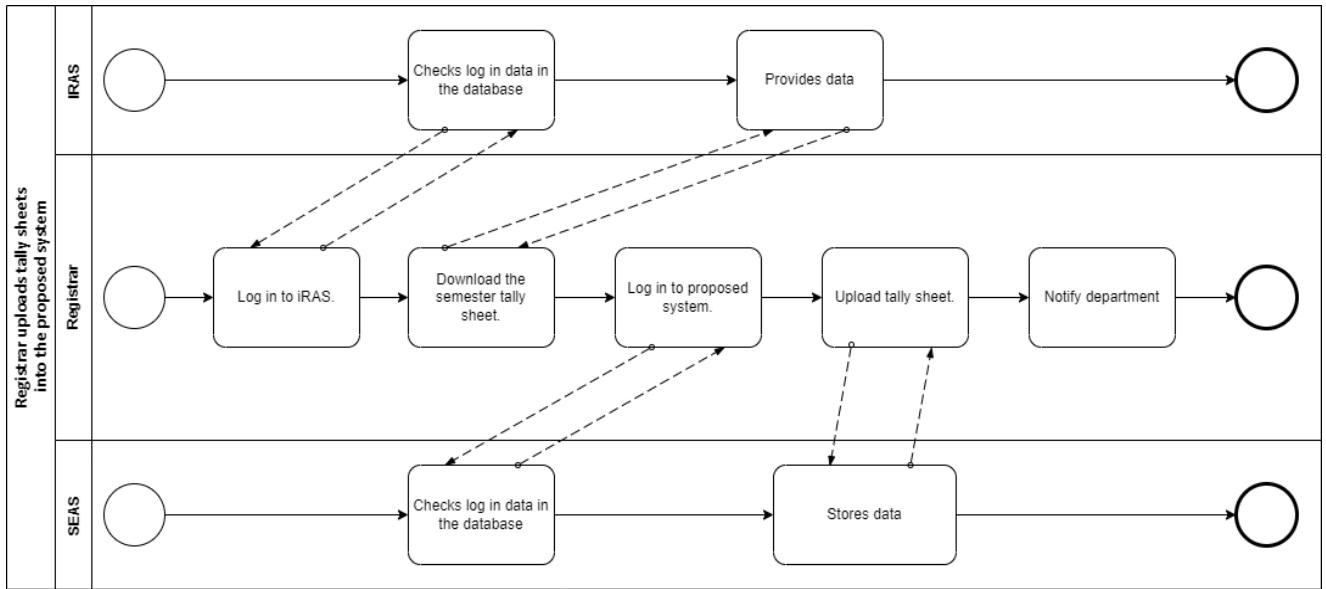


Fig 3: Register Uploads Tally Sheets

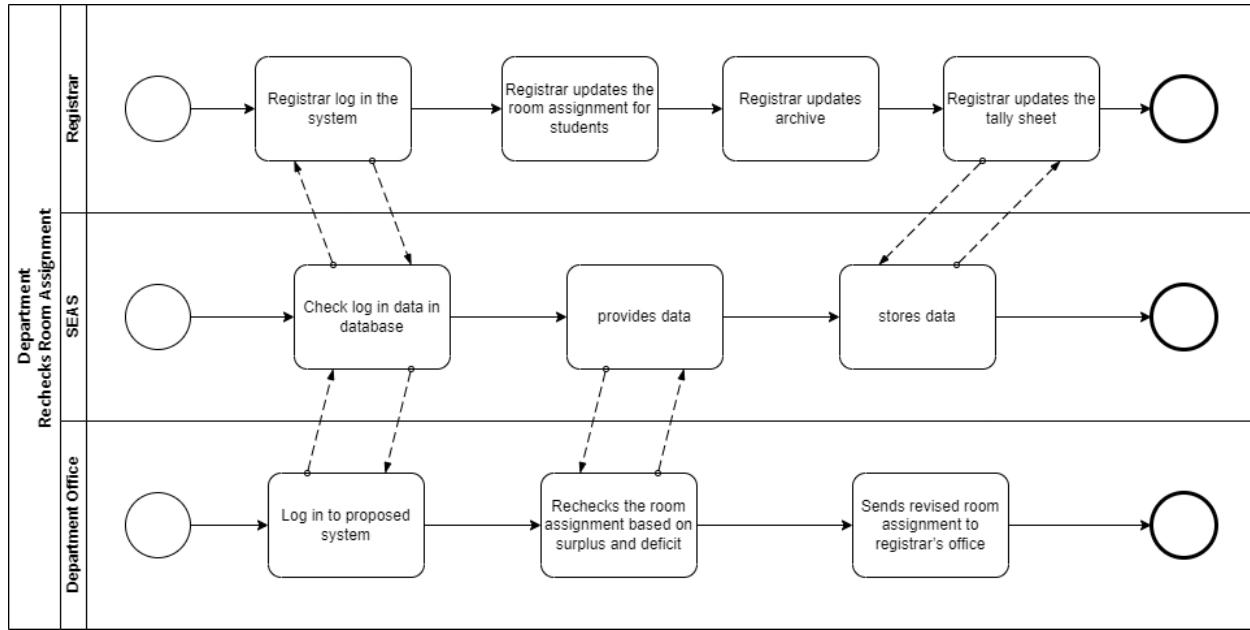


Fig 4: Department rechecks room assignment

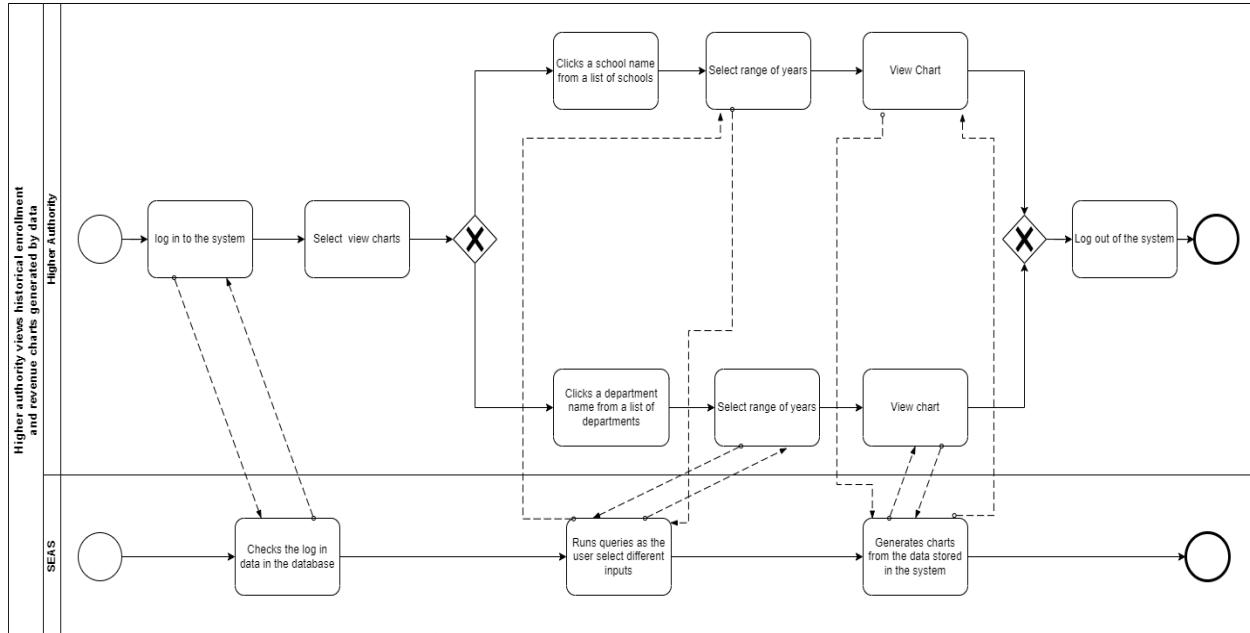


Fig 5: Higher authority views historical enrollment and revenue charts

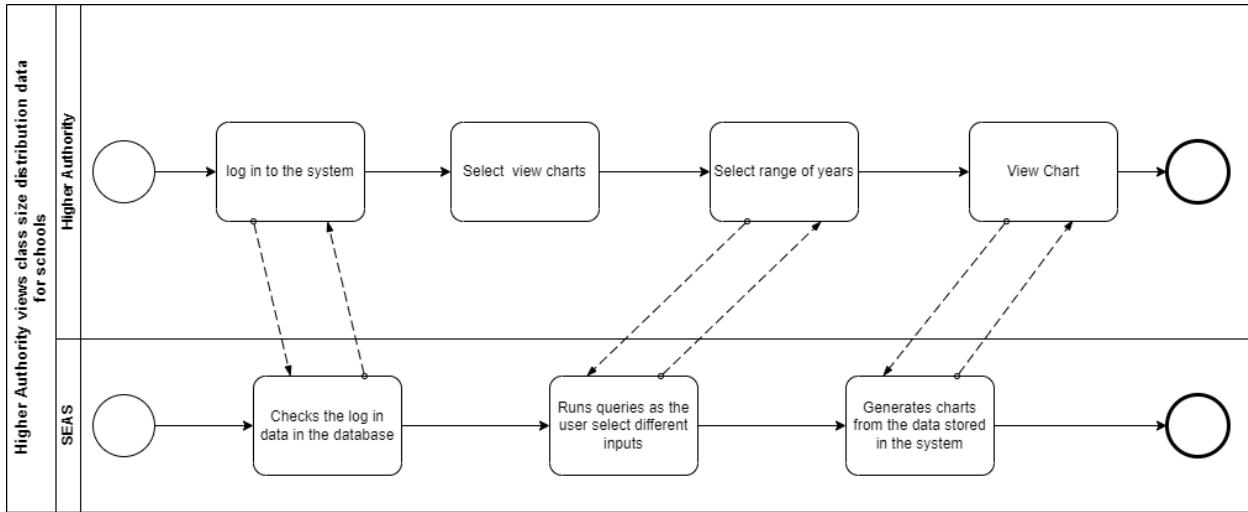


Fig 6: Higher authority views class size distribution

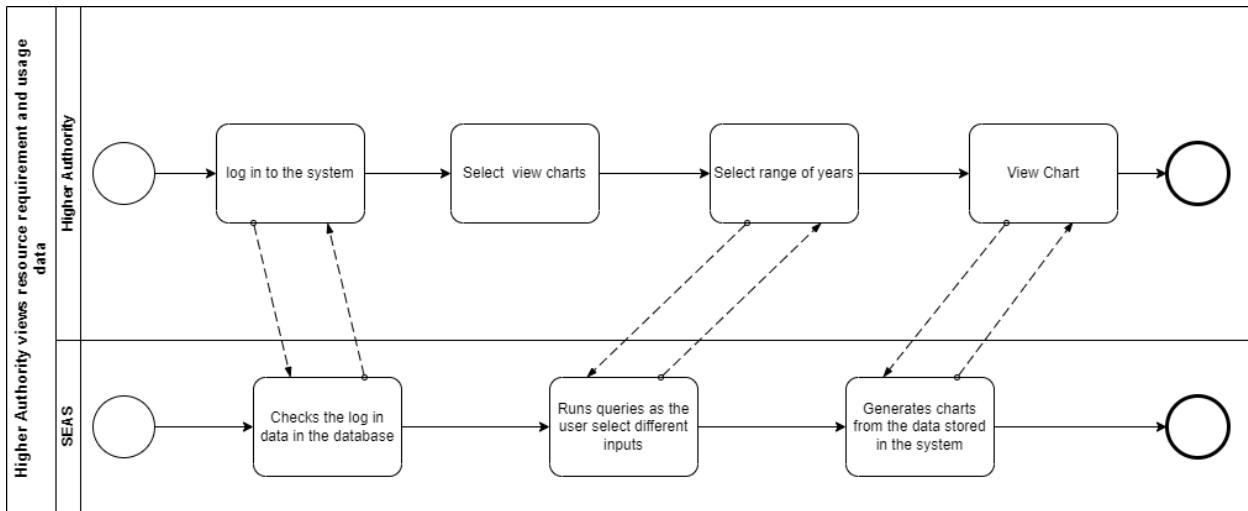


Fig 7: Higher authority views resource requirement and usage

C. PROBLEM ANALYSIS – EXISTING SYSTEM

Process Name	Stakeholders	Concerns (Issues/Problems)	Analysis (Reason of the problems)	Proposed Solution
Classroom Requirement Summary	1.Register Office 2.Higher Authority	1.If a user wants to see the specific semester classroom requirement summary then its not possible for this process. 2. Also user can not select the range of class size 3.This process is lengthy, time consuming, consists of trivial tasks and error prone	1 There are different departments & they give different types of tally sheets in different formats. So, it becomes extremely difficult to track specific course or section for a certain time	This issue can be resolved by storing these data in our database. The data won't get lost from the database & the process will be faster.
Available resource summary	1.Department 2. Dean of School	1. Storing softcopies and hardcopies becomes laborious and hard to manage. 2. There should be a problem of storage cloud of	1. Storing softcopies and hardcopies can become extremely difficult to manage when the organization	1. These issues can be resolved by storing these data tables in our database will internally calculated and generate table based on users

		hard copy and soft copy.	<p>has been operating for years.</p> <p>2. Sending hardcopy and softcopy of analysis report to the Registrar's Office and Leadership Team for storage could increase time consumption and delays or loss of information and important data.</p>	request and providing the relevant information.
Resource usage summary	1. Dean 2. Department 3. Registrar's Office 4. Leadership Team	1. There should be an error in the calculation. 2. Time consumption.	<p>1. The calculation is done by human so there is possibility to do some mistakes.</p> <p>2. This calculation process can took lot of time.</p>	1. Our system will collect all the data and calculate data and create a table populating the calculated data.

D. RICH PICTURE - PROPOSED SYSTEM

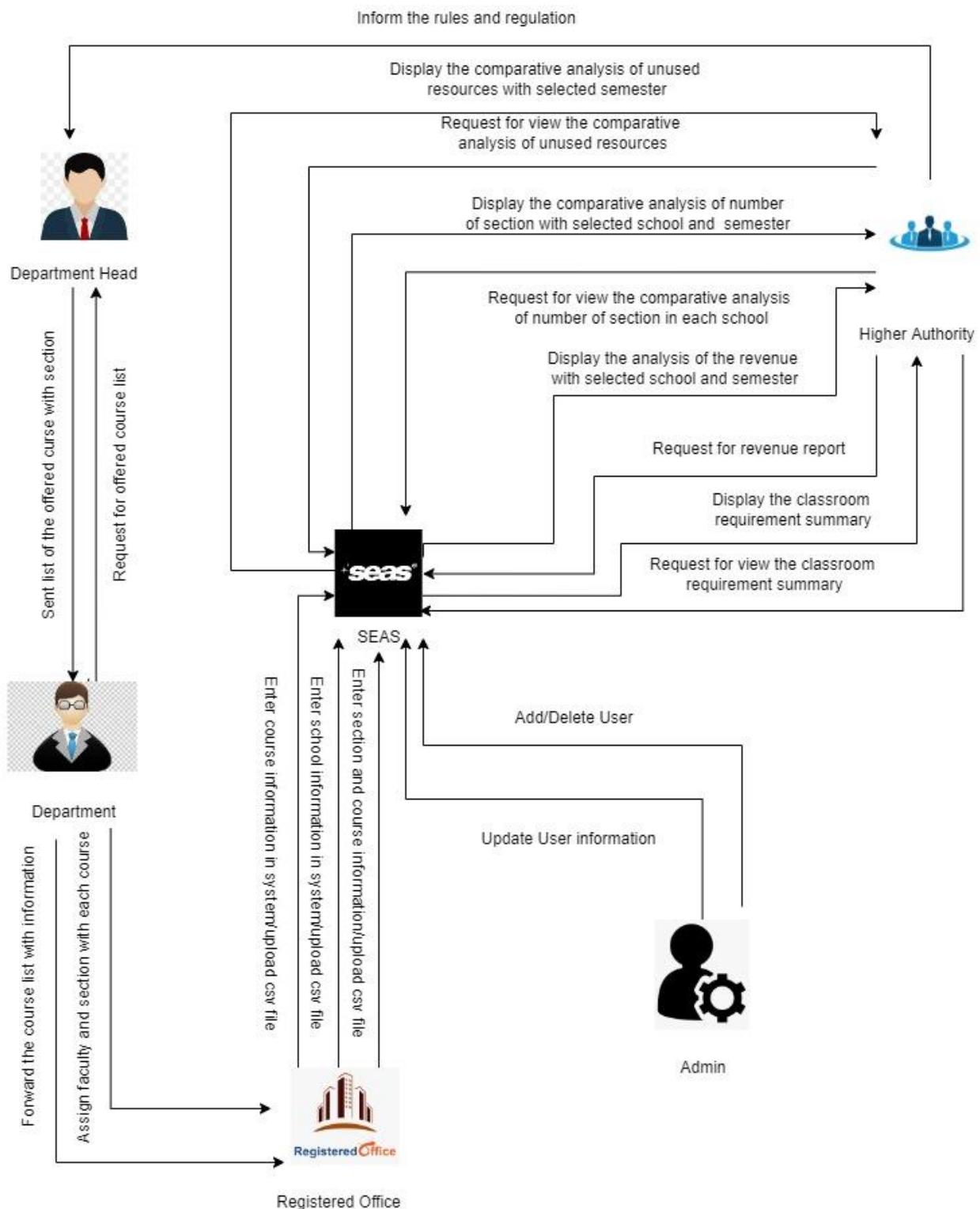


Fig 8: Rich Picture (Proposed System)

E.SIX ELEMENTS ANALYSIS - PROPOSED SYSTEM

Process	System Roles					
	Human	Non-Computing Hardware	Computing Hardware	Software	Database	Network and Communication
Generate revenue report	Higher Authority: 1.Login into the system using HA email and password. 2. Select school and semester 2.view the school and semester wise report 3.Display the analysis of the revenue report	Paper: Print all charts and tables.	Computer: To get access to the system. Printer: To print out of hard copies of this chart and table	Excel sheet: The sheet is used for store revenue data.	System Database: Store all revenue related data.	Internet: To access the system.
Display comparative analysis of number of sections	Higher Authority: 1. Login using HA id & password. 2. check email and password.	Paper: Print all charts and tables.	Computer: To get access to the system. Printer: To print out of hard	Excel sheet: The sheet is used for store number of section in each school.	System Database: Store all course and section info.	Internet: 1. The Internet is used to access SEAS 1.1

for each school	3. If match view the HA dashboard. 4. Select school and semester. 5. Display the comparative analysis in bar chart.		copies of this chart and table			
Display the unused resources with comparison for different school.	Higher Authority: 1. Login in using HA Email & password. 2. check email and password. 3. If match view the HA dashboard. 4. Select School and semester 5. Display the unused resources.	Paper: Print all charts and tables.	Computer: 1. Need to login into SEAS 1.1. Printer: To print out of hard copies of this chart and table	Excel sheet: The sheet is used for store the unused resources information.	System Database: Store all unused resources data	Internet/Mail: 1. Require to access SEAS 1.1
Display the number of sections of offered courses	Higher Authority: . Login in using HA Email & password.	Paper: Print all charts and tables.	Computer: 1. Need to login into SEAS 1.1. Printer:	Excel sheet: The sheet is used for store the offered courses information.	System Database: Store all unused resources data	Internet/Mail: 1. Require accessing SEAS 1.1

with selected with a range of class size.	<p>2. check email and password.</p> <p>3. If match view the HA dashboard.</p> <p>4. Select the department and class size.</p> <p>5. Display comparison of the number of sections of offered courses in bar chart</p>		<p>To print out of hard copies of this chart and table</p>			
Display Classroom Requirement Summary	<p>Register office:</p> <p>1.Login into SEAS system.</p> <p>2. Receive the offered course list from the department.</p> <p>3.Store the list & submit it to the SEAS.</p> <p>4. Gets the tally sheet from the department.</p>	<p>Pen and Paper:</p> <p>1.Used for noting down some important points if needed.</p> <p>2.Need to print if the user required.</p>	<p>Computer: All related data is searched and stored using a computer.</p> <p>Printer: Print the hard copy of the tally sheet.</p>	<p>Excel: 1.Used to create the available faculty & section list data.</p> <p>Email: 1.Used it to send the file</p>	<p>System Database: 1.The system database will store all the required data to generate a table.</p>	<p>Internet:</p> <p>1. All related data are searched through the internet.</p> <p>2.Required to login</p>

	<p>5. Download the semester tally sheet.</p> <p>6.Upolad the tally sheet in the system.</p> <p>Departmen t:</p> <p>1.Receive the list of offered courses from the department head.</p> <p>2.make the list of available faculties.</p> <p>3.Create a section list for each course.</p> <p>4.Store the faculty & course offer list</p> <p>5. Send the offered course list to the register office.</p> <p>Higher Authority:</p> <p>1.Login to the system.</p> <p>2.View the classroom requirement</p>				
--	--	--	--	--	--

	data as per course offer.				
Enter the course, section, and school information into the system	<p>Register Office:</p> <ol style="list-style-type: none"> 1. Login using RO id & password. 2. check email and password. 3. If match view the RO dashboard. 4. select add course/add school/add section option. 5. A input form will be display with required fill. 6. fill up all the required filled. 7. Select submit option <p>SEAS:</p> <ol style="list-style-type: none"> 1.Store all the information into the system database 		<p>Computer: All related data is searched and stored using a computer</p>	<p>System Database:</p> <p>1.The system database will store all the required data.</p> <p>.</p>	<p>Internet:</p> <ol style="list-style-type: none"> 1. All related data are searched through the internet. 2.Required to login

ADD/Delete the user into the system.	<p>Admin:</p> <ol style="list-style-type: none"> 1. Login using admin email & password. 2. check email and password. 3. If match view the admin dashboard. 4. select add/delete user option 5. For adding a user a form will be displayed. 6. Fill the username, password and select the type of the user. 7. select submit option. 	<p>Computer:</p> <ol style="list-style-type: none"> 1. User will need a computer to access SEAS 1.1 	<p>System Database:</p> <ol style="list-style-type: none"> 1.The system database will store all the user information 	<p>Internet:</p> <ol style="list-style-type: none"> 1. All related data are searched through the internet. 2.Required to login
--------------------------------------	---	---	--	---

F. BUSINESS PROCESS MODEL AND NOTATION 2.0 – PROPOSED SYSTEM

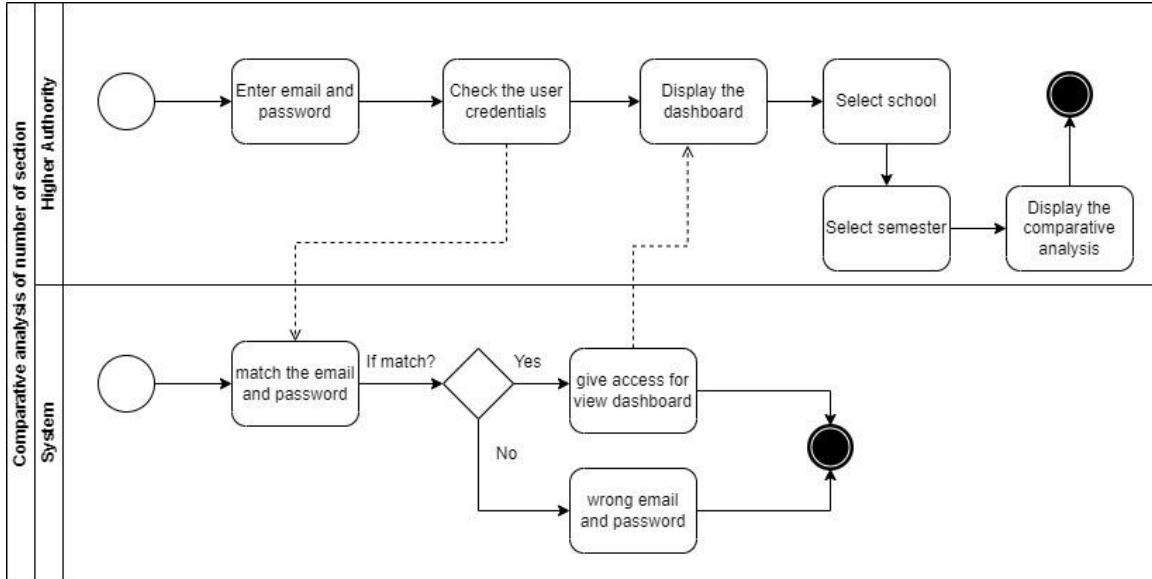


Fig 9: Comparative analysis of number of section

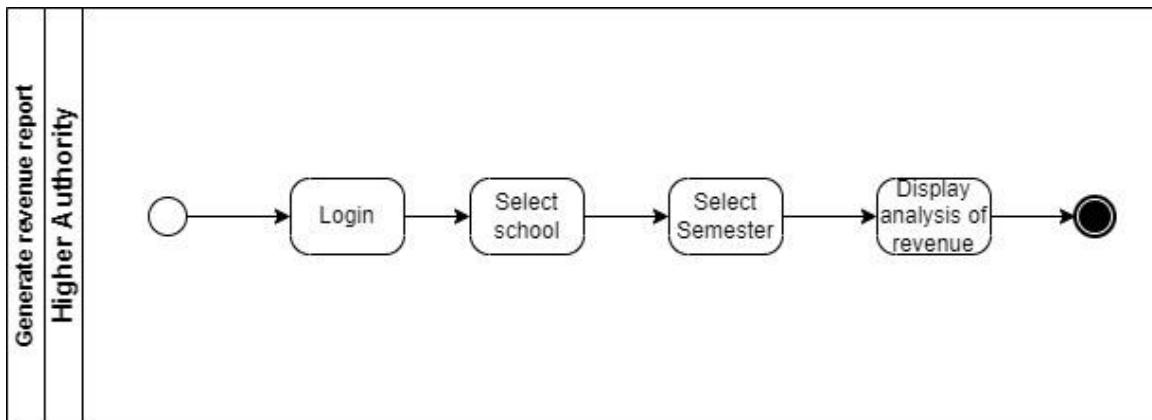


Fig 10: Generate revenue report

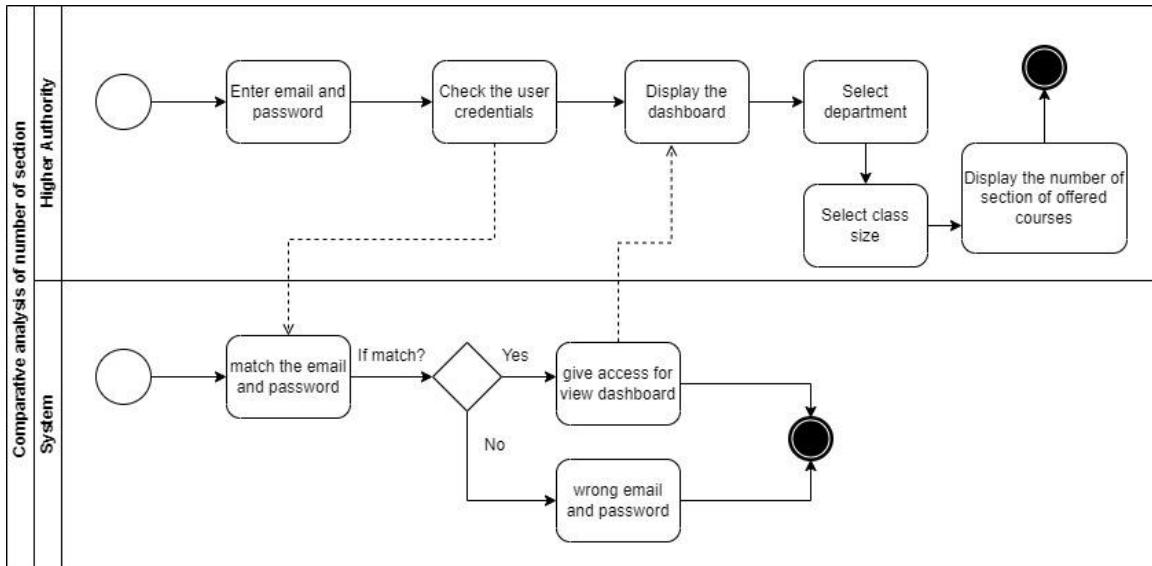


Fig 11: Department wise comparative analysis number of section

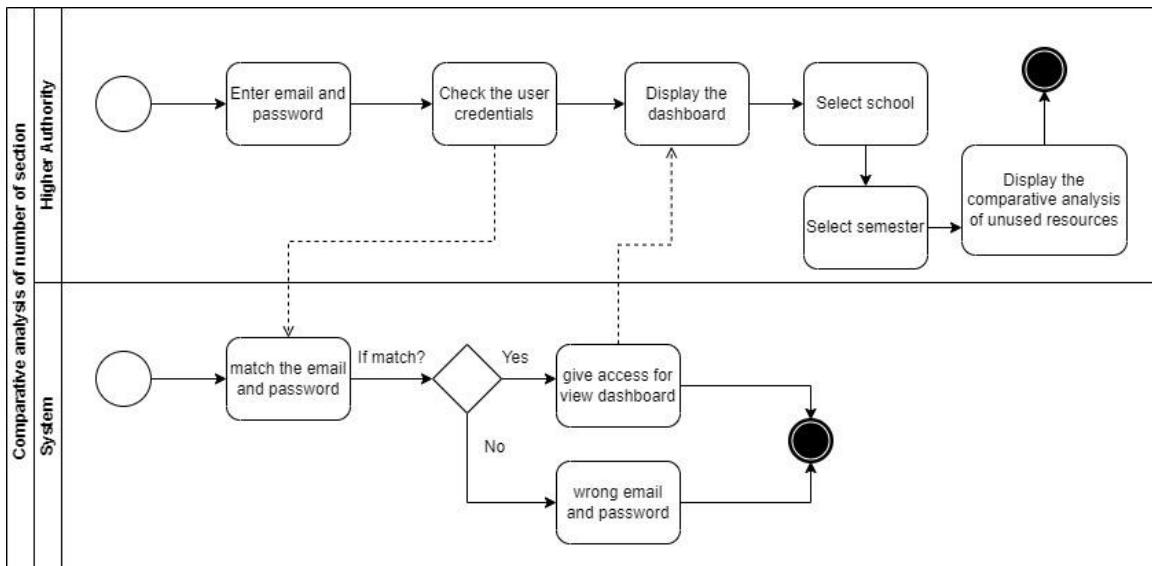


Fig 12: Comparative analysis of unused resources

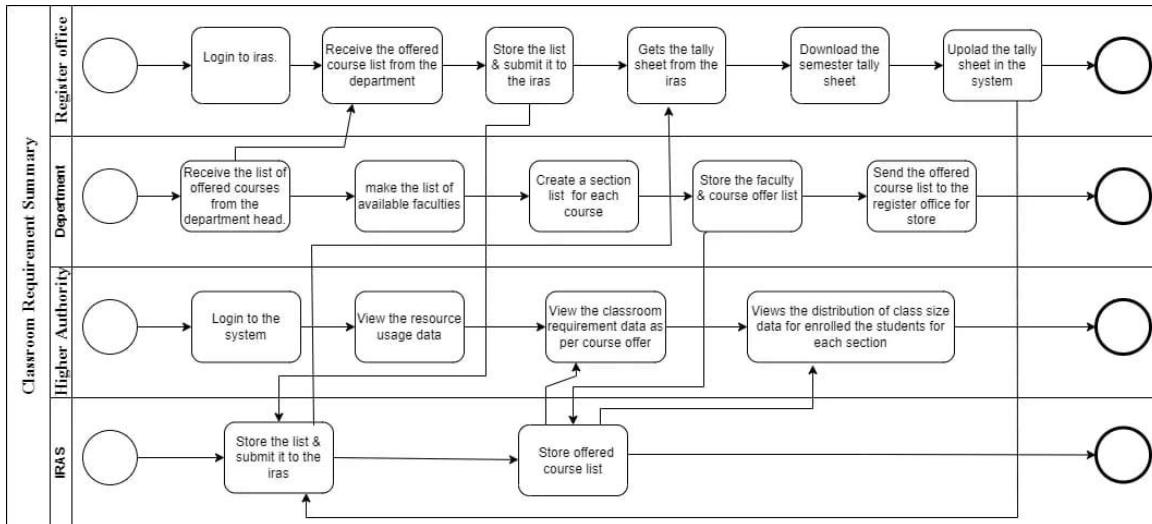


Fig 13: Classroom requirement summary

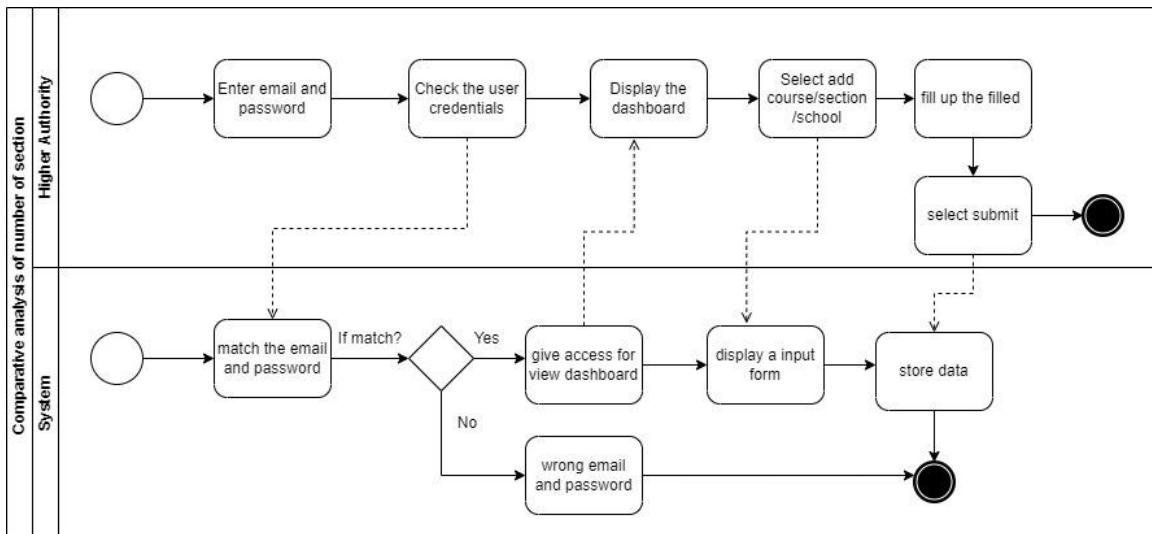


Fig 14: Input the data into the system

CHAPTER 3: LOGICAL DESIGN

A: ENTITY RELATIONSHIP DIAGRAM

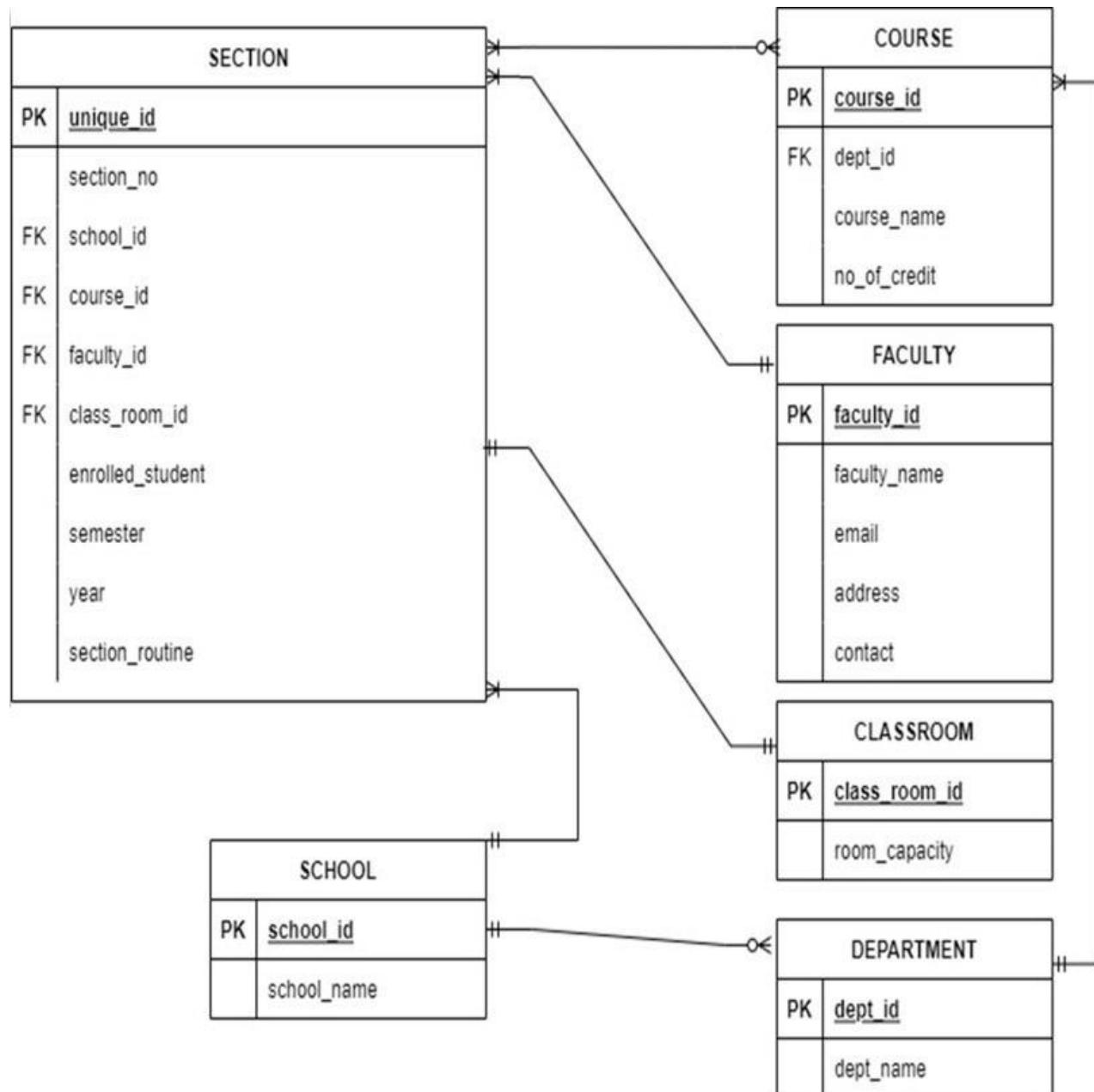


Fig 15: Entity Relationship Diagram (ERD)

B: RELATIONAL SCHEMA

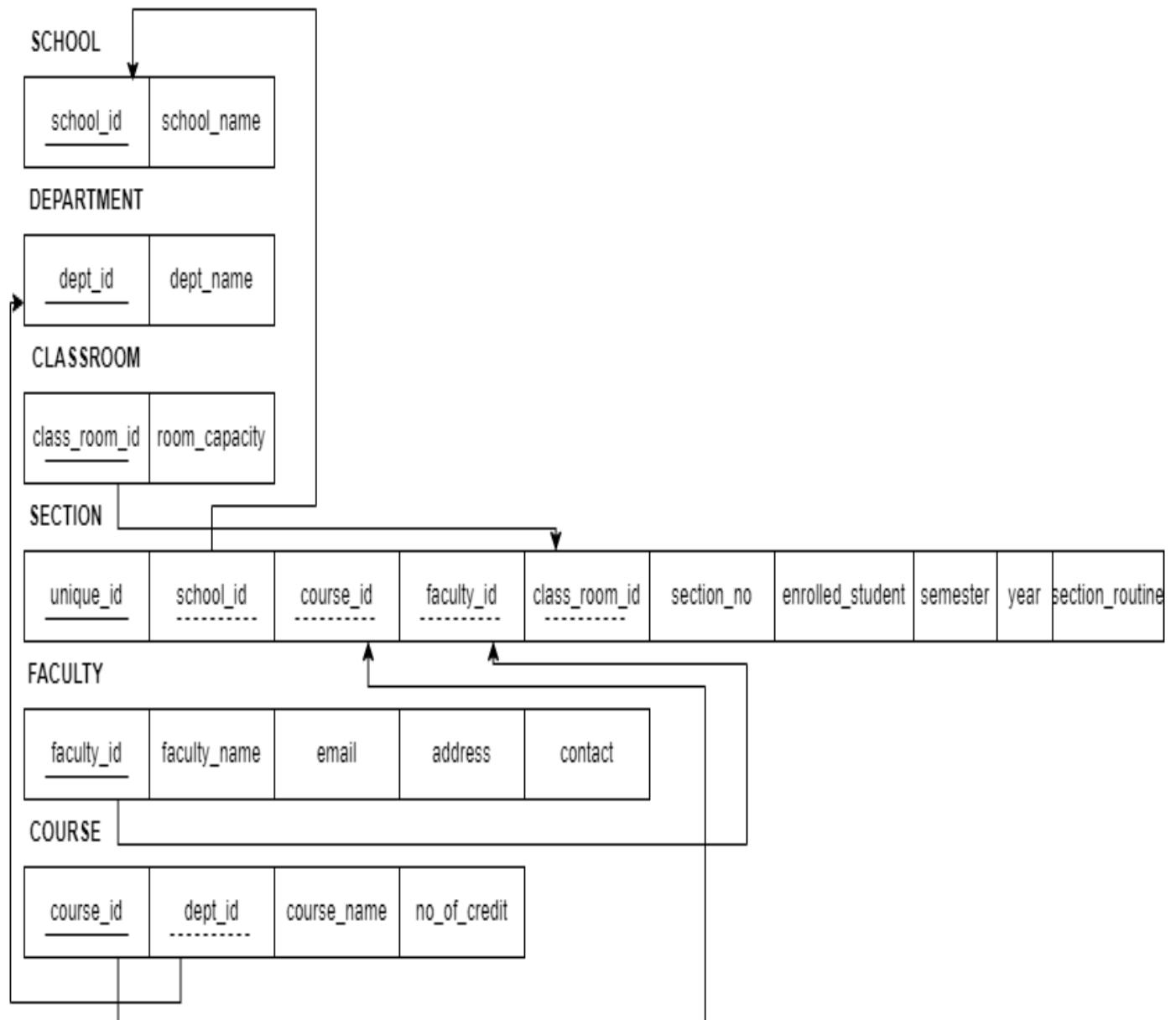


Fig 16: Relational Schema

C: NORMALIZATION

CLASSROOM	class_room_id	c1
	room_capacity	c2
SECTION	unique_id	a1
	school_id	s1
	course_id	b1
	faculty_id	f1
	classroom_id	c1
	section_no	a2
	enrolled_student	a3
	semester	a4
	year	a5
COURSE	section_routine	a6
	course_id	b1
	department_id	d1
	course_name	b2
DEPARTMENT	no_of_credits	b3
	dept_id	d1
SCHOOL	dept_name	d2
	school_id	s1
FACULTY	school_name	s2
	faculty_id	f1
	faculty_name	f2
	email	f3

	address	f4
	contact	f5
s1	s2	
d1	d2	
c1	c2	
a1	s1, b1, f1, c1, a2, a3, a4, a5, a6	
f1	f2, f3, f4, f5	
b1	b2, b3, d1	

school_id	school_name
dept_id	dept_name
class_room_id	room_capacity
unique_id	school_id, course_id, faculty_id, classroom_id, section_no, enrolled_student, semester, year, section_routine
faculty_id	faculty_name, email, address, contact
course_id	course_name, no_of_credit, dept_id

1NF:

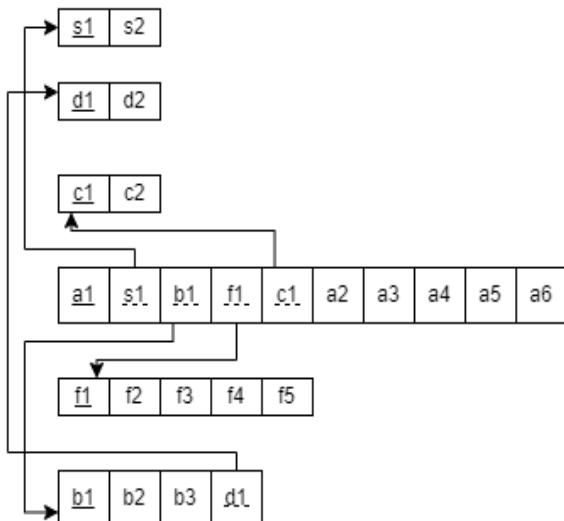
...

<u>a1</u>	s1	b1	f1	c1	a2	a3	a4	a5	a6	s2	d2	c2	f2	f3	f4	f5	b2	b3
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

2NF:

s1	b1	f1	c1	a2	a3	a4	a5	a6	s2	d2	c2	f2	f3	f4	f5	b2	b3
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

↑
a1

3NF:**BCNF:**

No non-key can identify any primary key or part of the primary key. Therefore, all the relations are in BCNF.

D: DATA DICTIONARY

SCHOOL_T

Name	Data type	Size	Remark
school_id	VARCHAR	7	PK of the table, it represents the school code which is assigned by the university. e.g., "SETS".
school_name	VARCHAR	50	Full name of the school e.g., " School of Engineering, Technology & Sciences "

DEPARTMENT_T

Name	Data type	Size	Remark
dept_id	VARCHAR	5	PK of the table, it represents the department code which is assigned by the university. e.g., "CSE".
department_name	VARCHAR	50	Name of the department e.g., "Computer Science and Engineering"

COURSE_T

Name	Data type	Size	Remark
course_id	VARCHAR	15	PK of the table, it represents the course code which is assigned by the university.
course_name	VARCHAR	30	Name of the course e.g., "Computer Architecture".
no_of_credit	INT	2	Number of credit hours earned in a course
dept_id	VARCHAR	5	FK from the Department table. e.g., "CSE".

FACULTY_T

Name	Data type	Size	Remark
faculty_id	INT	7	PK e.g., "4255". This is the university-assigned ID number of the faculty member.
faculty_name	VARCHAR	50	Name of the faculty member
email	VARCHAR	30	Email id of the faculty member. E.g., "javed@iub.edu.bd"
address	VARCHAR	30	Address of the faculty member. E.g., "Bashundhara,dhaka-1211"
contact	VARCHAR	15	Phone number of the faculty member. E.g., "01712345678".

CLASSROOM_T

Name	Data type	Size	Remark
class_room_id	VARCHAR	15	PK e.g., "BC7002-S ". This room number is assigned by the university.
room_capacity	INT	3	Room capacity of the classroom. E.g., "30"

SECTION_T

Name	Data type	Size	Remark
unique_id	VARCHAR	50	PK. Auto generated identifier
section_no	INT	3	Section of a course. e.g., "1"
school_id	VARCHAR	7	FK from the SCHOOL T. e.g., "SETS"
course_id	VARCHAR	15	FK from the COURSE T. e.g., "CSE-303"

faculty_id	INT	7	FK from the FACULTY T. e.g., "4255"
class_room_id	VARCHAR	15	FK from CLASSROOM T. e.g., "BC7002-S"
enrolled_student	INT	3	Number of enrolled students in a section. e.g., "31".
semester	VARCHAR	10	Semester of the year. e.g., "Autumn"
year	INT	4	Year of the semester. e.g., "2021"
section_routine	VARCHAR	20	It stores the class routine of the semester. e.g., "MW-12.30-2.00"

CHAPTER 4: PHYSICAL DESIGN

A: INPUT FORM

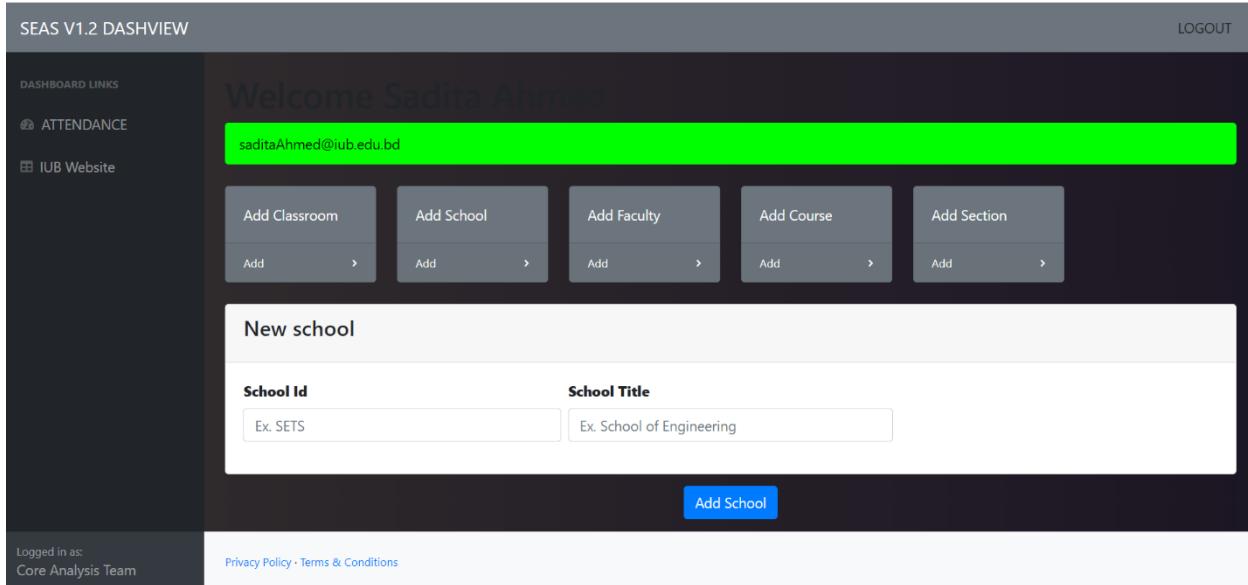
The screenshot shows the SEAS V1.2 DASHVIEW dashboard. At the top, it says "Welcome Sadita Ahmed" and displays the email "saditaAhmed@iub.edu.bd". On the left sidebar, there are "DASHBOARD LINKS" with options like "ATTENDANCE" and "IUB Website". Below the sidebar, there are five buttons: "Add Classroom", "Add School", "Add Faculty", "Add Course", and "Add Section", each with an "Add" button and a right-pointing arrow. The main content area is titled "New Classroom" and contains two input fields: "Classroom Id" (with placeholder "Ex. BC4001") and "Classroom Size" (with placeholder "Ex. 30"). A blue "Add Classroom" button is located at the bottom right of this section. At the bottom of the page, it says "Logged in as: Core Analysis Team" and provides links to "Privacy Policy · Terms & Conditions".

```
<?php
    include '../../conn.php';

    $id      = $_POST['classroom_id'];
    $size    = $_POST['classroom_size'];

    $query = "INSERT INTO classroom (classroom_id, room_capacity
) VALUES ('$id', '$size')";
    if($conn->query($query) == FALSE){
        header("Location: ../add_classroom.php?response=500");
    }

    header("Location: ../add_classroom.php?response=200");
?>
```



```

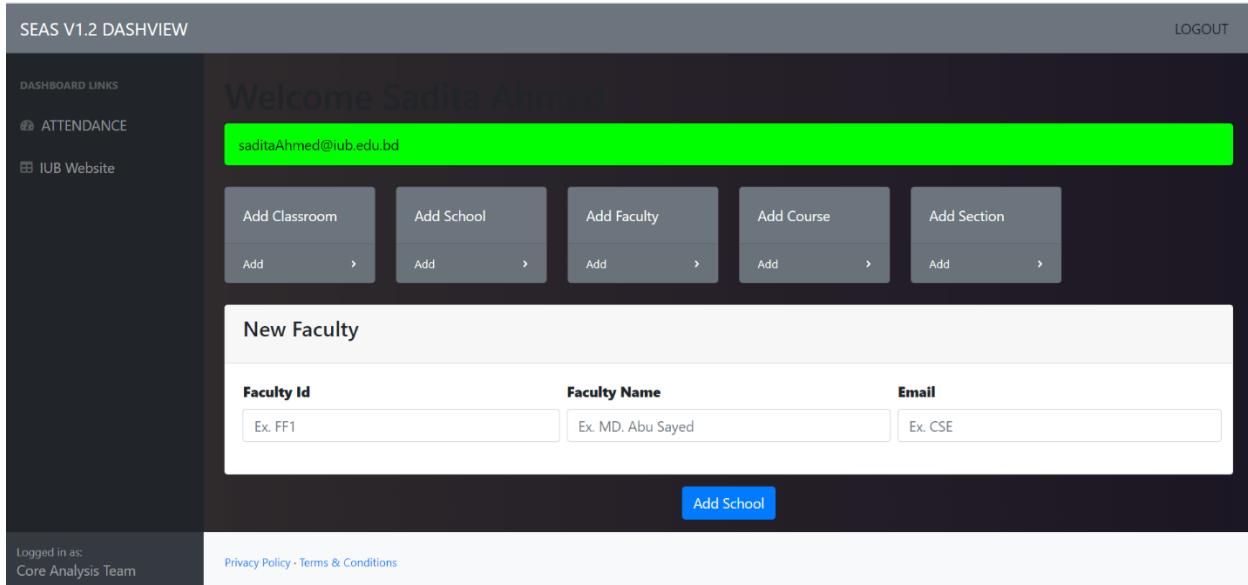
<?php
    include '../../conn.php';

    $sec      = $_POST['section_no'];
    $sem     = $_POST['section_semster'];
    $yr       = $_POST['section_yr'];
    $cap      = $_POST['section_cap'];
    $en       = $_POST['section_en'];
    $blo      = $_POST['section_bloc'];
    $st       = $_POST['section_st'];
    $et       = $_POST['section_et'];
    $day      = $_POST['section_day'];
    $room     = $_POST['section_room'];

    $query = "INSERT INTO section (cssy_id, section_no, school_id, course_id, faculty_id, classroom_id, enrolled_student , semester, year, section_routine
) VALUES ('$sec', '$sem', '$yr', '$cap', '$en', '$blo', '$st', '$et', '$day', '$room')";
    if($conn->query($query) == FALSE){
        header("Location: ../../R0/admin-add-section.php?response=500");
    }

    header("Location: ../../R0/admin-add-section.php?response=200");
?>

```



```

<?php
    include '../../conn.php';

    $id      = $_POST['faculty_id'];
    $title   = $_POST['faculty_title'];
    $email   = $_POST['faculty_email'];

    $query = "INSERT INTO faculty ( faculty_id, faculty_name,email
) VALUES ('$id', '$title','$email')";
    if($conn->query($query) == FALSE){
        header("Location: ../../add_faculty.php?response=500");
    }

    header("Location: ../../add_faculty.php?response=200");

?>

```

The screenshot shows the SEAS V1.2 DASHVIEW dashboard. At the top, it says "SEAS V1.2 DASHVIEW" and "LOGOUT". On the left sidebar, under "DASHBOARD LINKS", there are links for "ATTENDANCE" and "IUB Website". The main content area displays a welcome message "Welcome Sadita Ahmed" and an email "saditaAhmed@iub.edu.bd". Below this, there are five buttons: "Add Classroom", "Add School", "Add Faculty", "Add Course", and "Add Section", each with an "Add" button and a right-pointing arrow. A modal window titled "New Couese" is open, containing fields for "Course Id" (Ex. CSE101), "Course Name" (Ex. Introduction to Computer Science), and "Credit Hour" (Ex. 3). At the bottom of the modal is a blue "Add Course" button.

```
<?php
    include '../conn.php';

    $id   = $_POST['course_id'];
    $name      = $_POST['course_name'];
    $credit     = $_POST['course_credit'];

    $query = "INSERT INTO course (course_id, course_name, no_of_credit) VALUES ('$id', '$name', '$credit')";

    if($conn->query($query) == FALSE){
        header("Location: ../add_course.php?response=500");
    }

    header("Location: ../add_course.php?response=200");
?>
```

SEAS V1.2 DASHVIEW

LOGOUT

DASHBOARD LINKS

Add Add Add Add Add

ATTENDANCE

IUB Website

New Section

CSSY ID	Section No	
Ex. 1	Ex. Spring	
School Id	Course Id	Faculty Id
Ex. 2022	Ex. 30	Ex. 25
Classroom Id	Enrolled	Semester
Ex. y	Ex. 8.00	Ex. 9.30
Year	Section Time	
Ex. ST	Ex. BC4001	

Logged in as:
Core Analysis Team

Add Section

```
<?php
include '../conn.php';

$sec      = $_POST['section_no'];
$sem     = $_POST['section_semster'];
$yr       = $_POST['section_yr'];
$cap      = $_POST['section_cap'];
$en       = $_POST['section_en'];
$blo      = $_POST['section_bloc'];
$st        = $_POST['section_st'];
$et        = $_POST['section_et'];
$day      = $_POST['section_day'];
$room     = $_POST['section_room'];

$query = "INSERT INTO section (cssy_id, section_no, school_id, course_id, faculty_id, classroom_id, enrolled_student , semester, year, section_routine
) VALUES ('$sec', '$sem', '$yr', '$cap', '$en', '$blo', '$st', '$et', '$day', '$room')";
if($conn->query($query) == FALSE){
    header("Location: ../R0/admin-add-section.php?response=500");
}

header("Location: ../R0/admin-add-section.php?response=200");

?>
```

B: OUTPUT FORM

--select--

Slot:

Class Size	Sections	7 slot	8 slot
1-10	125	8.93	7.81
11-20	170	12.14	10.63
21-30	161	11.5	10.06
31-35	100	7.14	6.25
36-40	115	8.21	7.19
41-50	205	14.64	12.81
51-55	46	3.29	2.88
56-65	5	0.36	0.31
Total	927	66.21	57.94

```

<table class="table">
<?php
if (isset($_POST['submit'])) {
    $semester_name = $_POST['semester_name'];
    $year = $_POST['year'];
    $sql = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student
BETWEEN '1' AND '10'";
    $result = mysqli_query($conn, $sql);
    $section_by_range = array();
    array_push($section_by_range, mysqli_num_rows($result));

    $sql2 = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student
BETWEEN '11' AND '20'";
    $result2 = mysqli_query($conn, $sql2);
    array_push($section_by_range, mysqli_num_rows($result2));

    $sql3 = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student
BETWEEN '21' AND '30'";
    $result3 = mysqli_query($conn, $sql3);
    array_push($section_by_range, mysqli_num_rows($result3));

    $sql4 = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student
BETWEEN '31' AND '35'";
    $result4 = mysqli_query($conn, $sql4);
    array_push($section_by_range, mysqli_num_rows($result4));

    $sql5 = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student
BETWEEN '36' AND '40'";
    $result5 = mysqli_query($conn, $sql5);
    array_push($section_by_range, mysqli_num_rows($result5));

    $sql6 = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student
BETWEEN '41' AND '50'";
    $result6 = mysqli_query($conn, $sql6);
    array_push($section_by_range, mysqli_num_rows($result6));

    $sql7 = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student

```

```

$sql8 = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND enrolled_student
BETWEEN '56' AND '65'";
$result8 = mysqli_query($conn, $sql8);
$array_push($section_by_range, mysqli_num_rows($result8));
$total = 0;
$seven_total = 0;
$eight_total = 0;
echo '<h1>';

echo 'Class Size Table for ' . $semester_name . ' ' . $year;

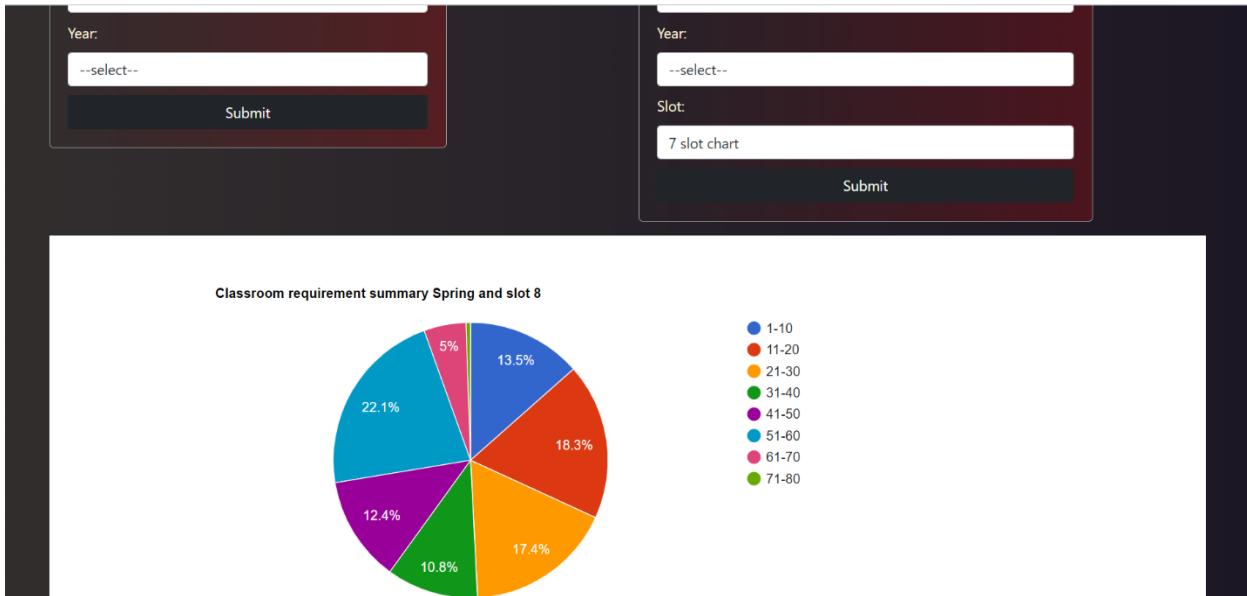
echo '</h1>';
echo '<th>Class Size</th>';
echo '<th>Sections</th>';
echo '<th>7 slot</th>';
echo '<th>8 slot</th>';
echo '</tr>';
echo '<tr>';
echo '<td>1-10</td>';
echo '<td>' . $section_by_range[0] . '</td>';
$total += $section_by_range[0];
echo '<td>' . round($section_by_range[0] / 14, 2) . '</td>';
$seven_total += round($section_by_range[0] / 14, 2);
echo '<td>' . round($section_by_range[0] / 16, 2) . '</td>';
$eight_total += round($section_by_range[0] / 16, 2);
echo '</tr>';
echo '<td>11-20</td>';
echo '<td>' . $section_by_range[1] . '</td>';
$total += $section_by_range[1];
echo '<td>' . round($section_by_range[1] / 14, 2) . '</td>';
$seven_total += round($section_by_range[1] / 14, 2);
echo '<td>' . round($section_by_range[1] / 16, 2) . '</td>';
$eight_total += round($section_by_range[1] / 16, 2);
echo '</tr>';
echo '<td>21-30</td>';
echo '<td>' . $section_by_range[2] . '</td>';

```

```

$var1 = 31;
$var2 = 35;
for ($i = 0; $i < 5; $i++) {
    echo '<tr>';
    echo '<td>' . $var1 . '-' . $var2 . '</td>';
    echo '<td>' . $section_by_range[$i + 3] . '</td>';
    $total += $section_by_range[$i + 3];
    echo '<td>' . round($section_by_range[$i + 3] / 14, 2) . '</td>';
    $seven_total += round($section_by_range[$i + 3] / 14, 2);
    echo '<td>' . round($section_by_range[$i + 3] / 16, 2) . '</td>';
    $eight_total += round($section_by_range[$i + 3] / 16, 2);
    echo '</tr>';
    if ($i == 1 or $i == 3) {
        $var1 += 5;
        $var2 += 10;
    } elseif ($i == 2) {
        $var1 += 10;
        $var2 += 5;
    } else {
        $var1 += 5;
        $var2 += 5;
    }
}
echo '<td>Total</td>';
echo '<td>' . $total . '</td>';
echo '<td>' . $seven_total . '</td>';
echo '<td>' . $eight_total . '</td>';
echo '</tr>';
}

```



```

<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
  google.charts.load('current', {'packages':['corechart']});
  google.charts.setOnLoadCallback(drawChart);

  function drawChart() {
    var data = google.visualization.arrayToDataTable([
      ['Enrolled range', 'Section'],
      <?php
      for($i=0;$i<8;$i++)
      {
        echo "[".$labels[$i].", ".$data[$i]."]";
      }
      ?>
    ]);
    var options = {
      title: 'Classroom requirement summary <?php echo $semester_name; echo " and slot "; echo $slot; ?>'
    };
    var chart = new google.visualization.PieChart(document.getElementById('piechart'));
    chart.draw(data, options);
  }
</script>
  
```

Generate Table							Generate Chart	
Semester:	<input type="text"/>		Semester:	<input type="text"/>				
Year:	<input type="text"/>		Year:	<input type="text"/>				
Submit							Submit	
Class Size	SBE	SELS	SETS	SLASS	SPPH	Total		
1-10	6	26	32	57	4	125		
11-20	12	35	36	60	27	170		
21-30	15	33	65	33	15	161		
31-35	9	10	57	16	8	100		
36-40	38	10	35	15	17	115		
41-50	99	5	33	56	12	205		
51-55	24	0	19	3	0	46		
56-65	3	0	1	1	0	5		

```

<table class="table">
    <?php
    if (isset($_POST['submit'])) {
        $semester_name = $_POST['semester_name'];
        $year = $_POST['year'];
        $range = array("1-10", "11-20", "21-30", "31-35", "36-40", "41-50", "51-55", "56-65");
        $school = array("SBE", "SELS", "SETS", "SLASS", "SPPH");
        $data = [];
        for ($i = 0; $i < sizeof($school); $i++) {
            for ($j = 0; $j < 8; $j++) {
                $val = explode("-", $range[$j]);
                $sql = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND school_id = '$school[$i]' AND enrolled_student BETWEEN '$val[0]' AND '$val[1]'";
                $result = mysqli_query($conn, $sql);
                $num_rows = mysqli_num_rows($result);
                array_push($data, $num_rows);
            }
        }
        echo '<th>' . Class Size . '</th>' . SBE . '</th>' . SELS . '</th>' . SETS . '</th>' . SLASS . '</th>' . SPPH . '</th>';
    }

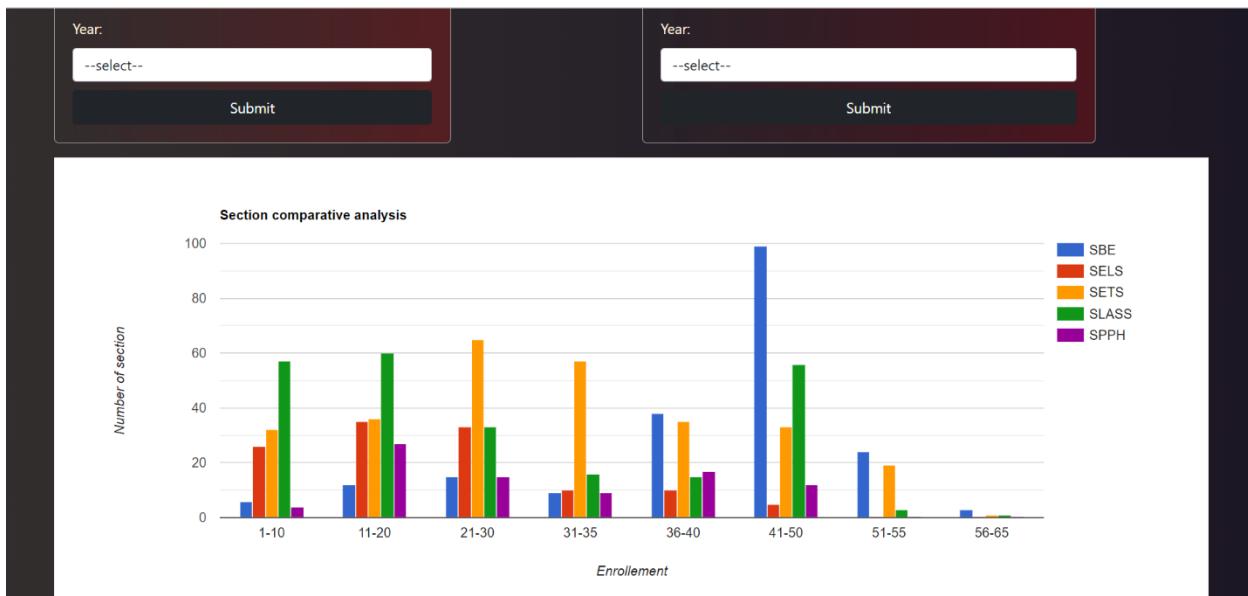
```

```

<th>
    Total
</th>;
for ($i = 0; $i < 8; $i++) {
    echo '<tr>';
    echo '<td>';
    echo $range[$i];
    echo '</td>';
    echo '<td>';
    echo $data[$i];
    echo '</td>';
    echo '<td>';
    echo $data[$i + 8];
    echo '</td>';
    echo '<td>';
    echo $data[$i + 16];
    echo '</td>';
    echo '<td>';
    echo $data[$i + 24];
    echo '</td>';
    echo '<td>';
    echo $data[$i + 32];
    echo '</td>';
    echo '<td>';
    $total = $data[$i] + $data[$i + 8] + $data[$i + 16] + $data[$i + 24] + $data[$i + 32];
    echo $total;
    echo '</td>';
    echo '</tr>';
}
}

?>
</table>

```



```

<!--
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
  google.charts.load('current', {'packages':['corechart']});
  google.charts.setOnLoadCallback(drawVisualization);

  function drawVisualization() {
    // Some raw data (not necessarily accurate)
    var data = google.visualization.arrayToDataTable([
      ['Range', 'SBE', 'SELS', 'SETS', 'SLASS', 'SPPH'],
      ['?php
        for($i=0;$i<8;$i++)
        {
          echo "[".$datax[$i].",
            ".$l2datay[$i].".$l3datay[$i].".$l4datay[$i].".$l5datay[$i].".$l6datay[$i].";
        }
      ?>
    ]);

    var options = {
      title : 'Section comparative analysis',
      vAxis: {title: 'Number of section'},
      hAxis: {title: 'Enrollement'},
      seriesType: 'bars',
      series: {5: {type: 'line'}}};

    var chart = new google.visualization.ComboChart(document.getElementById('combo_chart'));
    chart.draw(data, options);
  }
</script>
-->
```

School	Sum	Avg Enrolled	Avg Room	Difference	Unused %
SBE	8447	41	46.89	5.89	13%
SELS	2511	21.1	34.54	13.44	39%
SETS	8150	29.32	40.19	10.87	27%
SLASS	5970	24.77	36.68	11.91	32%
SPPH	2302	27.73	36.02	8.29	23%
Total Enrolled				27380	
Avg Enrolled				29.54	
Avg Room				39.67	
Difference				10.13	
Unused %				26 %	

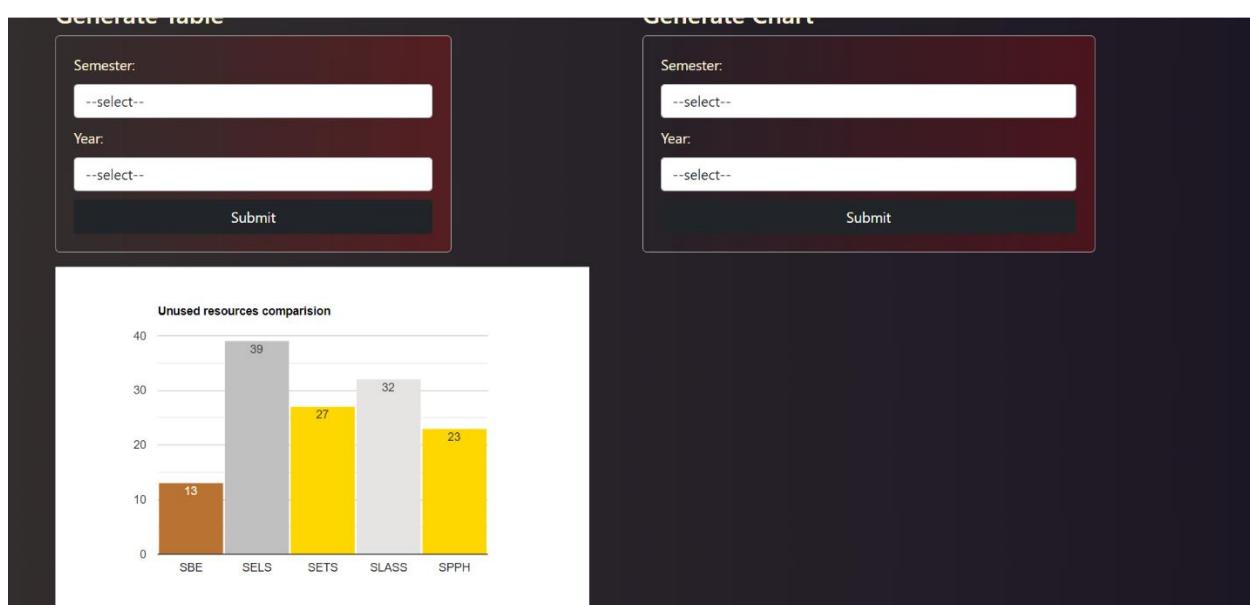
```

<?php
if (isset($_POST['submit'])) {
    $semester_name = $_POST['semester_name'];
    $year = $_POST['year'];
    // Student enrolled calculation
    $school = array("SBE", "SELS", "SETS", "SLASS", "SPPH");
    $sql = "SELECT enrolled_student FROM section WHERE semester = '$semester_name' AND year = '$year'";
    $result = mysqli_query($conn, $sql);
    $num_rows = mysqli_num_rows($result);

    $sql2 = "SELECT SUM(enrolled_student) FROM section WHERE semester = '$semester_name' AND year = '$year'";
    $result2 = mysqli_query($conn, $sql2);
    $data = mysqli_fetch_row($result2);
    // total enrolled student

    $total_enrolled = $data[0];
    $data_school_wise = [];
    $num_rows_school_wise = [];
    for ($i = 0; $i < sizeof($school); $i++) {
        $sql = "SELECT enrolled_student FROM section WHERE semester = '$semester_name' AND year = '$year' AND school_id = '$school[$i]'";
        $result = mysqli_query($conn, $sql);
        $num_rows_wise = mysqli_num_rows($result);
        array_push($num_rows_school_wise, $num_rows_wise);
        $sql2 = "SELECT SUM(enrolled_student) FROM section WHERE semester = '$semester_name' AND year = '$year' AND school_id = '$school[$i]'";
        $result2 = mysqli_query($conn, $sql2);
        $data_wise = mysqli_fetch_row($result2);
        array_push($data_school_wise, $data_wise[0]);
    }
}

```



```

<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
google.charts.load("current", {packages:['corechart']});
google.charts.setOnLoadCallback(drawChart);
function drawChart() {
  var data = google.visualization.arrayToDataTable([
    ["Element", "Density", { role: "style" } ],

    <?php
    $color = array("#b87333","silver","gold","#e5e4e2","gold");
      for($i=0;$i<5;$i++)
    {
      echo "[','".$school[$i]."', ".$percentage[$i].','".$color[$i]."',";
    }
    ?>

  ]);

  var view = new google.visualization.DataView(data);
  view.setColumns([0, 1,
      { calc: "stringify",
        sourceColumn: 1,
        type: "string",
        role: "annotation" },
      2]);

  var options = {
    title: "Unused resources comparision",
    width: 600,
    height: 400,
    bar: {groupWidth: "95%"},
    legend: { position: "none" },
  };
  var chart = new google.visualization.ColumnChart(document.getElementById("column_chart"));
  chart.draw(view, options);
}
</script>

```

Generate Table

Semester:

Year:

Generate Chart

Semester:

Year:

Semester	SBE	SELS	SETS	SLASS	SPPH	Total
Spring20	211	124	283	246	88	952
Summer20	212	125	284	247	89	957
Autumn20	213	126	285	248	90	962
Spring21	214	127	286	249	91	967
Summer21	215	128	287	250	92	972
Autumn21	216	129	288	251	93	977
Spring22	217	130	289	252	94	982
Summer22	218	131	290	253	95	987

```
<?php
if (isset($_POST['submit'])) {
    $semester_name = $_POST['semester_name'];
    $year = $_POST['year'];
    $range = array("Spring20", "Summer20", "Autumn20", "Spring21", "Summer21", "Autumn21", "Spring22", "Summer22");
    $school = array("SBE", "SELS", "SETS", "SLASS", "SPPH");
    $data = [];
    for ($i = 0; $i < sizeof($school); $i++) {
        for ($j = 0; $j < 8; $j++) {
            $val = explode("-", $range[$j]);
            $sql = "SELECT * FROM section WHERE semester = '$semester_name' AND year = '$year' AND school_id = '$school[$i]' ";
            $result = mysqli_query($conn, $sql);
            $num_rows = mysqli_num_rows($result);
            array_push($data, $num_rows);
        }
    }
}
```