



**CSE370: Database Systems**  
**Project Report**  
**Project Title: Project Title Here**

<b>Group Name: <u>Delta</u>, CSE370 Lab Section: <u>06</u>, Summer 2023</b>		
<b>ID</b>	<b>Name</b>	<b>Contribution</b>
20101618	Israt Jahan Ayshi	EER, Schema Diagram, Frontend, Backend, Report
20101021	Ms Rodsy Tahmid	EER, Schema Diagram, Frontend, Backend, Report
20201027	Sabikun Nahar	EER, Schema Diagram, Frontend, Backend, Report

## **Table of Contents**

<b>Section No</b>	<b>Content</b>	<b>Page No</b>
1	Introduction	3
2	Project Features	4
3	ER/EER Diagram	5
4	Schema Diagram	6
5	Frontend Development	7
6	Backend Development	13
7	Conclusion	22
8	References	22

## **Introduction**

The Student Management System Project is a comprehensive system made to manage tasks associated with students in a learning environment effectively. This project combines the strength of SQL for database management, HTML, CSS, Bootstrap for responsive front-end design, PHP for backend logic, and JavaScript for dynamic user interactions. This system's primary goal is to improve communication between students, instructors, staff members, and administrative personnel while streamlining managerial procedures.

Front-end technologies include HTML5, CSS, Bootstrap, and JavaScript. PHP for back-end server-side scripting. MySQL is a database used for storing and retrieving data. JavaScript interactivity for dynamic features and improved user experience.

The advantages are effectiveness, accuracy, user-friendliness, and data analysis, Firstly, By automating manual administrative processes, the system saves time and effort. Second of all, using computers to manage data decreases the chance of human error while keeping records. Users can access the system from any location that has an internet connection. Last but not least, everyone can easily navigate and engage with the system because of its intuitive interface. To make well-informed decisions, educators and administrators can examine data on student performance.

Using PHP, SQL, Bootstrap, and JavaScript, the Student Management System project provides educational institutions with a solid way to manage student-related duties efficiently, improve communication, and increase overall administrative effectiveness.

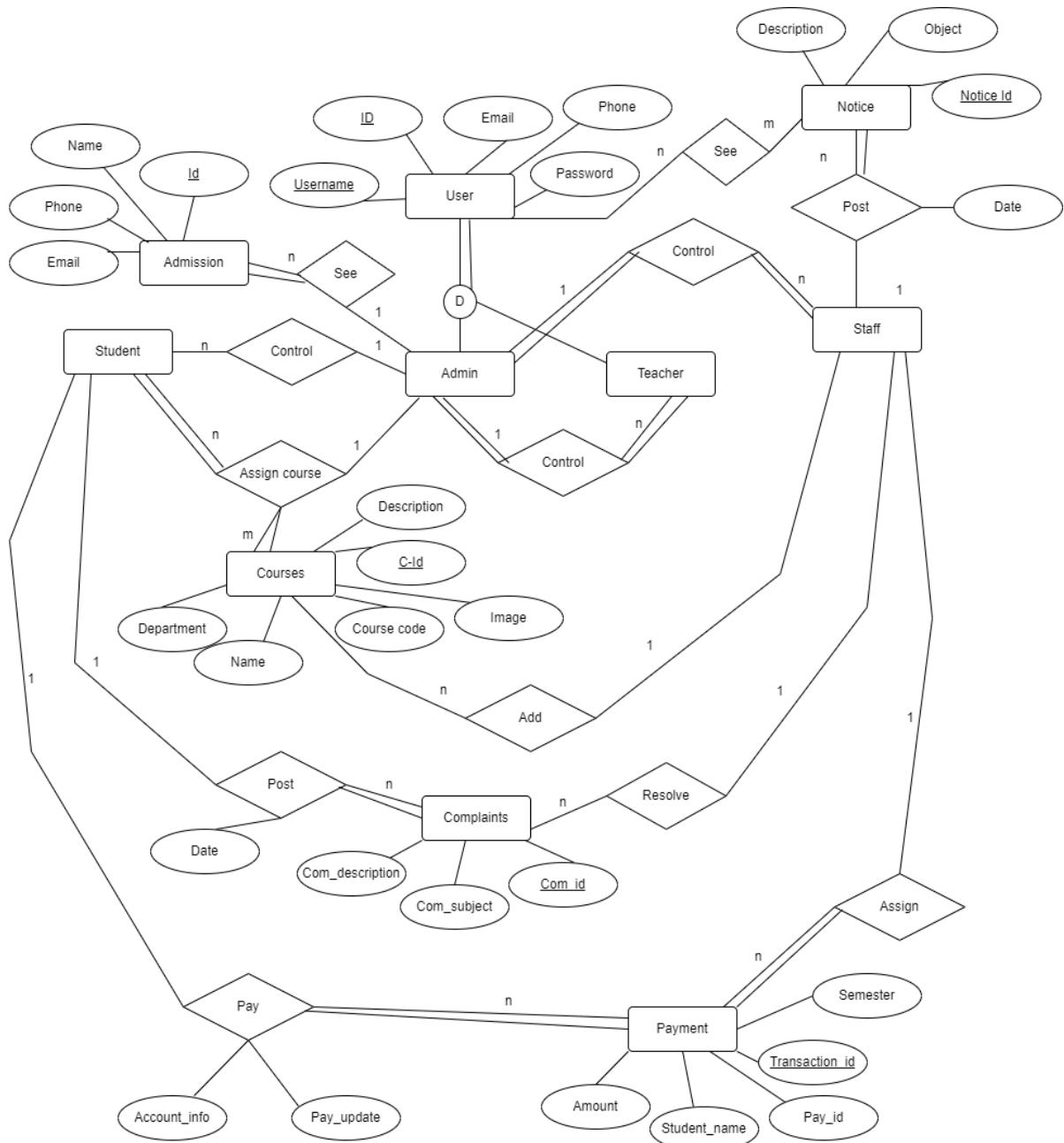
## **Project Features**

The proposed Student Management System is a small-scale database management system website project for managing user data with some other key features, such as:

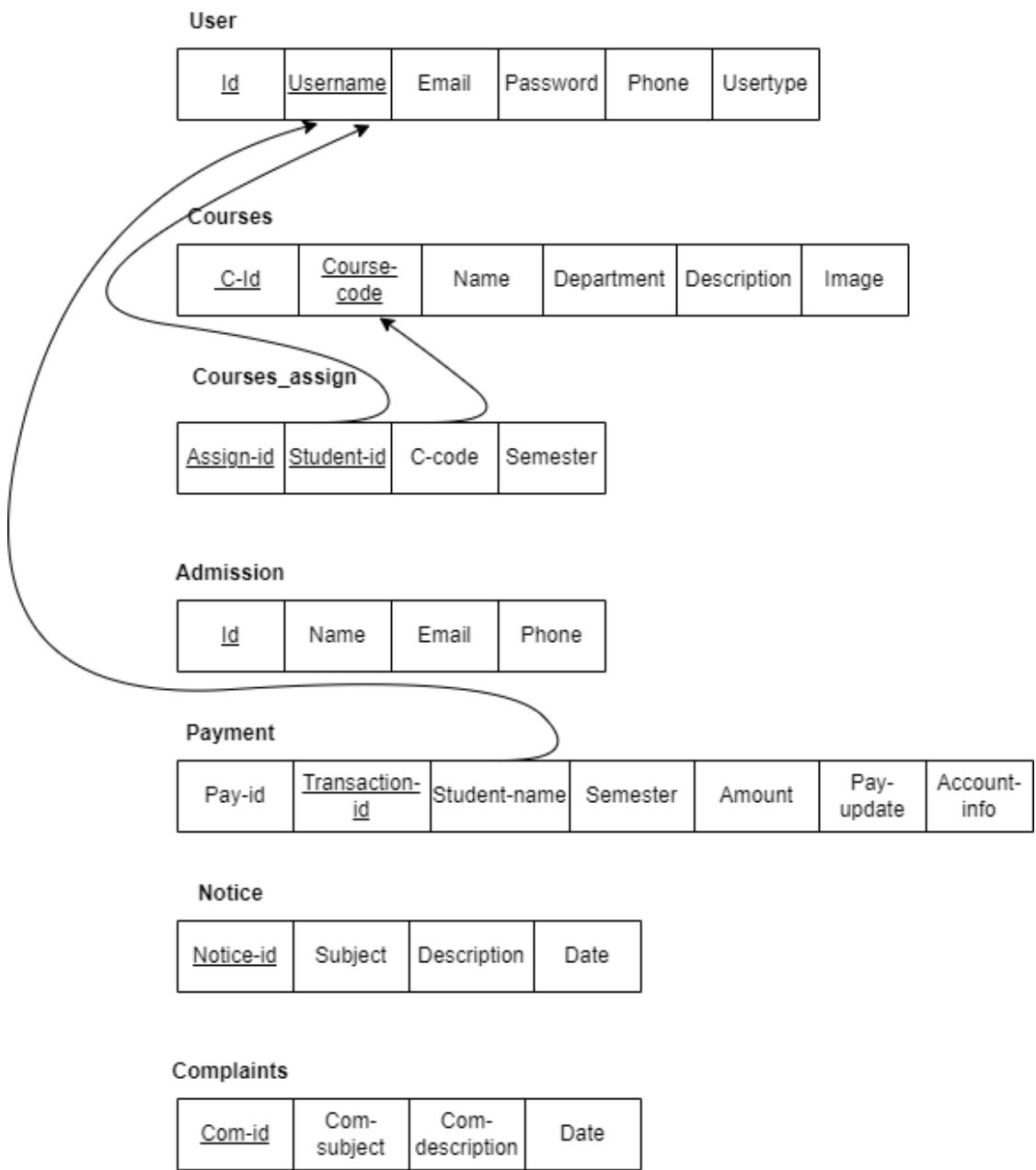
- **Homepage**
  - Login System
  - Admission form
  - Noticeboard
- **Admin Dashboard**
  - Admission
  - Add student
  - Update Student
  - View student
  - Add teacher
  - View teacher
  - Add staff
  - View staff
  - Delete students, teachers, and staff
  - Assign courses
- **Student Dashboard**
  - Update Profile
  - My courses
  - Payment
  - Post complaints
- **Teacher Dashboard**
  - Update Profile
  - Add courses
  - View courses
- **Staff Dashboard**
  - Update Profile
  - Post Notice
  - View complaints

- Resolve complaints
- Create payment
- See payment status
- Logout

## ER/EER Diagram



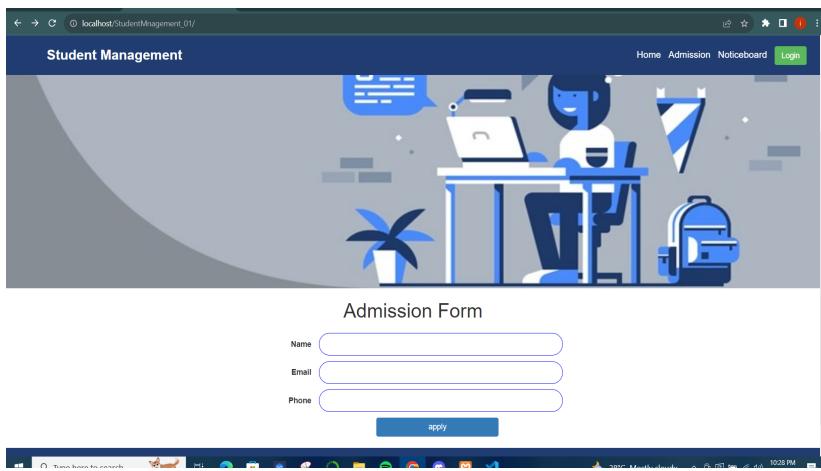
## Schema Diagram



# Frontend Development

**Contribution of ID: 20101618, Name: Israt Jahan Ayshi**

**Homepage:** The homepage, or index page, was designed using HTML, CSS, JavaScript, some PHP, and Bootstrap codes. In the navigation bar, there are four options, such as home, admission, noticeboard, and login button. In addition, the homepage contains an admission form that takes input from applicants, who can use the apply button to submit an application. The nav bar, application form, main image, and everything else were styled using CSS, and the buttons were styled using the Bootstrap btn class. The admin.css and style.css files contain most of the style elements of this project.



**Noticeboard:** The noticeboard page has a button Home in the nav bar, styled with Bootstrap btn that takes back to the homepage. The notice table was styled with the desired color, padding, height, and width using HTML and CSS wherever required.

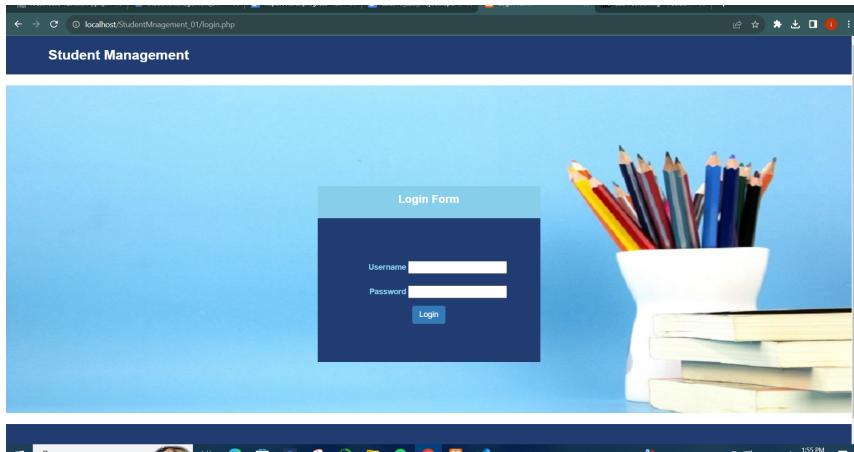
A screenshot of a web browser displaying the 'Notice Board' page. The title bar shows 'localhost/studentmanagement/view\_noticeboard.php'. The header features a dark blue navigation bar with a 'Home' button. The main content area displays a table with three rows of notices. The table has three columns: 'Notice Subject', 'Notice Description', and 'Post Date'.

Notice Subject	Notice Description	Post Date
Final Date	Summer2023 Final starts from 2nd September. Will continue till 9th September.	2023-08-25
Advising Fall2023	Your 2nd pre-advising will be held on 26th,27th,28th of August.	2023-08-25
Admission Fall2023	Admission is ongoing for Fall2023 semester. Apply with your name and other details now!	2023-08-25

Activate Windows  
Go to Settings to activate Windows.

**Login Form:** The homepage login button navigates to the Login Form page which takes username and password as input and has a button for logging in. The style.css file includes

the login page design, where margin, padding, color, font size, height, width, text decorations, and everything needed was assigned.



**Admin Dashboard:** The Admin dashboard was also designed using HTML, CSS, JavaScript, and Bootstrap. The Admin Dashboard has a navigation bar containing a title and a logout button. It also includes a sidebar that contains a list of options, each with a different hyperlink. Each option takes to a new page while the initial admin dashboard nav bar and sidebar design remain the same. The admin.css stylesheet contains all the design elements. For example:

UserName	Email	Phone	Password	Delete
Edward cullen	edwardo@gmail.com	1574862286	3465	<a href="#">Delete</a>
makesh	makeshdalle@gmail.com	1646382341	5539	<a href="#">Delete</a>
kaeya	kaeyaaab@gmail.com	1287956293	1234	<a href="#">Delete</a>
mikasa	sasageyo@gmail.com	1633642856	1234	<a href="#">Delete</a>
shamp	otter@gmail.com	1939478234	1385	<a href="#">Delete</a>

**Student Dashboard:** HTML, CSS, JavaScript, and Bootstrap were also used for frontend development in the Student Dashboard.

The screenshot shows a web browser window titled "Student Dashboard" with the URL "localhost/studentmanagement/student\_view\_courses.php". The page has a sidebar on the left with options: "My Profile", "My Courses", "Payment", and "Post Complaint". The main content area is titled "Courses Assigned to You" and contains a table with three rows. The columns are: Course Code, Course Title, Department, About Course, Image, and Semester. The data in the table is as follows:

Course Code	Course Title	Department	About Course	Image	Semester
MAT120	Integral Calculus and Differential Equations	MWB	Definitions of integration. Integration by the method of substitution. Integration by parts. Standard integrals. Integration by method of successive reduction.		Summer2023
CSE370	Database System	CSE	You will learn with database languages and popular		Summer2023
BUS201	Business and Human Communication	BSB	This course aims to teach the theory and process of communication, including barriers to effective communication, communication skills		Fall2023

At the bottom right, there is a message: "Activate Windows Go to Settings to activate Windows."

**Teacher Dashboard:** The teacher dashboard also has a similar design to the other users. Only the sidebar contains some different href choices.

The screenshot shows a web browser window titled "Teacher Dashboard" with the URL "localhost/StudentManagement\_01/teacher\_add\_courses.php". The page has a sidebar on the left with options: "My Profile", "Add Courses", and "View Courses". The main content area is titled "Add Courses" and contains a form with fields: Course Code, Course Name, Department, Description, and Image. Below the form are buttons for "Choose File" and "Add Course".

**Staff Dashboard:** The staff dashboard has some different options in the sidebar. Some designs are implemented using the same admin.css stylesheet and other style elements were added using CSS styling declarations inside cell tags.

The screenshot shows a web browser window titled "Staff Dashboard" with the URL "localhost/StudentManagement\_01/staff\_view\_complaints.php". The page has a sidebar on the left with options: "My Profile", "View Complaints", "Post Notice", "Assign Student Fee", and "Student Payment Status". The main content area is titled "All Complaints Posted by Students" and contains a table with two rows. The columns are: Student Username, Subject, Description, Date Submitted, and Resolved. The data in the table is as follows:

Student Username	Subject	Description	Date Submitted	Resolved
kaeaya	UB7 lift not working	One of the University Building 7 lift has stopped working. Please fix as soon as possible.	2023-08-25	<span style="background-color: green; color: white; padding: 2px;">Resolved</span>
maheesh	not enough seat in CSE471	Please add more seat or section for CSE471 fall2023	2023-08-27	<span style="background-color: green; color: white; padding: 2px;">Resolved</span>

localhost/StudentManagement\_01/staff\_assign\_fee.php

Staff Dashboard

Logout

Enter Semester Fee Details

Transaction\_ID:

Student Username:

Semester:

Amount:

**Assign Fee**

Lastly, all the necessary CSS, jQuery, and Javascript libraries metadata resources were linked to the system inside the HTML tag in every file.

### Contribution of ID: 20101021, Name: Ms Rodsy Tahmid

The below screen is visible once ‘View Student’ is clicked from the admin’s profile. This is achieved by implementing CSS for design, HTML for table, JavaScript for dynamic user interaction, and Bootstrap for specified color buttons like red delete buttons, and blue update buttons.

By clicking on ‘View Student’ from the sidebar from Admin Profile:

All The Student Data

UserName	Email	Phone	Password	Delete	Update
edward cullen	edwardc@gmail.com	1574862286	3465	<b>Delete</b>	<b>Update</b>
mahesh	maheshdale@gmail.com	1646382341	5539	<b>Delete</b>	<b>Update</b>
kaeya	kaeyaaib@gmail.com	1287956293	1234	<b>Delete</b>	<b>Update</b>
mikasa	sasageyo@gmail.com	1633642856	1234	<b>Delete</b>	<b>Update</b>
shamp	otter@gmail.com	1936478234	1385	<b>Delete</b>	<b>Update</b>

The view teacher and view staff page has a similar design to the add student page, where there is a delete button designed with the Bootstrap btn function.

By clicking on ‘View Teacher’ from the sidebar:

The screenshot shows a web browser window titled "Admin Dashboard" with the URL "localhost/studentmanagement/view\_teacher.php". The page has a sidebar on the left with links like "Admission", "Add Student", "View Students", etc. The main content area is titled "All The Teacher Data" and contains a table with two rows of teacher information:

UserName	Email	Phone	Password	Delete
teacher1	teacher1chad@gmail.com	1820558237	4538	<button>Delete</button>
teacher2	dave@gmail.com	193725473	1234	<button>Delete</button>

At the bottom right, there is a "Activate Windows" message: "Activate Windows Go to Settings to activate Windows."

By clicking on 'View Staffs' from the sidebar:

The screenshot shows a web browser window titled "Admin Dashboard" with the URL "localhost/studentmanagement/view\_staff.php". The sidebar links include "View Staffs". The main content area is titled "All The Staff Data" and contains a table with four rows of staff information:

UserName	Email	Phone	Password	Delete
staff1	staff01@gmail.com	1345628211	1234	<button>Delete</button>
staff2	hamudstaff@gmail.com	1376815592	1867	<button>Delete</button>
staff4	henrystaff@gmail.com	1836475832	1234	<button>Delete</button>

At the bottom right, there is a "Activate Windows" message: "Activate Windows Go to Settings to activate Windows."

From the student's profile, when the user clicks on 'My Profile', an update form page will appear as shown below which was also designed using HTML, CSS, and Javascript. The same is true for the teacher's profile and the staff's profile.

By clicking on 'My Profile' from the sidebar from Student Profile:

The screenshot shows a web browser window titled "Student Dashboard" with the URL "localhost/studentManagement\_01/student\_profile.php". The sidebar links include "My Profile". The main content area is titled "Update Profile" and contains a form with three input fields:

Email	otter@gmail.com
Phone	1936478234
Password	1385

Below the form is a "Update" button.

By clicking on ‘My Profile’ from the sidebar from Teacher Profile:

The screenshot shows the Teacher Dashboard with a dark blue sidebar on the left containing links: 'My Profile', 'Add Courses', and 'View Courses'. The main content area has a light blue header 'Update Profile'. Below it is a form with three input fields: 'Email' (teacher1chd@gmail.com), 'Phone' (1820558237), and 'Password' (4538). A blue 'Update' button is at the bottom right of the form.

By clicking on ‘My Profile’ from the sidebar from Staff Profile:

The screenshot shows the Staff Dashboard with a dark blue sidebar on the left containing links: 'My Profile', 'View Complaints', 'Post Notice', 'Assign Student Fee', and 'Student Payment Status'. The main content area has a light blue header 'Update Profile'. Below it is a form with three input fields: 'Email' (henrystaff@gmail.com), 'Phone' (1836475832), and 'Password' (1234). A blue 'Update' button is at the bottom right of the form. At the bottom right of the page, there is a watermark: 'Activate Windows. Go to Settings to activate Windows.'

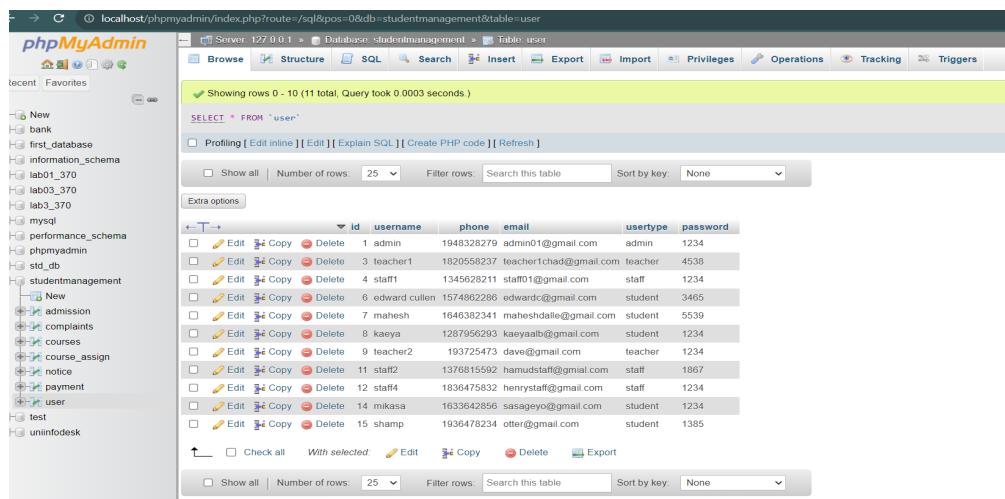
# Backend Development

## Contribution of ID: 20101618, Name: Israt Jahan Ayshi

For the backend development, PHP and SQL were used to connect to and interact with the database ‘studentmanagement’. Every time a form is submitted successfully a pop-up confirmation script will be echoed. For example:

**1. User Login System:** The homepage login button will take the user to the login form page.

Here, users can log in with their unique username and password. On the login request method, the system will select the user from the ‘user’ table in the database using PHP and SQL commands where the username matches and send them to their respective dashboards based on the user type, such as admin, student, staff or teacher.



The screenshot shows the PhpMyAdmin interface with the following details:

- Left sidebar: Shows various databases including 'New', 'bank', 'first\_database', 'information\_schema', 'labo1\_370', 'labo3\_370', 'lab3\_370', 'mysql', 'performance\_schema', 'phpmyadmin', 'std\_db', and 'studentmanagement'. The 'studentmanagement' database is selected.
- Top navigation bar: Shows 'localhost/phpmyadmin/index.php?route=/sql&pos=0&db=studentmanagement&t=table=user'.
- Current database: 'studentmanagement'.
- Table: 'user'.
- Table structure: Columns include id, username, phone, email, usertype, and password.
- Data: The table contains 11 rows of data, each with a unique id, username, phone number, email address, user type (admin, teacher, staff, student), and a password hash.
- Bottom navigation: Includes buttons for 'Check all', 'With selected', 'Edit', 'Copy', 'Delete', and 'Export'.

	id	username	phone	email	usertype	password
<a href="#">Edit</a>	1	admin	1948328279	admin01@gmail.com	admin	1234
<a href="#">Edit</a>	3	teacher1	1820558237	teacher1chad@gmail.com	teacher	4538
<a href="#">Edit</a>	4	staff1	1345628211	staff01@gmail.com	staff	1234
<a href="#">Edit</a>	6	edward cullen	1574962286	edwards@gmail.com	student	3465
<a href="#">Edit</a>	7	mahesh	1646382341	maheshdalle@gmail.com	student	5539
<a href="#">Edit</a>	8	kaeyna	1287956293	kaeynaalb@gmail.com	student	1234
<a href="#">Edit</a>	9	teacher2	193725473	dave@gmail.com	teacher	1234
<a href="#">Edit</a>	11	staff2	1376815592	hamudstaff@gmail.com	staff	1867
<a href="#">Edit</a>	12	staff4	1836475832	henrystaff@gmail.com	staff	1234
<a href="#">Edit</a>	14	mikasa	1633642856	sasageyo@gmail.com	student	1234
<a href="#">Edit</a>	15	shamp	1936478234	otteri@gmail.com	student	1385

**2. Admission:** The admission form on the homepage takes inputs and stores them in the ‘admission’ table in the database. SQL insert command inside the PHP tag was used to store these input values.

**3. Admin Dashboard:** After logging in as admin the admin can see all the admission applicants’ information in a table which was implemented using SQL select all command to fetch rows from the admission table. Admin can also add students, teachers, and staff to the database. This was implemented inside a PHP tag using the SQL insert command to store user information in the user table. Admin can also view all these user information using the view options. This particular task was implemented using SQL select, to fetch and show them inside a table. Additionally, the admin can assign existing courses in the database course table to existing students. Here, the dropdown shows the courses and students for the admin to choose from. The select and insert commands were used to implement this function. Lastly,

before each task, the system will check if the session usertype is set to admin or not using session variables inside PHP tag.

**4. Student Dashboard:** After login, the students can update their profile. Furthermore, by clicking the ‘My Courses’ option the students will be able to view the courses they were assigned to by the admin. In this case, SQL select and join commands were used to connect the ‘course\_assign’ and ‘courses’ table and then show only the information from the ‘courses’ table of the courses that particular student was assigned to. Students can also view if they were assigned transaction ID and payment details for their semester fee and choose to pay the fee. By clicking the pay fee button with each transaction ID row the user will be able to see a form where they can put their account details and make payment. By doing this the account info will be stored in the ‘payment’ table in the database and the already set unpaid payment status will be set to paid automatically. Students can also post complaints using a form that will be saved in the ‘complaints’ table inside our database.

The image consists of two screenshots. The left screenshot shows the 'Student Dashboard' with a sidebar menu ('My Profile', 'My Courses', 'Payment', 'Post Complaint') and a main section titled 'Your Semester Fee Details' showing a table with columns 'Transaction ID', 'Semester', 'Amount', and 'Payment'. One row is shown with 'transx\_2003897ch', 'spring2023', '50000', and a 'Pay Fee' button. The right screenshot shows a 'Post Your Complaint Here' form with fields for 'Student Username', 'Complaint Subject', 'About Complaint', and a 'Post Complaint' button. Below these screenshots is a screenshot of the MySQL phpMyAdmin interface showing the 'payment' table with several rows of data, including columns like 'pay\_id', 'transaction\_id', 'student\_name', 'semester', 'amount', 'pay\_update', and 'account\_info'.

**5. Teacher Dashboard:** After logging in to the teacher dashboard they can add courses by inserting a unique course code, name, department, description, and an image for the course.

One course code can be added only once and this was implemented by checking if the row count for a course code is  $\geq 1$  or not. Once the ‘add course’ button is clicked the information will be saved in the database using SQL insert command. Moreover, a teacher can also view all the course details added to the database.

c_id	course_code	name	department	description	image
3	CSE370	Database System	CSE	You will work with database languages and popular...	image/Database System.jpeg
6	BUS201	Business and Human Communication	BSS	This course aims to teach the theory and process o...	image/bss1.webp
7	MAT120	Integral Calculus and Differential Equat	MNS	Definitions of integration Integration by the met...	image/Integral Calculus.jpeg

**6. Staff Dashboard:** Staff members can view all the complaints posted by the students and push the resolved button once the concern is resolved, which will remove that complaint from the database. SQL select was used to show complaint details while the delete command was used to remove resolved complaint IDs from the database. Furthermore, staff members can post any notice using a form, these notices will be stored in the ‘notices’ table in database and appear on the noticeboard. Additionally, staff are tasked to assign payment details with unique transaction IDs to the students inside the database. Here they can choose from existing students using dropdown selection. This part brings out all the unique usernames of student usertype using select and mysqli query. The pay\_update will be set to unpaid initially and account\_info will be null inside the database. Other than these, the staff can also view the payment details of the students.

The screenshot shows the phpMyAdmin interface with the following details:

- Left Sidebar:** Shows various databases and tables under the 'studentmanagement' schema.
- Top Bar:** Includes tabs for Browse, Structure, SQL, Search, Insert, Export, Import, Privileges, Operations, Tracking, and Triggers.
- Table View:** The 'notice' table is displayed with columns: notice\_id, subject, description, and date.
- Data:**

notice_id	subject	description	date
1	Final Date	Summer2023 Final starts from 2nd September. Will c...	2023-08-25
2	Advising Fall2023	Your 2nd pre-advising will be held on 26th,27th,28...	2023-08-25
3	Admission Fall2023	Admission is ongoing for Fall2023 semester. Apply ...	2023-08-25
- Bottom Buttons:** Includes Print, Copy to clipboard, Export, Display chart, Create view, and a bookmark section.

## Contribution of ID: 20101021, Name: Ms Rodsy Tahmid

The admin can update and delete a particular student from the database by clicking on ‘View Students’ and a screen similar to the below picture will show. From there, he/she can delete any student they want by clicking on the ‘Delete’ button that is on the rightmost part of the table labeled ‘Delete’ column. Once they click ‘Delete’ of whichever student they want to remove, the entire row of that particular student will be deleted. The same is true for the ‘update’ function, here once the admin clicks on the ‘update’ button, it redirects it to the update student page which we’ve previously seen on the ‘Add Student’ sidebar.

By clicking on ‘View Student’ from the sidebar from Admin Profile:

The screenshot shows the Admin Dashboard with the following details:

- Left Sidebar:** Shows various administrative functions: Admission, Add Student, View Students, Add Teacher, View Teachers, Add Staff, View Staffs, and Assign Courses.
- Main Content:** A table titled "All The Student Data" listing student information.
- Data:**

UserName	Email	Phone	Password	Delete	Update
edward cullen	edwardc@gmail.com	1574862286	3465	<button>Delete</button>	<button>Update</button>
mahesh	maheshdalle@gmail.com	1646382341	5539	<button>Delete</button>	<button>Update</button>
kaeya	kaeyaalb@gmail.com	1287956293	1234	<button>Delete</button>	<button>Update</button>
mikasa	sasageyo@gmail.com	1633642856	1234	<button>Delete</button>	<button>Update</button>
shamp	otter@gmail.com	1936478234	1385	<button>Delete</button>	<button>Update</button>
- Top Right:** Logout button.

The top screenshot shows the 'user' table structure in 'Relation view'. It has six columns: id, username, phone, email, usertype, and password. The 'id' column is defined as int(11), 'username' as varchar(50) with a primary key constraint, 'phone' as int(30), 'email' as varchar(50), 'usertype' as varchar(50), and 'password' as varchar(50). The bottom screenshot shows the 'user' table data in 'Browse' view, displaying 11 rows of teacher and staff information.

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment		
<input type="button" value="Edit"/>		<input type="button" value="Rename"/>	<input type="button" value="Drop"/>	<input checked="" type="checkbox"/> PRIMARY	BTREE	Yes	No	id	10	A	No

Action	id	username	phone	email	usertype	password
<input type="button" value="Edit"/>	1	admin	1948328279	admin01@gmail.com	admin	1234
<input type="button" value="Edit"/>	3	teacher1	1820558237	teacher1chad@gmail.com	teacher	4538
<input type="button" value="Edit"/>	4	staff1	1345628211	staff01@gmail.com	staff	1234
<input type="button" value="Edit"/>	6	edward cullen	1574962266	edwardc@gmail.com	student	3465
<input type="button" value="Edit"/>	7	mahesh	1646392341	maheshdale@gmail.com	student	5539
<input type="button" value="Edit"/>	8	kaeysaith	1287956293	kaeysaith@gmail.com	student	1234
<input type="button" value="Edit"/>	9	teacher2	193725473	dave@gmail.com	teacher	1234
<input type="button" value="Edit"/>	11	staff2	1376815592	hamudstaff@gmail.com	staff	1867
<input type="button" value="Edit"/>	12	staff4	1836475832	henrystaff@gmail.com	staff	1234
<input type="button" value="Edit"/>	14	mikasa	1633642856	sasageyo@gmail.com	student	1234
<input type="button" value="Edit"/>	15	shamp	1936478234	otter@gmail.com	student	1385

The same update, add, and delete options are available for both the teacher and the staff from the admin panel:

By clicking on ‘View Teachers’ from the sidebar from the admin profile:

The screenshot shows the 'Admin Dashboard' with a sidebar containing links for Admission, Add Student, View Students, Add Teacher, View Teachers, Add Staff, View Staffs, and Assign Courses. The main content area displays a table titled 'All The Teacher Data' with two rows of teacher information: teacher1 and teacher2. Each row includes columns for UserName, Email, Phone, Password, and Delete. A 'Logout' button is visible in the top right corner.

UserName	Email	Phone	Password	Delete
teacher1	teacher1chad@gmail.com	1820558237	4538	<input type="button" value="Delete"/>
teacher2	dave@gmail.com	193725473	1234	<input type="button" value="Delete"/>

Activate Windows  
Go to Settings to activate Windows.

By clicking on ‘View Staffs’ from the sidebar from the admin profile:

The screenshot shows the Admin Dashboard with a sidebar on the left containing links: Admission, Add Student, View Students, Add Teacher, View Teachers, Add Staff, View Staffs, and Assign Courses. The main content area is titled "All The Staff Data" and displays a table with three rows of staff information. The table has columns for UserName, Email, Phone, Password, and Delete. The data is as follows:

UserName	Email	Phone	Password	Delete
staff1	staff01@gmail.com	1345628211	1234	<button>Delete</button>
staff2	hamudstaff@gmail.com	1376815592	1867	<button>Delete</button>
staff4	henrystaff@gmail.com	1836475832	1234	<button>Delete</button>

At the bottom right of the main area, there is a message: "Activate Windows Go to Settings to activate Windows."

From the student’s profile, they can only change their own information like email, phone, or password. This can be done by clicking on the ‘My Profile’ option from the sidebar and the below screen will be shown where the student can update their own information. However, they cannot add or delete themselves or any other student.

By clicking on ‘My Profile’ from the sidebar from Student Profile:

The screenshot shows the Student Dashboard with a sidebar on the left containing links: My Profile, My Courses, Payment, and Post Complaint. The main content area is titled "Update Profile" and contains a form with three input fields: Email, Phone, and Password. The "Email" field is populated with "otter@gmail.com", the "Phone" field with "19304782234", and the "Password" field with "1385". Below the form is a blue "Update" button.

Just like the student, a teacher member can change their own information but they cannot add or delete themselves. Neither can they add or delete any teacher or student like the admin.

By clicking on ‘My Profile’ from the sidebar from Teacher Profile:

The screenshot shows a 'Teacher Dashboard' header with a 'Logout' button. On the left, a sidebar menu includes 'My Profile' (which is currently selected and highlighted in blue), 'Add Courses', and 'View Courses'. The main content area is titled 'Update Profile' and contains three input fields: 'Email' (teacher1chad@gmail.com), 'Phone' (1820558237), and 'Password' (4538). A blue 'Update' button is at the bottom.

Just like the student and teacher, a staff member can change their own information but they cannot add or delete themselves. Neither can they add or delete any teacher or student like the admin.

By clicking on ‘My Profile’ from the sidebar from Staff Profile:

The screenshot shows a 'Staff Dashboard' header with a 'Logout' button. On the left, a sidebar menu includes 'My Profile' (selected), 'View Complaints', 'Post Notice', 'Assign Student Fee', and 'Student Payment Status'. The main content area is titled 'Update Profile' and contains three input fields: 'Email' (henrystaff@gmail.com), 'Phone' (1836475832), and 'Password' (1234). A blue 'Update' button is at the bottom. A watermark 'Activate Windows Go to Settings to activate Windows.' is visible in the bottom right corner.

### Contribution of ID: 20201027, Name: Sabikun Nahar

- Student Complaints.
- Staff can view and resolve the complaints.
- Admin can only view the complaints, but not resolve them.

#### 1. Student Complaints Dashboard:

A student can create a complaint regarding any subject. For that, the student has to go to the “Complaints” page, and then automatically his/her username and phone number will be displayed. The student has to fill up the “Subject” part of the complaint and

then fill up the “Complaint” part with his/her actual complaint. Then it will be submitted and will be stored in the “complain” table of the database.

**Student Dashboard**

**Logout**

**My Profile**

**My Courses**

**My Result**

**Complaints**

**Complaints**

Name: mikasa

Phone: 1633642856

Subject: Lecture

Complaint: Lecture notes are not provided.

Submit

**phpMyAdmin**

Recent Favorites

Server: 127.0.0.1 » Database: studentmanagement » table: complain

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Showing rows 0 - 2 (3 total. Query took 0.00002 seconds.)

SELECT \* FROM `complain`

Profile [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

Show all Number of rows: 25 Filter rows Search this table Sort by key: None

Extra options

	assign_id	email	student_id	com_description	com_subject
<input type="checkbox"/>	61	otter@gmail.com	shamp	Khaeya is bullying me.	Bullying
<input type="checkbox"/>	62	kaeyaaib@gmail.com	kaeya	Classroom is not spacious.	Space
<input type="checkbox"/>	63	sasageyo@gmail.com	mikasa	Lecture notes are not provided.	Lecture

Show all Number of rows: 25 Filter rows Search this table Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Bookmark this SQL query

Label:  Let every user access this bookmark

Bookmark this SQL query

## 2. View Complaints Part for Staff:

Here, staff can see and resolve all the complaints made by the students. If the staff has resolved the complaint, clicking the “Resolve” button will remove the complaint of the student from the complaint table of the database. The staff has full authority over resolving and fixing the complaints of the students.

Staff Dashboard

[Logout](#)

[My Profile](#)  
[Add Courses](#)  
[View Courses](#)  
[View Complaints](#)

### Student's Complaints

User Name	Email	Subject	Description	Resolve
shamp	otter@gmail.com	Bullying	Khaeya is bullying me.	<a href="#">Resolve</a>
kaeya	kaeyaalb@gmail.com	Space	Classroom is not spacious.	<a href="#">Resolve</a>
mikasa	sasageyo@gmail.com	Lecture	Lecture notes are not provided.	<a href="#">Resolve</a>

### 3. View Complaints Part for Admin:

An admin can only view the complaints made by the students. However, he can not resolve or remove the complaints from the actual “complain” table of the database. The authority for resolving the complaints goes to the staff.

Admin Dashboard

[Admission](#)  
[Add Student](#)  
[View Students](#)  
[Add Teacher](#)  
[View Teachers](#)  
[Add Staff](#)  
[View Staffs](#)  
[Assign Courses](#)  
[View Complaints](#)

### Student's Complaints

User Name	Email	Subject	Description
shamp	otter@gmail.com	Bullying	Khaeya is bullying me.
kaeya	kaeyaalb@gmail.com	Space	Classroom is not spacious.
mikasa	sasageyo@gmail.com	Lecture	Lecture notes are not provided.

## **Conclusion**

The Student Management System project successfully integrates SQL, Bootstrap, PHP, and JavaScript to create a comprehensive platform for streamlined student administration. With user-specific dashboards, the system handles tasks ranging from profile updates to payment processing and complaint submissions. Frontend development yielded visually appealing interfaces, while the backend, driven by PHP and SQL, facilitated efficient data management. Notably, the system's approach to addressing student complaints promotes transparency and quick resolution. Overall, this project exemplifies how collaboration between frontend and backend technologies can revolutionize educational administration, promising enhanced efficiency and user experiences.

## **References**

1. W3school.  
[https://www.w3schools.com/bootstrap/bootstrap\\_get\\_started.asp](https://www.w3schools.com/bootstrap/bootstrap_get_started.asp)