



College of Computer Studies  
Advanced Database Management Systems

<b>Name</b>	Russell C. Pagcu
<b>USN</b>	23000958010
<b>Course Code</b>	IT6314

**Activity Name: Student Directory Management**

**Project Overview**

The **Student Directory Management System** is a simple **CRUD-based web application** developed to assist a **college registrar** in efficiently managing student records. Built using **PHP (with PDO)**, **MySQL**, and basic **HTML/CSS/JS**, the system allows for easy **addition, viewing, editing, and deletion** of student entries. It ensures that all relevant student details such as **ID, names, program, year level, and email** are properly recorded, updated, or removed when needed.

This project is designed to provide students with **hands-on experience in full-stack development**, particularly in **linking frontend interfaces with backend logic** using **secure, prepared statements in PHP**. It also emphasizes the importance of structured code organization, user-friendly interfaces, and secure data handling.

**Database Schema**

The system uses a single table in a MySQL database named **student\_management**. Below is the SQL script used to set up the database and the **students** table:

```
CREATE DATABASE student_management;  
USE student_management;  
CREATE TABLE students (  
    student_id INT AUTO_INCREMENT PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    year_level VARCHAR(10) NOT NULL,  
    program VARCHAR(100) NOT NULL,  
    email VARCHAR(100) NOT NULL  
);
```

Each student has a unique **student\_id**, along with fields for their **personal and academic information**. Data types and constraints were defined to ensure only valid data is stored.

**CRUD Feature Walkthrough**

The system implements the four fundamental **CRUD operations**: Create, Read, Update, and Delete. Each function is handled through a dedicated form interface and a PHP backend script using **PDO with prepared statements** for secure database interaction.



College of Computer Studies  
Advanced Database Management Systems

### Create (Add New Student)

- **Page:** insert.html
- **Script:** records.php
- **Main Coding Schema:** \$stmt = \$conn->prepare("INSERT INTO students (first\_name, last\_name, year\_level, program, email) VALUES (:first\_name, :last\_name, :year\_level, :program, :email)");
- **Process:** Users access a clean and responsive form where they enter student details. Upon submission, the data is sent via POST to **records.php**, which validates inputs and uses a prepared PDO statement to insert the new record into the **students** table. A confirmation message is displayed after successful insertion. This approach ensures data integrity and protects against SQL injection.

### Read (View Student Records)

- **Pages:** records.html (for viewing all records) and retrieve.html (for retrieving a specific record by ID)
- **Scripts:** show.php and retrieve.php
- **Main Coding Schema:** SELECT \* FROM students WHERE student\_id = :student\_id
- **Process:** The **records.html** page embeds **show.php** in an iframe, which executes a SELECT query to fetch and display all student records in a structured table. For more specific searches, **retrieve.html** allows the user to input a **student ID**, which is then handled by **retrieve.php**. This script uses a secure SELECT statement to retrieve and display the matching student. If no match is found, an appropriate message is shown. Both read functions ensure clean presentation and meaningful feedback.

### Update (Modify Existing Student)

- **Page:** update.html
- **Script:** update.php
- **Main Coding Schema:** UPDATE students SET first\_name = :first\_name, last\_name = :last\_name, year\_level = :year\_level, program = :program, email = :email WHERE student\_id = :student\_id
- **Process:**  
Users enter the **student ID** along with any fields they wish to update. The system first retrieves the existing record using the ID. Any new



College of Computer Studies  
Advanced Database Management Systems

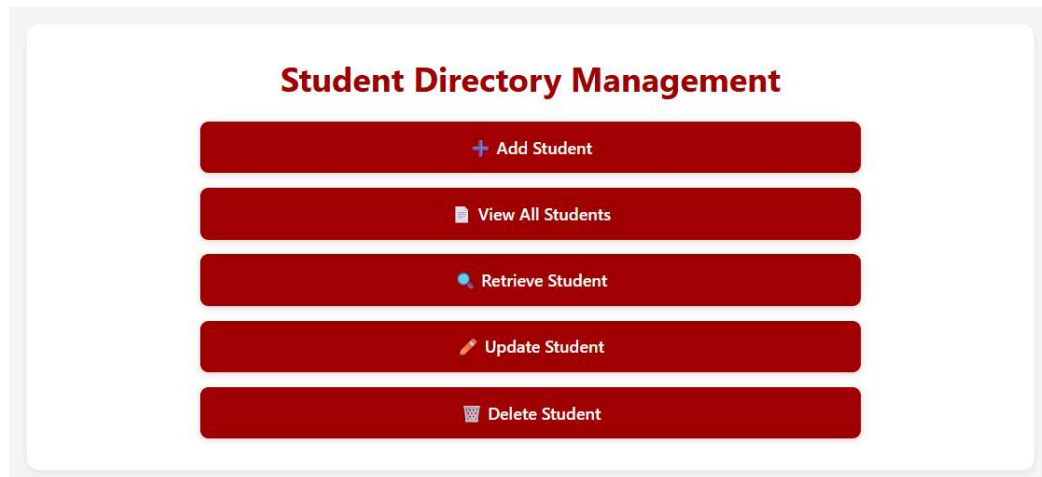
values submitted are used to replace the old ones, while unchanged fields retain their original values. The script uses a PDO **UPDATE** query to save the changes. This method enables **partial updates**, improves flexibility, and avoids overwriting data unnecessarily.

### Delete (Remove Student Record)

- **Page:** delete.html
- **Script:** delete.php
- **Main Coding Schema:** DELETE FROM students WHERE student\_id = :student\_id
- **Process:** Users provide a **student ID** through a simple form. The value is passed via POST to **delete.php**, which runs a DELETE query using a prepared statement. If a matching record exists, it is removed from the database and a success message is displayed. If no record matches, the user is informed. This process ensures that data removal is deliberate, controlled, and traceable.

### Screenshots of the App in Action

#### 1. Homepage - Main Menu (index.html)



**Annotation:** This screenshot shows the **homepage (index.html)**, which acts as the **central navigation menu** for the Student Directory Management System. From here, users can access all core CRUD functionalities: **Add Student**, **View All Records**, **Retrieve by ID**, **Update Existing Record**, and **Delete Entry**. Each link routes the user to a specific HTML form or interface that interacts with its respective backend PHP logic using **modular file structure and clean UI design**.



College of Computer Studies  
Advanced Database Management Systems

## 2. Insert Form (insert.html)

### Add New Student

First Name:

Last Name:

Year Level:

Program:

Email:

[Add Student](#)

[Back to Menu](#)

### Insert Student Record

Student record added successfully. ✓

[Add Another](#) [Main Menu](#)

After the insertion of the **Data**, a message will appear that it was successfully added to the back-end.

	student_id	first_name	last_name	year_level	program	email
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	Gillian	Florano	2nd Year	Bachelor of Science in Chemistry	gillian.florano@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Russell	Pagcu	2nd Year	Bachelor of Science in Information Technology	russell.pagcu@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	Jaira	Canteras	3rd Year	Bachelor of Science in Nursing	jai.canteras@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Justine	Placio	4th Year	Bachelor of Science in Computer Science	justine.placio@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	Micaella	Eliab	5th Year	Bachelor of Science in Architecture	micaella.eliab@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	Andres	David	1st Year	Bachelor of Science in Economics	andres.david@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	Asnaira	Usman	2nd Year	Bachelor of Science in Internation Tourism Managem...	asnaira.usman@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	Hermione	Benitez	3rd Year	Bachelor of Science in Cybersecurity	hermione.benitez@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	Myka Ella	Tumesa	4th Year	Bachelor of Science in Medical Technology	myka.tumesa@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	31	Stephany	Ortega	2nd Year	Bachelor of Science in Accountancy	stephany.ortega@gmail.com

Here we can conclude that the information we inputted was added on the back-end (local host).



College of Computer Studies  
Advanced Database Management Systems

**Annotation:** This screenshot shows the **Create operation** where the user submits a new student record using the **insert.html** form. The fields include First Name, Last Name, Year Level, Program, and Email. The success message confirms the data has been inserted into the database using a **PDO INSERT statement**.

## 2. All Records Table (records.html - show.php)

Student Records					
ID	First Name	Last Name	Year Level	Program	Email
1	Gillan	Florano	2nd Year	Bachelor of Science in Chemistry	gillian.florano@gmail.com
2	Russell	Pagcu	2nd Year	Bachelor of Science in Information Technology	russell.pagcu@gmail.com
3	Jaira	Canteras	3rd Year	Bachelor of Science in Nursing	jai.canteras@gmail.com
4	Justine	Placio	4th Year	Bachelor of Science in Computer Science	justine.placio@gmail.com
5	Micaella	Eliab	5th Year	Bachelor of Science in Architecture	micaella.eliab@gmail.com
6	Andres	David	1st Year	Bachelor of Science in Economics	andres.david@gmail.com
7	Asnaira	Usman	2nd Year	Bachelor of Science in Internation Tourism	asnaira.usman@gmail.com

[Back to Menu](#)

**Annotation:** This screenshot demonstrates the **Read operation** for viewing all student entries. The data is fetched from the **students** table using a **SELECT \* query** and displayed inside a styled HTML table. The records are embedded using an **iframe** to call the **show.php** script.

## 3. Retrieve Record (retrieve.html - retrieve.php)

Retrieve Student Record					
Enter Student ID:					
<input type="text" value="6"/>					
<a href="#">Retrieve</a>					
<a href="#">Back to Menu</a>					

Student Record					
ID	First Name	Last Name	Year Level	Program	Email
6	Andres	David	1st Year	Bachelor of Science in Economics	andres.david@gmail.com

[Back](#)



College of Computer Studies  
Advanced Database Management Systems

**Annotation:** Here, the user performs a **Read operation** by retrieving a specific student using their **student ID**. The form in **retrieve.html** sends the input via POST to **retrieve.php**, which displays a single matching record using a **SELECT WHERE** query. If no match is found, a warning is shown.

#### 4. Update Form (update.html - update.php)

<div><div><div></div><div>→</div></div></div>			▼ student_id	first_name	last_name	year_level	program	email
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		1	Gillan	Florano	2nd Year	Bachelor of Science in Chemistry	gillian.florano@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		2	Russell	Pagcu	2nd Year	Bachelor of Science in Information Technology	russell.pagcu@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		3	Jaira	Canteras	3rd Year	Bachelor of Science in Nursing	jai.canteras@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		4	Justine	Placio	4th Year	Bachelor of Science in Computer Science	justine.placio@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		5	Micaella	Eliab	5th Year	Bachelor of Science in Architecture	micaella.eliab@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		6	Andres	David	1st Year	Bachelor of Science in Economics	andres.david@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		7	Asnaira	Usman	2nd Year	Bachelor of Science in Internation Tourism Managem...	asnaira.usman@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		8	Hermione	Benitez	3rd Year	Bachelor of Science in Cybersecurity	hermione.benitez@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		10	Myka Ella	Tumesa	4th Year	Bachelor of Science in Medical Technology	myka.tumesa@gmail.com
<input type="checkbox"/>	<div><div><div><div></div><div>Edit</div></div><div><div></div><div>Copy</div></div><div><div></div><div>Delete</div></div></div></div>		31	Stephany	Ortega	2nd Year	Bachelor of Science in Accountancy	stephany.ortega@gmail.com

As you can see, this is the **original data** that I have on the table in the back-end.

### Update Student Record

Student ID:

New First Name:

New Last Name:

New Year Level:

New Program:

New Email:

Update Student

Back to Menu

### Update Student Record

Student record updated successfully. ✓

Update Another

Main Menu





College of Computer Studies  
Advanced Database Management Systems

Here, I **changed** the **program** of **student\_id = 1**, after that I got a successful message saying that it was updated on the back-end.

	student_id	first_name	last_name	year_level	program	email
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	Gillian	Florano	2nd Year	Bachelor of Science in Biochemistry	gillian.florano@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Russell	Pagcu	2nd Year	Bachelor of Science in Information Technology	russell.pagcu@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	Jaira	Canteras	3rd Year	Bachelor of Science in Nursing	jai.canteras@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Justine	Placio	4th Year	Bachelor of Science in Computer Science	justine.placio@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	Micaella	Eliab	5th Year	Bachelor of Science in Architecture	micaella.eliab@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	Andres	David	1st Year	Bachelor of Science in Economics	andres.david@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	Asnaira	Usman	2nd Year	Bachelor of Science in Internation Tourism Managem...	asnaira.usman@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	Hermione	Benitez	3rd Year	Bachelor of Science in Cybersecurity	hermione.benitez@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	Myka Ella	Tumesa	4th Year	Bachelor of Science in Medical Technology	myka.tumesa@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	31	Stephany	Ortega	2nd Year	Bachelor of Science in Accountancy	stephany.ortega@gmail.com

Upon checking on the back-end table, I can conclude that it is working. The information that I updated was successful.

**Annotation:** This screenshot illustrates the **Update operation**, where the user updates a student's record by submitting their ID and new details. Fields left blank are ignored. The backend logic retrieves the existing values and selectively updates only those that were modified using a **PDO UPDATE query**.

## 5. Delete Student (delete.html - delete.php)

	student_id	first_name	last_name	year_level	program	email
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	Gillian	Florano	2nd Year	Bachelor of Science in Biochemistry	gillian.florano@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Russell	Pagcu	2nd Year	Bachelor of Science in Information Technology	russell.pagcu@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	Jaira	Canteras	3rd Year	Bachelor of Science in Nursing	jai.canteras@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Justine	Placio	4th Year	Bachelor of Science in Computer Science	justine.placio@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	Micaella	Eliab	5th Year	Bachelor of Science in Architecture	micaella.eliab@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	Andres	David	1st Year	Bachelor of Science in Economics	andres.david@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	Asnaira	Usman	2nd Year	Bachelor of Science in Internation Tourism Managem...	asnaira.usman@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	Hermione	Benitez	3rd Year	Bachelor of Science in Cybersecurity	hermione.benitez@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	Myka Ella	Tumesa	4th Year	Bachelor of Science in Medical Technology	myka.tumesa@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	31	Stephany	Ortega	2nd Year	Bachelor of Science in Accountancy	stephany.ortega@gmail.com

As you can see, this is the **original data** that I have on the table in the back-end.

### Delete Student Record

Enter Student ID to Delete:

Delete Student

Back to Menu

### Delete Student Record

Student with ID 31 has been deleted. ☒

Delete AnotherMain Menu



College of Computer Studies  
Advanced Database Management Systems

Here, I was targeting to **delete** the **data** of the **student with id = 31**. After clicking the delete button, I got a successful message saying that it was deleted in the back-end table.

	student_id	first_name	last_name	year_level	program	email
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	Gillian	Florano	2nd Year	Bachelor of Science in Biochemistry	gillian.florano@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Russell	Pagcu	2nd Year	Bachelor of Science in Information Technology	russell.pagcu@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	Jaira	Canteras	3rd Year	Bachelor of Science in Nursing	jai.canteras@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Justine	Placio	4th Year	Bachelor of Science in Computer Science	justine.placio@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	5	Micaella	Eliab	5th Year	Bachelor of Science in Architecture	micaella.eliab@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	6	Andres	David	1st Year	Bachelor of Science in Economics	andres.david@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	7	Asnaira	Usman	2nd Year	Bachelor of Science in Internation Tourism Managem...	asnaira.usman@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	8	Hermione	Benitez	3rd Year	Bachelor of Science in Cybersecurity	hermione.benitez@gmail.com
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	10	Myka Ella	Tumesa	4th Year	Bachelor of Science in Medical Technology	myka.tumesa@gmail.com

Upon checking, I can therefore conclude that it is working. The student with id = 31 which I was targeting to delete was successfully removed from the table.

**Annotation:** This image shows the **Delete operation**, where a student record is removed from the database based on the provided ID. The user enters the ID in **delete.html**, and the script delete.php uses a **PDO DELETE statement**. A confirmation or error message is shown based on the result.

### Challenges Encountered and Solutions

#### **1. Database Connection Issues**

**Issue:** Initial PDO connection failed due to missing DB or wrong credentials.

**Fix:** Added try-catch block and verified MySQL configuration in XAMPP. Used correct dbname and default root user with no password.

#### **2. Empty Inputs in Update**

**Issue:** Overwrote values with blanks during partial updates.

**Fix:** Fetched original values first, then used updated fields only if not empty.

#### **3.Record Not Found on Delete/Retrieve**

**Issue:** Users tried to retrieve/delete IDs that didn't exist.

**Fix:** Added feedback messages to inform users of unsuccessful actions.

### Learnings & Reflections

Building this project gave me practical experience with the complete web development process, from designing a structured and efficient database to implementing secure and organized back-end logic, and finally integrating it with a clean and functional front-end interface. I gained a deeper understanding of how PHP with PDO handles data securely through prepared statements, and how proper file structuring contributes to maintainability. Through debugging and testing, I improved my ability to identify and resolve issues across both the server and client sides. Most importantly, I learned that





College of Computer Studies  
Advanced Database Management Systems

even a basic CRUD application requires careful coordination between every layer of the stack to ensure data integrity, user experience, and scalability. This project strengthened my foundation and prepared me to take on more advanced web development challenges with confidence.