National Teachers College
since 1928

School of Arts, Science & Technology
629 J. Nepomuceno, Quiapo, Manila

"SDG 4 - Quality Education: Attendance Monitoring System Project Report"

Bachelor of Science in Information & Technology

Submitted by:
Cledera, Anthony
Peñaver, Mark Joseph
Popes, Nathaniel

Submitted to:
Justin Louise R. Neypes

Dec 12, 2025

I. Introduction

1.1 Project Overview & UN SDG Target

This project implements a Student Attendance Monitoring System that records daily attendance, maintains student profiles (including total absences), manages teacher and class information, and produces attendance reports. The system supports accurate tracking of student participation and helps institutions identify absenteeism patterns to improve learning outcomes.

UN SDG alignment: SDG 4 — Quality Education. By improving attendance tracking and reporting, the system supports interventions that increase school retention, instructional quality, and inclusive access to learning.

1.2 Problem Statement

Many schools rely on paper or fragmented spreadsheets for attendance. These approaches lead to:

- Inconsistent records and lost data
- Difficulty in generating timely reports
- Manual computation of total absences leading to human error

This system solves the data problem by providing a relational DB-backed application that ensures data integrity (PK/FK constraints), automates absence counts, and enables reliable queries and reports by date, class, or student.

II. Requirements & Analysis

2.1. Functional Requirements and Non-Functional Requirements

Functional Requirements

FR1: System shall allow recording of student attendance with status (Present, Absent, Late, Excused).

FR2: System shall maintain student records including name, section, and total absences.

FR3: System shall maintain teacher and class information.

FR4: System shall automatically update total_absent in the students table when a new “Absent” record is inserted.

FR5: System shall generate attendance reports using predefined VIEWs.

FR6: System shall enforce referential integrity using primary and foreign keys.

FR7: System shall support data retrieval for any date, class, or student.

Non-Functional Requirements

NFR1 – Accuracy: The system must reduce attendance errors through constraints and triggers.

NFR2 – Reliability: All attendance data must be consistent across all related tables.

NFR3 – Security: Only valid foreign key references are allowed.

NFR4 – Scalability: System must support multiple classes and hundreds of students.

NFR5 – Maintainability: Normalized schema ensures easy updates and low redundancy.

NFR6 – Performance: Queries and view generation must respond within seconds.

2.2. Data Requirements (Description of input data structure and size)

Student table

- 50 student records
- Student_id, first_name, last_name, student_name, total _absent

Attendance table

- 50 attendance records
- Contains student_id FK and class_id FK

Teacher table

- 2 class entries
- Full_name and subject

2.3. Schema Normalization Analysis:

The database is normalized up to Third Normal Form (3NF).

Functional Dependencies

STUDENTS

- student_id → first_name, last_name, student_section, total_absent

  No partial or transitive dependencies.

  → 3NF

CLASSES

- class_id → class_name, teacher_id

  → 3NF because teacher_id is a foreign key, not a derived attribute.

TEACHERS

- teacher_id → full_name, subject

  → Fully dependent on PK

ATTENDANCE

- attendance_id → student_id, class_id, attendance_date, status

  → No non-key attribute depends on another non-key attribute.

III. Design Specification

3.1. Core DBMS Concepts Used (The Three):

For each of the required DBMS concepts, include a section detailing:

1. Primary Keys (PK) and Foreign Keys (FK)

Primary Keys are special fields in a table that give each record a unique identity.

Example:

- student_id in the Students table
- class_id in the Classes table

A primary key makes sure no two students or classes have the same ID, preventing confusion and duplicate entries.

Foreign Keys are fields that create a connection between tables.

They “link” tables together and make sure the data matches correctly.

Example in your system:

- student_id inside the Attendance table must match a real student in the Students table.
- class_id in the Attendance table must match a real class in the Classes table.

Why this matters:

- It prevents invalid data (like attendance for a non-existent student).
- It keeps your database organized, connected, and accurate.
- It satisfies Functional Requirement FR6 (referential integrity).

2. Trigger

A trigger is an automatic action performed by the database whenever a specific event happens, such as inserting, updating, or deleting a record.

In your project, the trigger activates every time a new attendance record is inserted.

What the trigger does:

When a student is marked Absent, the trigger instantly and automatically:

- Look for that student in the Students table.
- Increases their total_absent count by 1.

Why this is important:

- It removes the need for manual counting of absences.
- It reduces human error.

- It keeps the student's total absence record always correct.
- It supports FR4 (automatic absence updating).

This is more powerful than a CHECK constraint because:

- A CHECK rule can only validate data,
- But a trigger can perform calculations, updates, or logic, such as incrementing a counter.

3. View

A View is like a "virtual table" that shows data from multiple tables combined together.

It does not store new data instead, it displays information already inside the database in a clean and organized format.

In your Attendance System, views are used for reports.

Example:

A view can show:

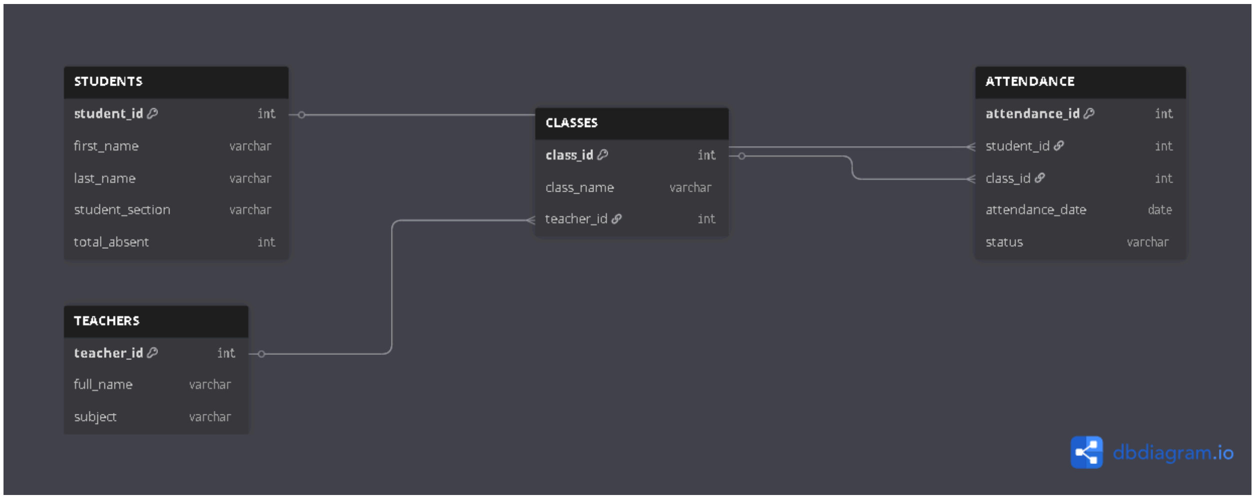
- Student name
- Class name
- Date
- Attendance status

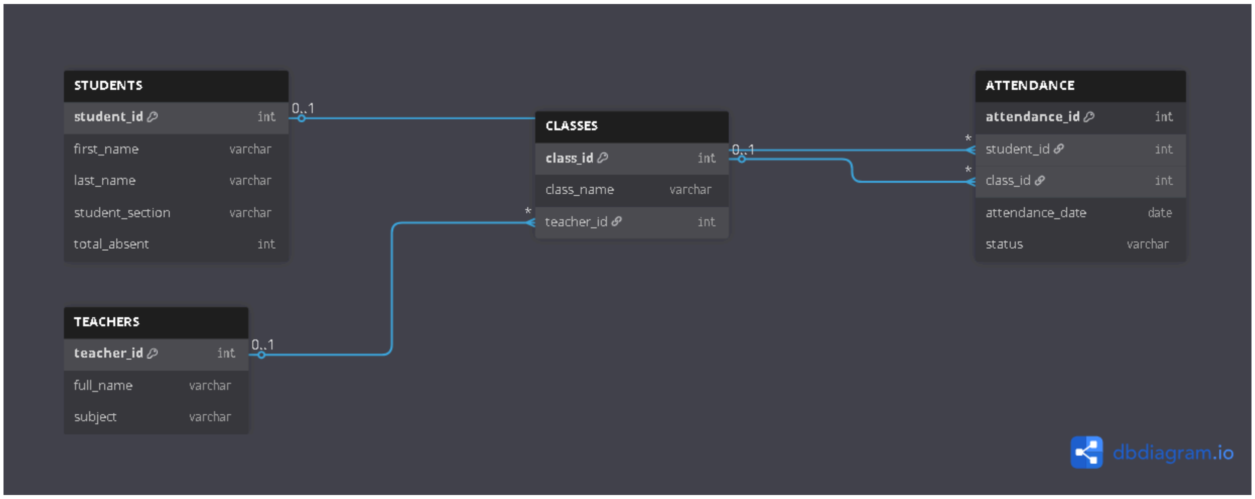
All in one screen, even though this data comes from different tables (Students, Classes, Attendance).

Why views are important:

- They make reporting faster and easier.
- Users don't need to write long JOIN queries.
- They help generate clean summaries for teachers or admins.
- They support FR5 (report generation using predefined views).

3.2. ER Diagram:





The **ER Diagram** of the *Student Attendance Monitoring System* shows how the main entities in the database are related to each other. The four main entities are **Students**, **Teachers**, **Classes**, and **Attendance**

- **Students store information such as student ID, name, section, and total absences. The student_id acts as the primary key, uniquely identifying each student.**

- **Teachers contain teacher details like full name and subject, with teacher_id as the primary key.**

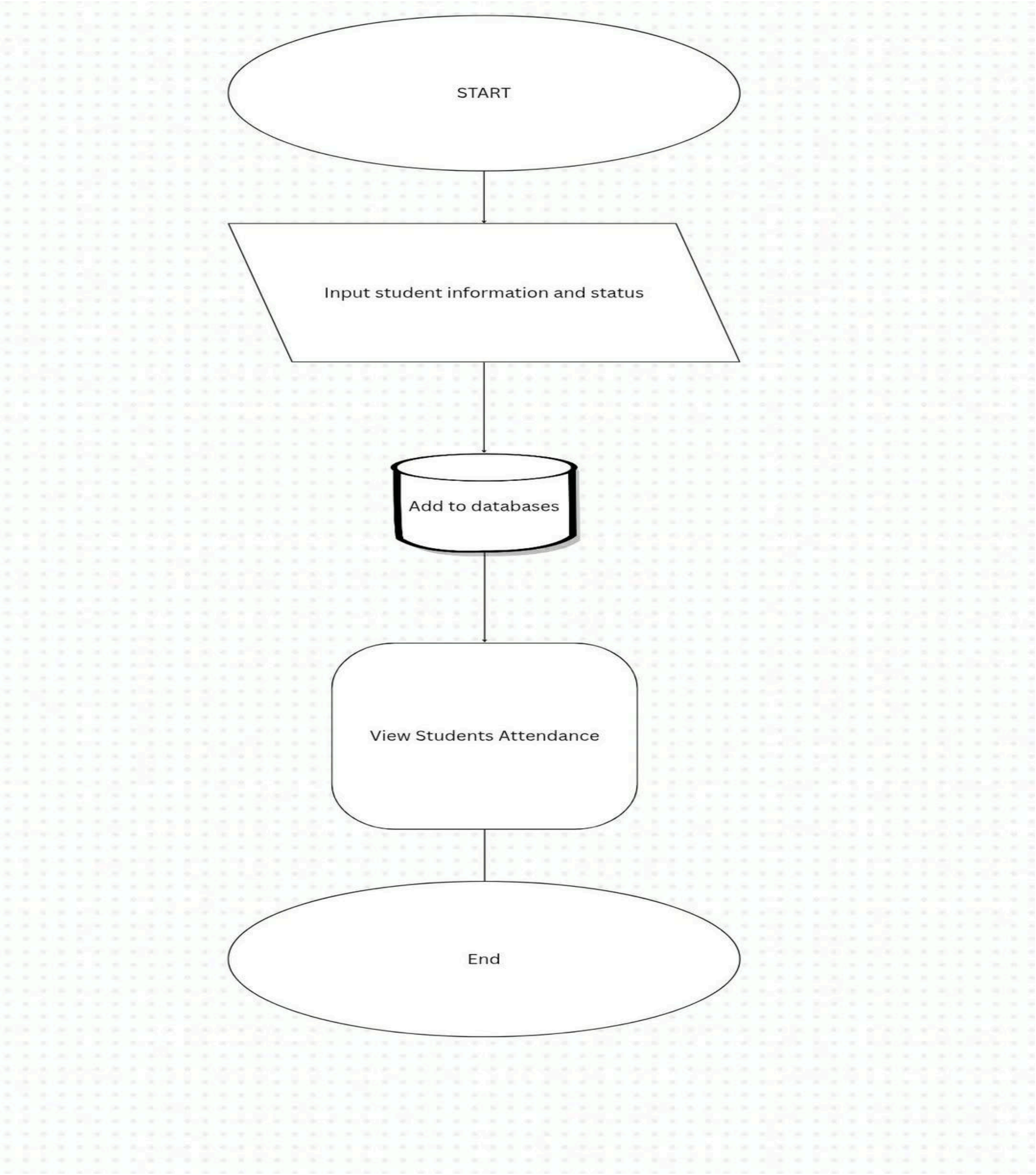
- **Classes represent class records and link teachers to classes. Each class has a class_id as its primary key and a teacher_id as a foreign key to show which teacher handles**

the class.

- **Attendance is the central table that connects students and classes. It records the attendance date and status (Present, Absent, Late, Excused). It uses student_id and class_id as foreign keys.**

These relationships ensure referential integrity, meaning attendance records cannot exist without valid students and classes. This design avoids data redundancy and supports accurate attendance tracking.

Flowchart


START
Input student information and status
Add to databases
View Students Attendance
End

The flowchart illustrates the basic process of recording and viewing student attendance.

The process starts when the system is opened.

First, the user inputs the student's information and attendance status, such as the student's name, ID, and whether they are present or absent. This step ensures that accurate attendance data is collected.

Next, the entered information is saved into the database. Storing the data allows the system to keep attendance records securely and makes them available for future reference.

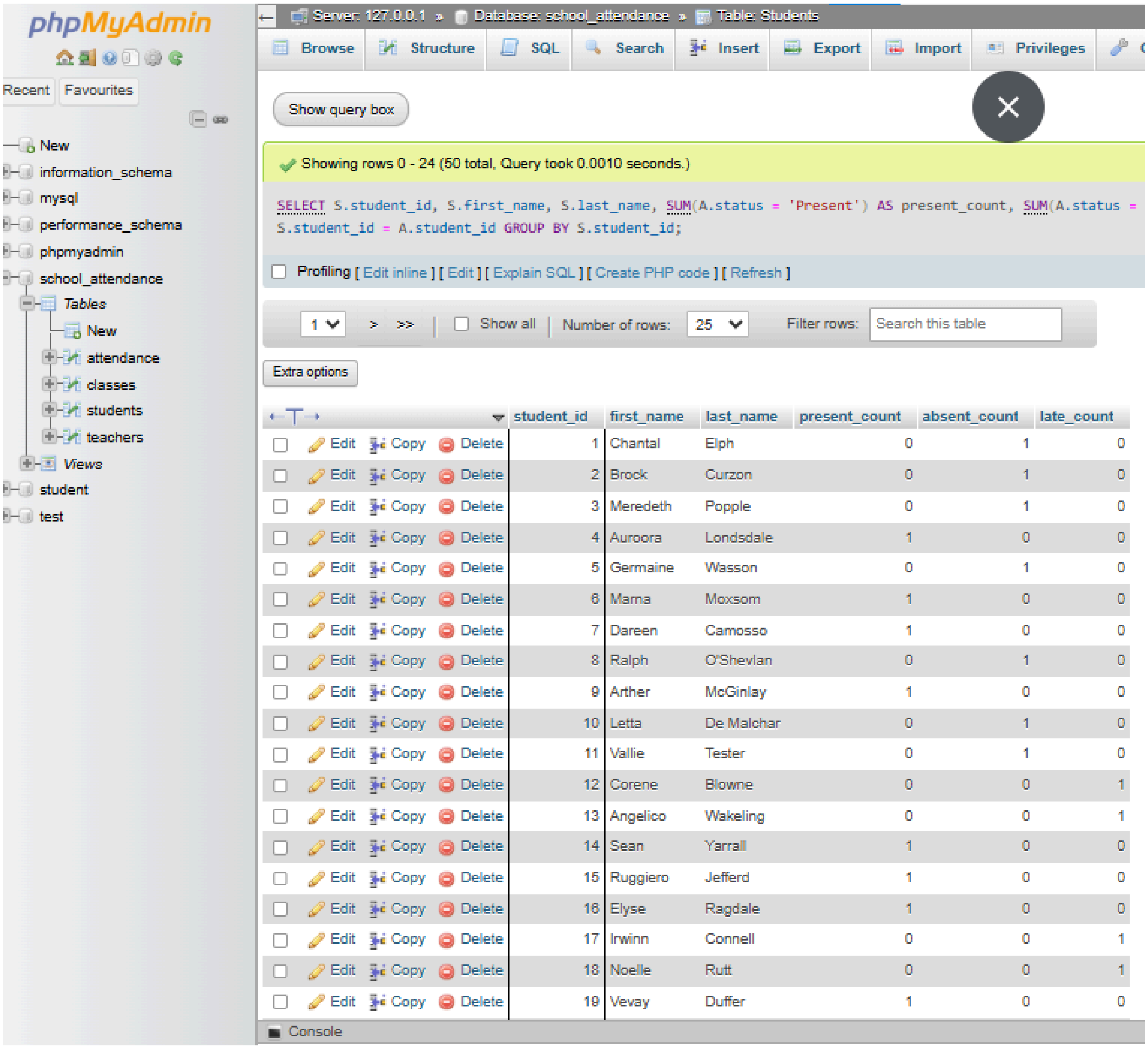
After the data is stored, the system allows the user to view the students' attendance. This step displays the recorded attendance information so teachers or administrators can monitor student presence and track attendance history.

Finally, the process ends once the attendance data has been successfully viewed.

Testing and Result

4.1. Test Cases :

Test 1:



✓ Showing rows 0 - 24 (50 total, Query took 0.0010 seconds.)

```
SELECT S.student_id, S.first_name, S.last_name, SUM(A.status = 'Present') AS present_count, SUM(A.status =
S.student_id = A.student_id GROUP BY S.student_id;
```

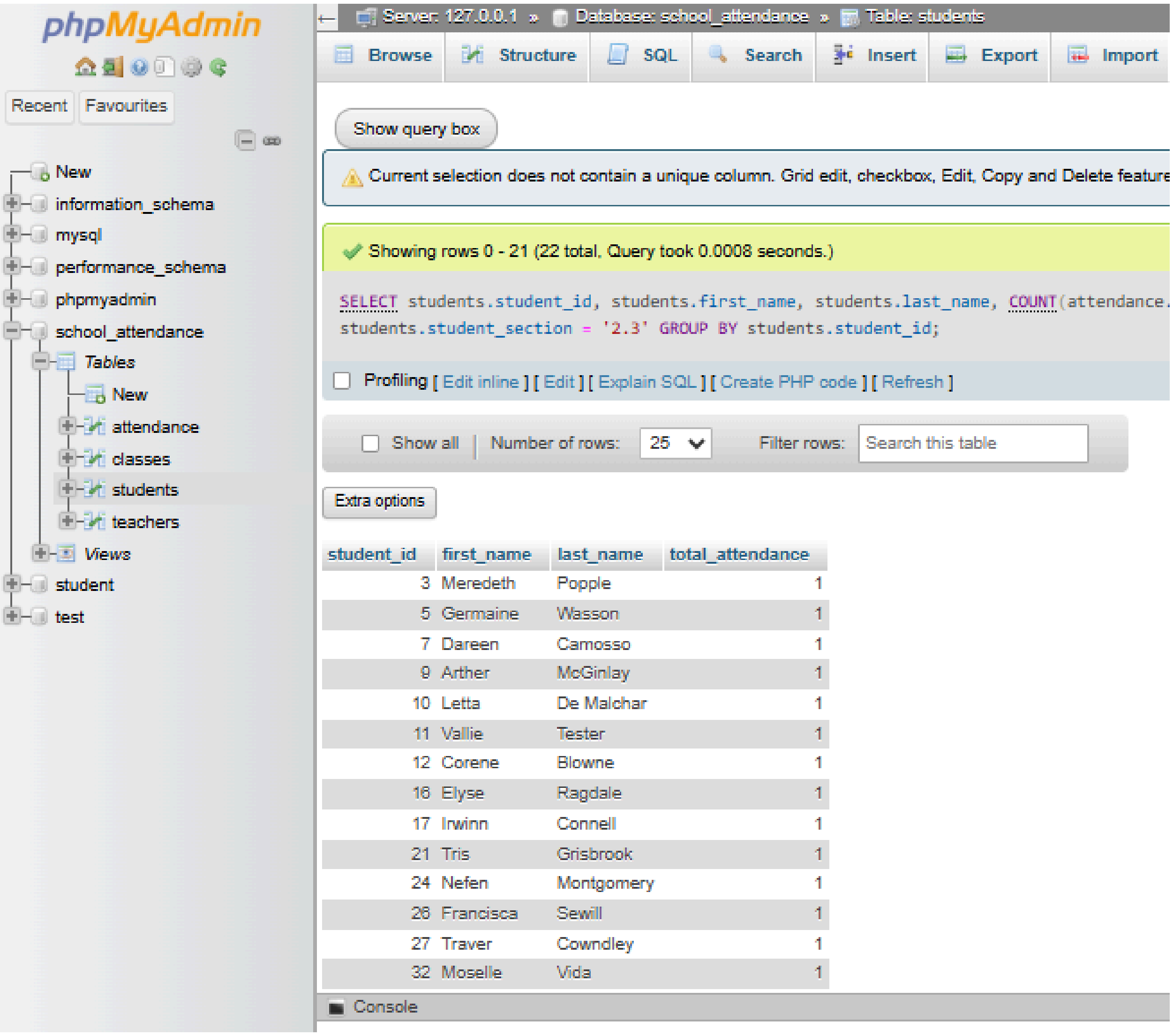
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> | □ Show all | Number of rows: 25 Filter rows: Search this table

	student_id	first_name	last_name	present_count	absent_count	late_count
□ Edit Copy Delete	1	Chantal	Elph	0	1	0
□ Edit Copy Delete	2	Brock	Curzon	0	1	0
□ Edit Copy Delete	3	Meredeth	Popple	0	1	0
□ Edit Copy Delete	4	Auroora	Londsdale	1	0	0
□ Edit Copy Delete	5	Germaine	Wasson	0	1	0
□ Edit Copy Delete	6	Marna	Moxsom	1	0	0
□ Edit Copy Delete	7	Dareen	Camosso	1	0	0
□ Edit Copy Delete	8	Ralph	O'Shevlan	0	1	0
□ Edit Copy Delete	9	Arther	McGinlay	1	0	0
□ Edit Copy Delete	10	Letta	De Malchar	0	1	0
□ Edit Copy Delete	11	Vallie	Tester	0	1	0
□ Edit Copy Delete	12	Corene	Blowne	0	0	1
□ Edit Copy Delete	13	Angelico	Wakeling	0	0	1
□ Edit Copy Delete	14	Sean	Yarrall	1	0	0
□ Edit Copy Delete	15	Ruggiero	Jefferd	1	0	0
□ Edit Copy Delete	16	Elyse	Ragdale	1	0	0
□ Edit Copy Delete	17	Irwinn	Connell	0	0	1
□ Edit Copy Delete	18	Noelle	Rutt	0	0	1
□ Edit Copy Delete	19	Vevay	Duffer	1	0	0

Test 1 verifies that attendance records can be inserted successfully with valid student and class IDs. This confirms that primary and foreign key constraints are working.

Test 2:



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete feature

✓ Showing rows 0 - 21 (22 total, Query took 0.0008 seconds.)

```
SELECT students.student_id, students.first_name, students.last_name, COUNT(attendance.
students.student_section = '2.3' GROUP BY students.student_id;
```

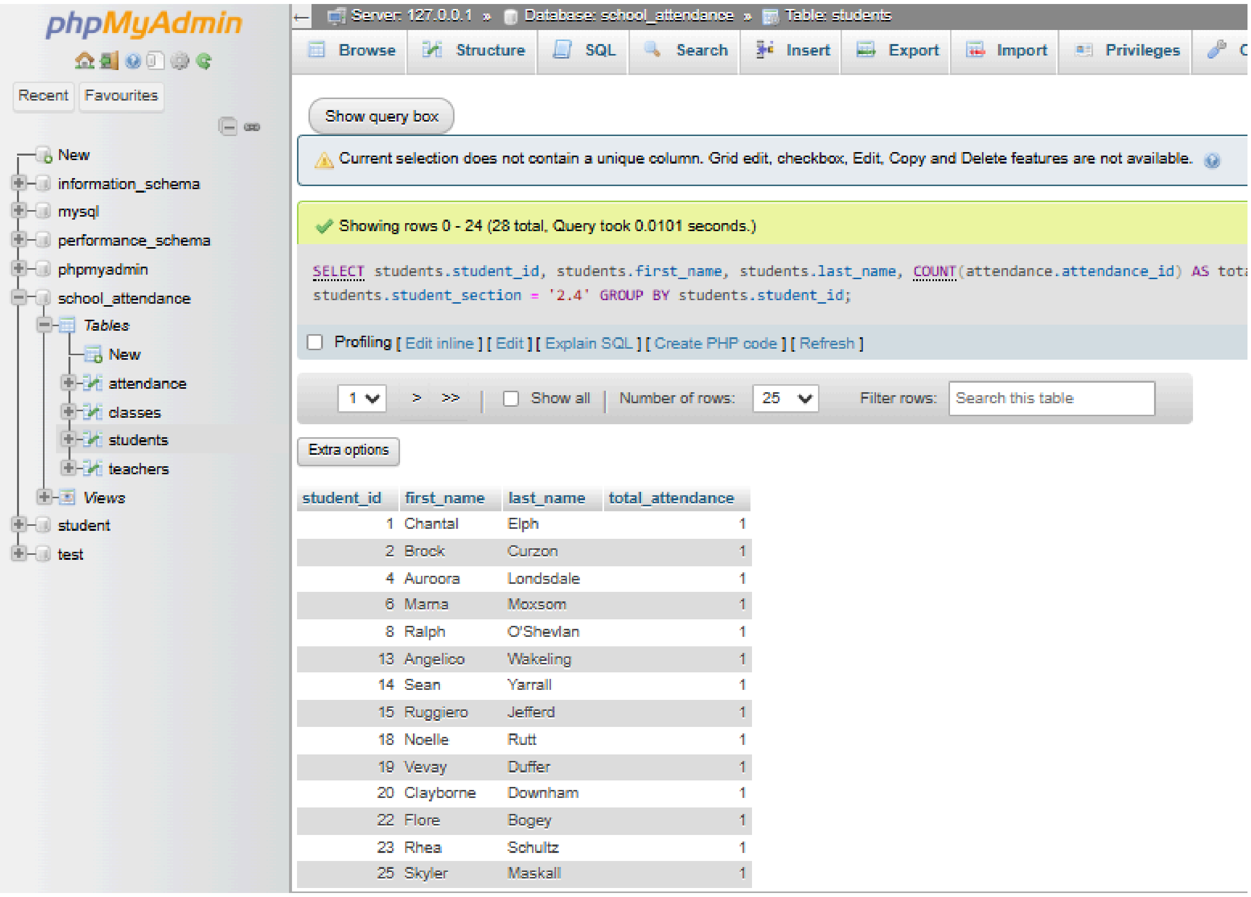
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table

student_id	first_name	last_name	total_attendance
3	Meredeth	Popple	1
5	Germaine	Wasson	1
7	Dareen	Camosso	1
9	Arther	McGinlay	1
10	Letta	De Malchar	1
11	Vallie	Tester	1
12	Corene	Blowne	1
16	Elyse	Ragdale	1
17	Irwinn	Connell	1
21	Tris	Grisbrook	1
24	Nefen	Montgomery	1
26	Francisca	Sewill	1
27	Traver	Cowndley	1
32	Moselle	Vida	1

Test 2 checks the **trigger functionality**. When a student is marked as *Absent*, the trigger automatically updates the **total_absent** field in the Students table. This proves that the system correctly automates absence counting and reduces human error.

Test 3:



Current selection does not contain a unique column. Grid edit, checkbox, Edit, Copy and Delete features are not available.

✓ Showing rows 0 - 24 (28 total, Query took 0.0101 seconds.)

```
SELECT students.student_id, students.first_name, students.last_name, COUNT(attendance.attendance_id) AS tota
students.student_section = '2.4' GROUP BY students.student_id;
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

1 > >> | □ Show all | Number of rows: 25 Filter rows: Search this table

student_id	first_name	last_name	total_attendance
1	Chantal	Elph	1
2	Brock	Curzon	1
4	Auroora	Londsdale	1
6	Marna	Moxsom	1
8	Ralph	O'Shevlan	1
13	Angelico	Wakeling	1
14	Sean	Yarrall	1
15	Ruggiero	Jefferd	1
18	Noelle	Rutt	1
19	Vevay	Duffer	1
20	Clayborne	Downham	1
22	Flore	Bogey	1
23	Rhea	Schultz	1
25	Skyler	Maskall	1

Test 3 validates the views used for reporting. The system successfully generates attendance reports that combine data from multiple tables, such as student name, class, date, and status.

Overall, the tests show that the system is accurate, reliable, and efficient, and that it successfully supports attendance **monitoring aligned with UN SDG 4 – Quality Education.**

Conclusion and Contributions

5.1. Conclusion

The Student Attendance Monitoring System successfully demonstrates how a structured relational database can improve the accuracy, reliability, and efficiency of attendance tracking. By applying normalization, triggers, views, and transactional procedures, the system ensures consistent data and supports the educational goal of promoting student participation and performance. The project aligns with UN SDG 4 (Quality Education) by enabling better monitoring and analysis of absenteeism trends.

5.2. Individual Contributions (Detailed breakdown of each member's assigned module/script).

Cledera, Anthony - Presentation, Documentation
Peñaver, Mark Joseph - Documentary, UI concept
Popes, Nathaniel- Database SQL Arrangement, Idea