requirements you’ve outlined:

* Teachers can only manage attendance for their own subjects and sessions.
* A central timetable (calendar) for the department or college dictates when classes (sessions) are scheduled.
* Teachers can add, edit, update, or delete attendance records for their students based on their assigned sessions.

I’ll expand on the earlier design to include teachers, subjects, a timetable, and appropriate restrictions. Below is the detailed schema with explanations.

**Key Requirements**

1. **Entities**:
   * Teachers (who take attendance).
   * Students (whose attendance is tracked).
   * Subjects (e.g., Math, Physics).
   * Timetable (central schedule for classes).
   * Sessions (specific instances of classes based on the timetable).
   * Attendance (records of student presence).
2. **Rules**:
   * Teachers are linked to specific subjects and sessions.
   * Only the assigned teacher can manage attendance for their sessions.
   * Sessions are scheduled based on a central timetable.

**Database Schema**

**1. Teachers Table**

Stores information about teachers.

* teacher\_id (unique ID)
* first\_name
* last\_name
* email (optional, for login or contact)

sql

CollapseWrapCopy

CREATE TABLE Teachers (

teacher\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

email VARCHAR(100) UNIQUE

);

**2. Students Table**

Stores information about students (replacing the generic "Users" table).

* student\_id (unique ID)
* first\_name
* last\_name
* roll\_number (unique student identifier)

sql

CollapseWrapCopy

CREATE TABLE Students (

student\_id INT PRIMARY KEY AUTO\_INCREMENT,

first\_name VARCHAR(50) NOT NULL,

last\_name VARCHAR(50) NOT NULL,

roll\_number VARCHAR(20) UNIQUE NOT NULL

);

**3. Subjects Table**

Stores the subjects taught in the college/department.

* subject\_id (unique ID)
* subject\_name (e.g., "Mathematics")
* teacher\_id (links to the teacher assigned to this subject)

sql

CollapseWrapCopy

CREATE TABLE Subjects (

subject\_id INT PRIMARY KEY AUTO\_INCREMENT,

subject\_name VARCHAR(100) NOT NULL,

teacher\_id INT,

FOREIGN KEY (teacher\_id) REFERENCES Teachers(teacher\_id) ON DELETE SET NULL

);

**4. Timetable Table**

Central timetable defining when classes occur (e.g., weekly or semester schedule).

* timetable\_id (unique ID)
* subject\_id (links to Subjects)
* day\_of\_week (e.g., Monday, Tuesday)
* start\_time
* end\_time
* room (optional, e.g., "Room 101")

sql

CollapseWrapCopy

CREATE TABLE Timetable (

timetable\_id INT PRIMARY KEY AUTO\_INCREMENT,

subject\_id INT,

day\_of\_week ENUM('Monday', 'Tuesday', 'Wednesday', 'Thursday', 'Friday', 'Saturday', 'Sunday') NOT NULL,

start\_time TIME NOT NULL,

end\_time TIME NOT NULL,

room VARCHAR(50),

FOREIGN KEY (subject\_id) REFERENCES Subjects(subject\_id) ON DELETE CASCADE

);

**5. Sessions Table**

Specific instances of classes based on the timetable.

* session\_id (unique ID)
* timetable\_id (links to Timetable)
* date (actual date of the session)
* status (e.g., Scheduled, Completed, Cancelled)

sql

CollapseWrapCopy

CREATE TABLE Sessions (

session\_id INT PRIMARY KEY AUTO\_INCREMENT,

timetable\_id INT,

date DATE NOT NULL,

status ENUM('Scheduled', 'Completed', 'Cancelled') DEFAULT 'Scheduled',

FOREIGN KEY (timetable\_id) REFERENCES Timetable(timetable\_id) ON DELETE CASCADE

);

**6. Attendance Table**

Records attendance for students in specific sessions.

* attendance\_id (unique ID)
* student\_id (links to Students)
* session\_id (links to Sessions)
* status (e.g., Present, Absent, Late)
* timestamp (when attendance was recorded)

sql

CollapseWrapCopy

CREATE TABLE Attendance (

attendance\_id INT PRIMARY KEY AUTO\_INCREMENT,

student\_id INT,

session\_id INT,

status ENUM('Present', 'Absent', 'Late') DEFAULT 'Absent',

timestamp DATETIME DEFAULT CURRENT\_TIMESTAMP,

FOREIGN KEY (student\_id) REFERENCES Students(student\_id) ON DELETE CASCADE,

FOREIGN KEY (session\_id) REFERENCES Sessions(session\_id) ON DELETE CASCADE

);

**7. Student\_Subject Table (Optional)**

If students are enrolled in specific subjects, this links students to subjects (many-to-many relationship).

* student\_id
* subject\_id

sql

CollapseWrapCopy

CREATE TABLE Student\_Subject (

student\_id INT,

subject\_id INT,

PRIMARY KEY (student\_id, subject\_id),

FOREIGN KEY (student\_id) REFERENCES Students(student\_id) ON DELETE CASCADE,

FOREIGN KEY (subject\_id) REFERENCES Subjects(subject\_id) ON DELETE CASCADE

);

**Relationships**

* **Teachers ↔ Subjects**: One teacher teaches one or more subjects (1-to-many).
* **Subjects ↔ Timetable**: Each subject has scheduled slots in the timetable (1-to-many).
* **Timetable ↔ Sessions**: Each timetable entry generates specific session instances (1-to-many).
* **Students ↔ Attendance**: One student has many attendance records (1-to-many).
* **Sessions ↔ Attendance**: One session has many attendance records (1-to-many).
* **Students ↔ Subjects** (via Student\_Subject): Students are enrolled in multiple subjects, and subjects have multiple students (many-to-many).

**How It Works**

1. **Timetable Setup**:
   * Define subjects (e.g., "Math" taught by Teacher X).
   * Schedule them in the Timetable (e.g., Math on Mondays, 9:00–10:00).
2. **Session Creation**:
   * Based on the timetable, sessions are created for specific dates (e.g., Math session on 2025-03-24).
   * Only the assigned teacher (via Subjects.teacher\_id) can manage these sessions.
3. **Attendance Management**:
   * Teacher X logs in, sees their sessions (filtered by their teacher\_id through Subjects and Timetable).
   * They add/edit/delete attendance records for students in their sessions.
   * Other teachers can’t access or modify these records.
4. **Restrictions**:
   * Enforce teacher-specific access at the application level (e.g., SQL query: WHERE Subjects.teacher\_id = <current\_teacher\_id>).

**Example Data**

* **Teachers**: Teacher X (teacher\_id: 1, "John", "Doe").
* **Subjects**: Math (subject\_id: 1, teacher\_id: 1).
* **Timetable**: Math on Mondays 9:00–10:00 (timetable\_id: 1, subject\_id: 1).
* **Sessions**: Math on 2025-03-24 (session\_id: 1, timetable\_id: 1).
* **Students**: Alice (student\_id: 1), Bob (student\_id: 2).
* **Attendance**: Alice is Present, Bob is Absent for session\_id: 1.