Here’s the **complete database schema** for your **School Management System** using **Laravel 10**, **Tailwind CSS**, **JS**, **jQuery**, and **AJAX**.

This schema now includes:

* Students
* Classes, Sections, Shifts
* Teachers and their subject assignments
* Attendance
* Results / Exams
* Student Accounting & Fees (Tuition, Late Fine, Exam Fee, etc.)
* Optional fines and miscellaneous payments

Everything is fully normalized and ready for Laravel migrations.

# **1. Core Academic Tables**

### ****1.1.**** classes

Stores all class levels (Play, Nursery, KG-01, etc.).

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID for class |
| name | VARCHAR(50) | Class name (e.g., Play, Class-05) |
| order\_number | INT | Order (1=Play, 2=Nursery, etc.) |
| description | TEXT (nullable) | Additional details |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****1.2.**** sections

Stores section data (A, B, C).

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| name | VARCHAR(10) | Section name |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****1.3.**** shifts

Stores shift info (Morning, Day).

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| name | VARCHAR(20) | Morning / Day |
| start\_time | TIME (nullable) | Shift start |
| end\_time | TIME (nullable) | Shift end |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****1.4.**** class\_sections

Defines which section and shift belongs to a class.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| class\_id (FK) | BIGINT | Reference → classes.id |
| section\_id (FK) | BIGINT | Reference → sections.id |
| shift\_id (FK) | BIGINT | Reference → shifts.id |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

# **2. Student Management**

### ****2.1.**** students

Stores student master data.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| student\_code | VARCHAR(30) | Unique Student ID (auto-generated) |
| first\_name | VARCHAR(50) | First name |
| last\_name | VARCHAR(50) | Last name |
| gender | ENUM('Male','Female','Other') | Gender |
| dob | DATE | Date of birth |
| guardian\_name | VARCHAR(100) | Guardian full name |
| guardian\_phone | VARCHAR(20) | Guardian phone number |
| address | TEXT | Address |
| admission\_date | DATE | Admission date |
| photo | VARCHAR(255) (nullable) | Photo path |
| status | ENUM('Active','Inactive','Transferred') | Student status |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****2.2.**** enrollments

Links students to a class, section, and shift for a specific academic year.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| student\_id (FK) | BIGINT | Reference → students.id |
| class\_section\_id (FK) | BIGINT | Reference → class\_sections.id |
| academic\_year | VARCHAR(10) | e.g., 2025-2026 |
| roll\_number | VARCHAR(20) | Roll number within class-section |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

# **3. Teacher & Subject Management**

### ****3.1.**** teachers

Stores teacher information.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique teacher ID |
| name | VARCHAR(100) | Full name |
| phone | VARCHAR(20) | Phone number |
| email | VARCHAR(100) | Email |
| specialization | VARCHAR(100) | Specialization subject |
| photo | VARCHAR(255) (nullable) | Photo path |
| status | ENUM('Active','Inactive') | Status |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****3.2.**** subjects

Stores all subjects taught at school.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique subject ID |
| name | VARCHAR(50) | Subject name (e.g., Math, English) |
| code | VARCHAR(20) | Subject code |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****3.3.**** class\_subjects

Assigns subjects to classes.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| class\_id (FK) | BIGINT | Reference → classes.id |
| subject\_id (FK) | BIGINT | Reference → subjects.id |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****3.4.**** teacher\_assignments

Assigns teachers to specific class-section-subject combinations.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| teacher\_id (FK) | BIGINT | Reference → teachers.id |
| class\_section\_id (FK) | BIGINT | Reference → class\_sections.id |
| subject\_id (FK) | BIGINT | Reference → subjects.id |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

# **4. Attendance Management**

### ****4.1.**** attendance

Tracks student attendance daily.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| student\_id (FK) | BIGINT | Reference → students.id |
| class\_section\_id (FK) | BIGINT | Reference → class\_sections.id |
| date | DATE | Attendance date |
| status | ENUM('Present','Absent','Late','Excused') | Attendance status |
| remarks | TEXT (nullable) | Additional notes |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

# **5. Exam & Result Management**

### ****5.1.**** exams

Stores exam events.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique exam ID |
| name | VARCHAR(50) | Exam name (e.g., Mid Term, Final) |
| start\_date | DATE | Start date |
| end\_date | DATE | End date |
| academic\_year | VARCHAR(10) | Academic year |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****5.2.**** exam\_results

Stores student results per subject.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| student\_id (FK) | BIGINT | Reference → students.id |
| exam\_id (FK) | BIGINT | Reference → exams.id |
| subject\_id (FK) | BIGINT | Reference → subjects.id |
| marks\_obtained | DECIMAL(5,2) | Marks obtained |
| grade | VARCHAR(5) | Grade (A, B, C, etc.) |
| remarks | TEXT (nullable) | Notes |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

# **6. Student Accounting & Fees**

### ****6.1.**** fee\_types

Defines different types of fees.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| name | VARCHAR(50) | Fee type (Tuition, Exam, Book, Fine, etc.) |
| description | TEXT (nullable) | Notes |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****6.2.**** fee\_structures

Assigns fee amounts to each class and fee type.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| class\_id (FK) | BIGINT | Reference → classes.id |
| fee\_type\_id (FK) | BIGINT | Reference → fee\_types.id |
| amount | DECIMAL(10,2) | Amount for that class & fee type |
| due\_day\_of\_month | INT (nullable) | Day of month tuition fee is due |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****6.3.**** student\_fees

Tracks fees charged to a student.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| student\_id (FK) | BIGINT | Reference → students.id |
| fee\_type\_id (FK) | BIGINT | Reference → fee\_types.id |
| amount | DECIMAL(10,2) | Amount charged |
| due\_date | DATE | Due date for payment |
| status | ENUM('Pending','Paid','Overdue') | Payment status |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

### ****6.4.**** payments

Logs actual payments made by students.

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| student\_fee\_id (FK) | BIGINT | Reference → student\_fees.id |
| amount\_paid | DECIMAL(10,2) | Paid amount |
| payment\_date | DATE | Payment date |
| payment\_method | ENUM('Cash','Card','Bank','Mobile Banking') | Payment method |
| transaction\_id | VARCHAR(50) (nullable) | For tracking |
| remarks | TEXT (nullable) | Notes |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

# **7. Optional: Fines & Penalties**

### ****7.1.**** student\_fines

For special fines (e.g., discipline issue, library late fine).

| **Column** | **Type** | **Description** |
| --- | --- | --- |
| id (PK) | BIGINT | Unique ID |
| student\_id (FK) | BIGINT | Reference → students.id |
| reason | VARCHAR(100) | Reason for fine |
| amount | DECIMAL(10,2) | Fine amount |
| imposed\_date | DATE | Date imposed |
| due\_date | DATE | Due date |
| status | ENUM('Unpaid','Paid') | Fine status |
| created\_at | TIMESTAMP | Auto timestamp |
| updated\_at | TIMESTAMP | Auto timestamp |

# **8. Summary of Tables**

| **#** | **Table Name** | **Purpose** |
| --- | --- | --- |
| 1 | classes | Store class list |
| 2 | sections | Store sections |
| 3 | shifts | Store shifts |
| 4 | class\_sections | Class + Section + Shift mapping |
| 5 | students | Student master data |
| 6 | enrollments | Student class enrollment |
| 7 | teachers | Teacher data |
| 8 | subjects | Subject list |
| 9 | class\_subjects | Subject per class |
| 10 | teacher\_assignments | Teacher assigned to class-section-subject |
| 11 | attendance | Student attendance |
| 12 | exams | Exam list |
| 13 | exam\_results | Student results |
| 14 | fee\_types | Define fee types |
| 15 | fee\_structures | Class-wise fee amounts |
| 16 | student\_fees | Fees charged to each student |
| 17 | payments | Actual payments by students |
| 18 | student\_fines | Special fines imposed |

# **Next Steps for Laravel**

* Create migrations using php artisan make:migration for each table.
* Use **Laravel Eloquent relationships**:
  + Student → Enrollment → ClassSection
  + Teacher → TeacherAssignments → Subject
  + StudentFee → Payment
* Integrate **AJAX and jQuery** for live updates (attendance, fee payment, etc.).
* Use **Tailwind CSS** for UI.

Would you like me to generate **Laravel migration code** for these tables step-by-step?