🎯 Project Title: *Smart Student Activity & Performance Tracker (SSAPT)*

🧩 Problem Statement:

In many schools, tracking student engagement in non-academic areas (clubs, discipline, performance records, etc.) is manual or fragmented. The goal of this project is to build a **Smart Web App** that allows:

- **Students** to view their own records and submit club/journal entries.
- **Teachers** to manage performance, discipline, and activity tracking.
- **Admins** to oversee everything, assign roles, and generate analytics.

🛠️ Core Features (Team Competition Tasks)

Module	Feature	Roles
Authentication	Login/Register with role-based access	Admin, Teacher, Student
Student Management	Add/edit/view students	Admin
Activity Tracking	Students join clubs, submit journal entries	Student
Performance Tracking	Teachers assign performance scores (per term)	Teacher
Discipline Tracking	Teachers add/remove incidents	Teacher
Feedback	Students view feedback; teachers respond to journal entries	Both
Notifications	Admin sends messages to teachers/students	Admin

Dashboard	Graphs of top performing students, most active clubs	Admin
Export/Report	PDF or Excel download of records	All roles (limited)
Audit Logs	Who did what, and when?	Admin only

💡 Bonus “Curious” Features (for advanced teams)

- **OTP Verification** on first login or account changes
- **Live Search** for students using AJAX or Livewire
- **Dark Mode / Light Mode**
- **AI Suggestion** – A PHP script that auto-suggests clubs based on interests
- **QR Code** – Print QR for each student to scan and view their profile

🧰 Suggested Technologies

- **PHP (Laravel preferred)** or Pure PHP with MVC structure
- **MySQL**
- **Bootstrap or TailwindCSS** for styling
- **Livewire** (optional for interactivity)
- **Blade** templating
- **PDF Generator** (like dompdf or barryvdh/laravel-dompdf)

🧱 Database Design (Simplified)

🔹 `users`

| id | name | email | password | role (admin/teacher/student) |

🔹 `students`

| id | user_id | registration_no | gender | birth_date |

🔹 `clubs`

| id | name | description |

🔹 `student_club_entries`

| id | student_id | club_id | joined_at |

🔹 `journals`

| id | student_id | content | created_at |

🔹 `performances`

| id | student_id | subject | score | term |

🔹 `discipline_records`

| id | student_id | incident | action_taken | reported_by (teacher_id) | date |

🔹 `notifications`

| id | title | message | target_role | created_by |

🔹 `audit_logs`

| id | action | actor_id | actor_role | target | created_at |

🏆 Evaluation Criteria

Criteria	Weight

Functionality & Completeness	30%
Code Structure (MVC, reusability)	20%
UI/UX & Styling	20%
Creativity & Bonus Features	15%
Presentation & Teamwork	15%

🧭 Team Roles (for group competition)

- **Frontend Leader:** Forms, dashboards, styling
- **Backend Leader:** Routes, validation, controllers
- **Database Engineer:** Relationships, migrations, seeders
- **Tester/Presenter:** Demos, testing, reporting bugs