

Student Achievement Tracking System

Database Schema

1. Users

- user_id (PK)
- name
- email (unique)
- password (hashed)
- role (Admin, Teacher, Student, Parent)
- created_at
- updated_at

2. Students

- student_id (PK)
- user_id (FK)
- date_of_birth
- gender
- address
- contact_number
- guardian_name
- guardian_contact
- created_at
- updated_at

3. Grades

- grade_id (PK)
- student_id (FK)
- subject
- grade
- created_at
- updated_at

4. Attendance

- attendance_id (PK)
- student_id (FK)
- date
- status (Present, Absent, Excused)
- created_at

5. Extracurricular Activities

- activity_id (PK)

- student_id (FK)
- activity_name
- participation_date
- created_at

6. Reports

- report_id (PK)
- student_id (FK)
- type (Progress, Achievement)
- generated_at
- report_url

7. Notifications

- notification_id (PK)
- user_id (FK)
- message
- status (read, unread)
- sent_at

Backend Architecture

- RESTful API developed using Python Flask / FastAPI.
- Database: SQLite / PostgreSQL for storing student data.
- Authentication: JWT-based authentication for secure login.
- ORM: SQLAlchemy for database interactions.
- Role-Based Access Control for managing permissions.
- Cloud Storage: Amazon S3 / Firebase Storage for report and media storage.
- Caching: Redis for improving performance.

API Endpoints

Authentication:

- POST /api/register: Register a new user.
- POST /api/login: Login user and return a token.
- POST /api/logout: Logout user.

Student Management:

- GET /api/students: Retrieve all students.
- POST /api/students: Add a new student.
- GET /api/students/{id}: Retrieve a specific student.
- PUT /api/students/{id}: Update student details.
- DELETE /api/students/{id}: Remove a student.

Performance Tracking:

- POST /api/students/{id}/grades: Add grade.
- GET /api/students/{id}/grades: Retrieve grades.
- GET /api/students/{id}/gpa: Calculate GPA.

Attendance:

- POST /api/students/{id}/attendance: Mark attendance.
- GET /api/students/{id}/attendance: Retrieve attendance records.

Reports:

- POST /api/students/{id}/generate-report: Generate report.
- GET /api/reports/{id}: Retrieve report.

Notifications:

- POST /api/notifications/send: Send notifications.
- GET /api/notifications/{user_id}: Retrieve user notifications.