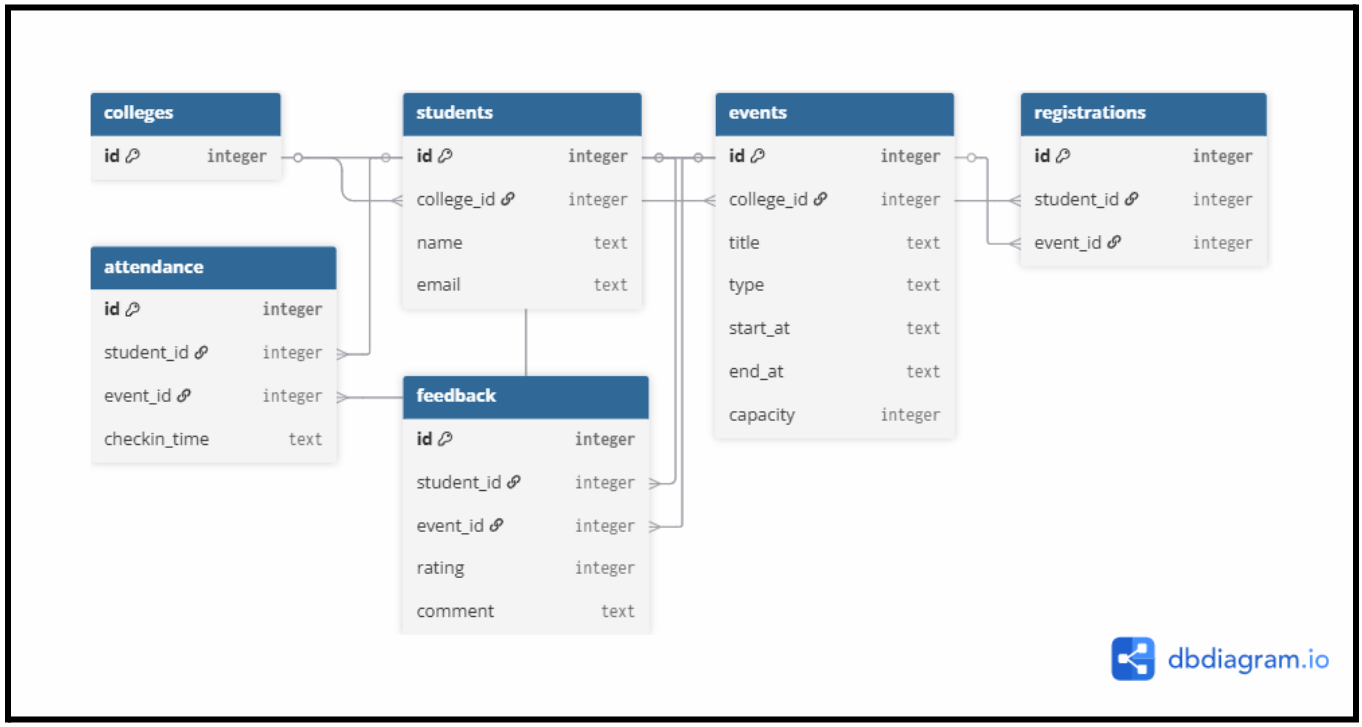
Design Document — Campus Event Reporting

Name: Akshata
Date: 07-09-2025
Project: Webknot Technologies — Campus Event Reporting (Prototype)

1. Data to Track:

Colleges → id.
Students → id, college_id, name, email.
Events → id, college_id, title, type, start_at, end_at, capacity.
Registrations → id, student_id, event_id.
Attendance → id, student_id, event_id, checkin_time.
Feedback → id, student_id, event_id, rating, comment.

2. Database Schema (ER Diagram):



3. API Design (Endpoints):

Student APIs

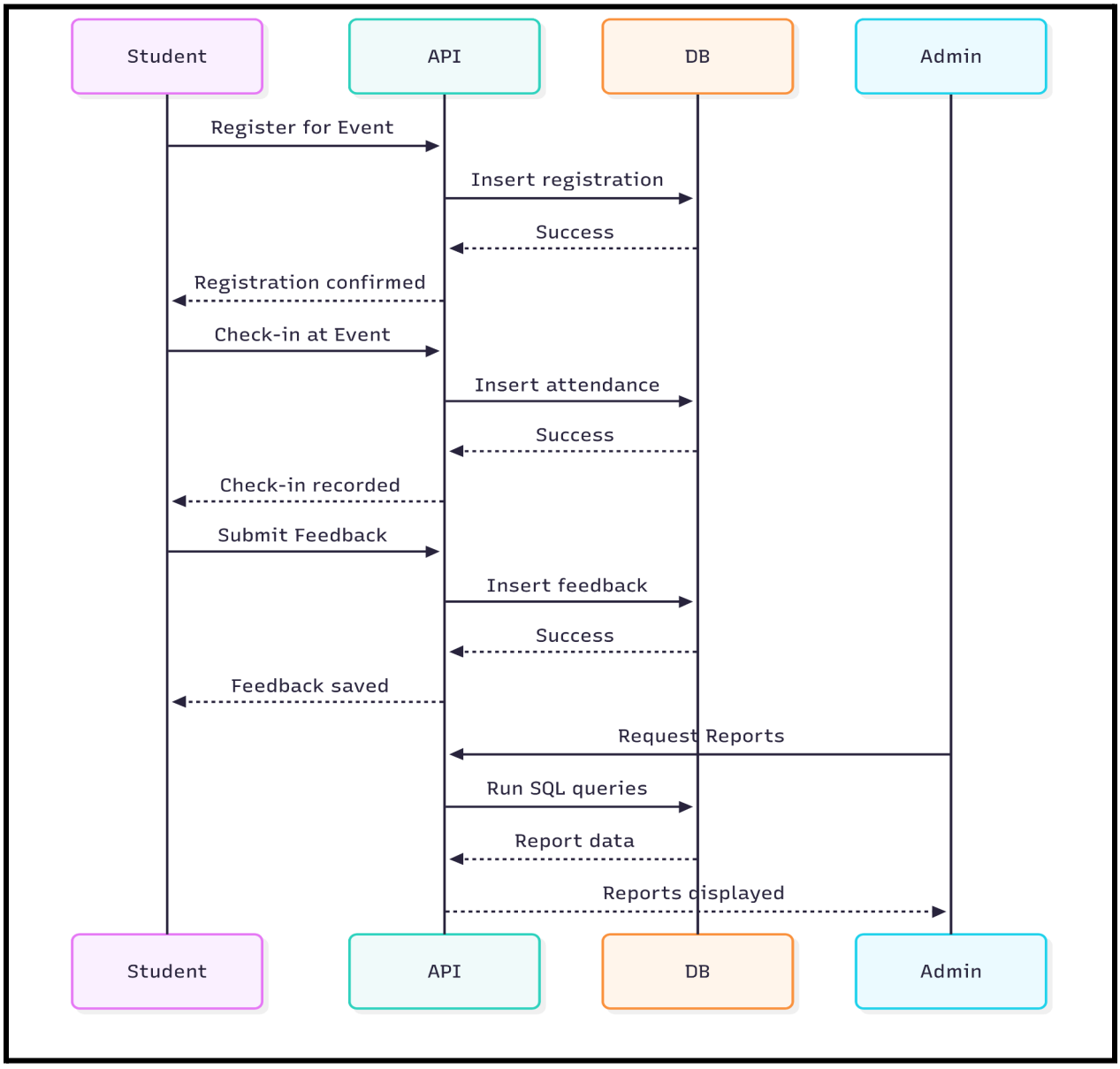
- POST /students → Create a student
- POST /events/{event_id}/register → Register student

- POST /events/{event_id}/attendance/checkin → Mark attendance
- POST /events/{event_id}/feedback → Submit feedback

Admin APIs

- POST /events → Create event
- GET /reports/event-popularity → Event popularity
- GET /reports/student-participation/{student_id} → Student participation
- GET /reports/top-students → Top students
- GET /reports/events?type=Workshop → Events by type

4. Workflow (Sequence Diagram):



5. Assumptions & Edge Cases:

- Each student/event has a unique ID (autoincrement).
- One registration per student per event.
- Attendance & feedback allowed only if registered.
- Duplicate registration, attendance and feedback blocked.
- Wrong IDs, API returns error.
- Event cancellation not implemented in prototype.

6. Security Notes:

- Input validation with Pydantic.
- SQL injection prevented via parameterized queries.
- Ratings restricted to 1–5.
- Prototype does not include login/authentication (future scope).