🔴 A. Limitations of EVENTFLOW (Realistic & Academic)

1️⃣ Cold Start Limitation (Recommendation Engine)

👉 কী সমস্যা?

Recommendation engine ইউজারের **past data** (কোন ইভেন্টে গেছে, কীতে interest দেখিয়েছে, কোন ডিপার্টমেন্ট, কোন কোর্স করেছে ইত্যাদি) ব্যবহার করে সাজেশন দেয়।
নতুন ইউজার যখন প্রথমবার সিস্টেমে আসে, তখন তার সম্পর্কে কোনো historical ডেটা থাকে না।

👉 কেন সমস্যা?

- কোনো ইভেন্টে অংশ নেয়নি → past participation নেই
- interest/hobby/major এখনো ঠিকমতো ভরেনি
- system আসলে বুঝতেই পারে না, এই ইউজারকে কী ধরনের ইভেন্ট সাজেস্ট করলে ভালো লাগবে

ফলে শুরুতে recommendations **কম relevant** হতে পারে।

👉 প্রভাব

- প্রথম impression খারাপ হলে ইউজার ভাবতে পারে "এই system আমার জন্য না"
- Engagement কমে যেতে পারে

✅ Solution (যেটা তোমরা বলবে)

আমরা তিন-স্তরের fallback strategy ব্যবহার করব:

1. **Trending events দেখানো**
    - সবার মধ্যে currently যেগুলো জনপ্রিয়, বেশি বুকিং হচ্ছে, সেই ইভেন্টগুলো দেখাবো।

- একদম new user-এর জন্য safe default।

2. **Department-based generic suggestions**
    - ডিপার্টমেন্ট যদি CSE হয়, ধরে নেয়া যায় সে programming/tech/swe সম্পর্কিত ইভেন্টে আগ্রহী হতে পারে।
    - তাই new user-এর জন্য department-based ইভেন্ট সাজেশন দেওয়া হবে।
3. **Onboarding-এ Interests নেওয়া**
    - প্রথম লগইনেই ছোট একটা form/step: "Which types of events are you interested in?"
    - সেখান থেকে initial preference নিয়ে recommendation চালু করা।

এভাবে আমরা cold start প্রভাব অনেকটাই কমিয়ে আনতে পারি।

2️⃣ Heavy Text-Based Computation (TF-IDF + Cosine Similarity)

👉 কী সমস্যা?

TF-IDF + cosine similarity মানে হলো অনেকগুলো **word vector** আর **high-dimensional গণনা**।
যখন:

- ইভেন্ট সংখ্যা বেশি
- ইউজার সংখ্যা বেশি

তখন:

- প্রত্যেক user-এর জন্য প্রত্যেকটা event-এর সাথে similarity হিসাব করতে হলে CPU তে extra load পড়ে
- বারবার online হিসাব করলে system slow হতে পারে

👉 প্রভাব

- Recommendation generate হতে বেশি সময় লাগতে পারে
- বেশি concurrent user থাকলে lag হতে পারে

✅ Solution

1. **Caching ব্যবহার**
   - same user জন্য বারবার same recommendation দরকার হলে, আগে হিসাব করা result সাময়িকভাবে cache-এ রেখে দেওয়া যায়
   - পরের বার direct cache থেকে serve করা যায়
2. **Preprocessing schedule (cron jobs)**
   - সব calculation real-time না করে, কিছু কাজ background এ করা যায়
   - যেমন: রাতে বা ১ ঘন্টা পরপর event vectors generate করা
3. **Vector storage reuse**
   - User profile vector আর event vector বারবার তৈরি না করে database/storage-এ রেখে দেওয়া
   - শুধু similarity part dynamic রাখা

এতে system লাইটওয়েট আর fast হয়।

3️⃣ No Deep Learning Based Recommendation (Yet)

👉 কী সমস্যা?

বর্তমান system কন্টেন্ট-ভিত্তিক (content-based):

- TF-IDF
- Cosine similarity
- Rule-based filtering

এগুলো ভালো, কিন্তু **advanced ML models** (যেমন collaborative filtering, embeddings, BERT ইত্যাদি) এখনো ব্যবহার করা হয়নি।

👉 প্রভাব

- Recommendation যথেষ্ট ভালো, কিন্তু "Netflix-level smart" না
- খুব complex behavior capture করতে পারছে না (যেমন, "যারা এটাতে যায়, তারা এইটাতেও যায়")

✅ Solution (এটা Future Scope হিসেবে বলবে)

- **Collaborative Filtering** – user-user similarity বা item-item similarity
- **Embeddings (BERT, word2vec ইত্যাদি)** – semantic similarity capture করার জন্য
- **Behavior tracking** – clicks, time spent, skips – এগুলো নিয়ে future model train করা

এইগুলো future enhancement হিসেবে উল্লেখ করলে বোর্ড খুশি হবে, কারণ they see you know the direction.

4️⃣ Internet Dependent System

👉 কী সমস্যা?

EventFlow **purely web-based**, database সবসময় online এ থাকে (Supabase)। user বা admin-এর device-এ internet না থাকলে:

- event list load হবে না
- QR scan result sync হতে দেরি হবে

👉 প্রভাব

- Offline ক্যাম্পাস event হলে user অসুবিধা মহসূস করতে পারে

- কিছুক্ষণের জন্য data sync delay হতে পারে

✅ **Solution (ভবিষ্যৎ scope)**

- **Offline caching system**:
  - last synced event list/QR validation rules local storage-এ রাখা
- **Sync when online**
  - internet ফিরে আসলে offline attendance data auto-sync

এগুলো এখনই না থাকলেও, future enhancement হিসেবে mention করবে।

5️⃣ **Payment Gateway Limitations**

👉 কী সমস্যা?

তোমরা local payment gateway (bKash, Nagad) ব্যবহার করবে।
কখনো কখনো:

- API down থাকে
- Transaction pending হয়ে যায়
- Callback না আসতে পারে

👉 প্রভাব

- User money কেটে গেছে কিন্তু system-এ ticket দেখাচ্ছে না – এই ধরনের situation খুব sensitive
- Refund/issue handle করতে না পারলে user trust হারায়

✅ **Solution**

1. **Retry Mechanism**
   - API call fail হলে নির্দিষ্ট সময় gap দিয়ে ১–২ বার পুনরায় চেষ্টা করা

2. **Transaction Logs**
    - প্রতিটি payment-এর status log রাখা
    - “Pending”, “Success”, “Failed” status clear রাখা
3. **Reconciliation System**
    - যদি কোনো সময় mismatch হয়, manual/admin reconciliation করার অপশন রাখা
    - user complaint দিলে transaction ID দিয়ে verification করা যাবে

6️⃣ QR Attendance Device Dependency

👉 কী সমস্যা?

QR scan করার জন্য:

- mobile অথবা scanner device দরকার
- slow device, ক্যামেরা খারাপ, network slow – এগুলোতে delay হতে পারে

👉 প্রভাব

- Event entry gate এ line বড় হয়ে যেতে পারে
- Multiple user একসাথে এলে congestion হতে পারে

✅ Solution

1. **Offline QR validation concept**
    - device লোকালেই কিছু validation করতে পারবে online না থেকেও
    - পরে পুরো attendance database-এ sync হবে
2. **Auto-sync when online**
    - নেট এলেই pending scan সব server এ পাঠিয়ে দেওয়া

3. **Multi-device scanning**
    - এক event-এ ১টা মাত্র device না রেখে একাধিক device দিয়ে doorway entry করা

🟠 **B. Major Challenges (Technical + Operational – Deep Explanation)**

1️⃣ **Real-Time Performance**

👉 **কী সমস্যা?**

যখন:

- সবাই একই সময়ে registration করছে
- popular event-এর ticket open হয়েছে
- অনেক concurrent request আসছে

তখন:

- server slow
- response time বেশি
- user frustration

✅ **Solution**

1. **NGINX Load Balancing**
    - একাধিক backend instance এর মধ্যে traffic ভাগ করে দেওয়া
2. **Optimized Queries**
    - DB query গুলো unnecessary join/scan বাদ দিয়ে lean করা
    - শুধু প্রয়োজনীয় data fetch করা

3. **Horizontal Scaling**
    - নতুন server instance add করার সুবিধা রাখা
    - ভবিষ্যতে cloud infra use করলে auto-scale সম্ভব

2️⃣ Data Consistency Across Supabase

👉 কী সমস্যা?

ধরো একটি ইভেন্টে ১০টা সিট আছে,
১৫ জন একসাথে ticket book করতে চাচ্ছে —

Concurrency এর কারণে:

- দুইজনই ভাবতে পারে সে last slot পেয়েছে
- Overbooking হয়ে যেতে পারে

Attendance side এও similar problem হতে পারে।

✅ Solution

1. **Row Level Security + Policies**
    - Supabase-এ rules লিখে দেওয়া—
        - এক user এক ticket
        - seat count limit cross করতে না দেওয়া
2. **Transaction Locks / Atomic Operation**
    - "decrease seat count + ticket create"—এগুলো একসাথে execute করা
    - মাঝখানে error হলে cancel করে roll back করা

3️⃣ Security Challenges (Authentication & Identity)

👉 কী সমস্যা?

Multiple login method:

- Email login
- Google login

এগুলো ঠিকমতো handle না করলে:

- same user-এর দুইটা আলাদা account তৈরি হয়ে যেতে পারে
- identity spoofing হতে পারে

✅ Solution

1. **Strong Identity Linking**
    - এক user-এর জন্য unique user_id রাখা
    - আলাদা login method-এর accountগুলো একই user_id-এর সাথে link করা
2. **Single Source of Truth**
    - ব্যবহারকারীর মূল identity metric (যেমন email বা student ID) consistent রাখা
3. **OAuth + Verification**
    - Google OAuth + verified email ব্যবহার করা

4️⃣ Admin Misuse or Incorrect Event Data

👉 কী সমস্যা?

যদি:

- Admin ভুল তারিখ/সময় দেয়
- Capacity ভুল সেট করে

- ভুল ticket rule apply করে

তাহলে event-এর পুরো management badly affect হবে।

✅ **Solution**

1. **Validation Rules**
    - Event form এ required field, date validation, capacity > 0 etc.
2. **Preview Before Publish**
    - Publish-এর আগে admin preview দেখবে "public view"
3. **Error Notification**
    - যদি কোনো inconsistency detect হয় (event time < current time, negative capacity), system warning পাঠাবে

5️⃣ **Certificate Authenticity**

👉 কী সমস্যা?

কেউ যদি externally fake certificate বানিয়ে বললো "এটা EventFlow থেকে এসেছে" — তখন verify করা difficult হতে পারে।

✅ **Solution**

1. **Unique Certificate ID**
    - প্রতিটি certificate-এর একটি unique ID থাকবে
2. **Backend Verification Page**
    - Public verification page থাকবে যেখানে সেই ID দিয়ে ছাত্রের নাম ও ইভেন্টের নাম verify করা যাবে
3. **QR on Certificate**

- Certificate-এ থাকা QR code scan করলে original verification page এ নিয়ে যাবে

6️⃣ User Adoption Challenge

👉 কী সমস্যা?

সিস্টেম যত ভালোই হোক, যদি ছাত্ররা ব্যবহার না করে, তাহলে কোনো লাভ নেই।

- সবাই হয়তো একসাথে migrate করতে চাইবে না
- কেউ কেউ শুধু Facebook ইভেন্টেই comfortable

✅ Solution

1. **Gamification**
   - event অংশগ্রহণ করলে points বা badges
   - top attendees leaderboard
2. **Strong UI/UX**
   - সহজ, clean interface
   - fewer clicks, clear navigation
3. **Push Notifications & Value**
   - ব্যবহারকারী দেখবে:
     - সব event এক জায়গায়
     - certificate, history, recommendation—সব benefits
   - এতে naturally usage বাড়বে

🔵 **C. Presentation Board**

📌 Question 1: এই প্রজেক্ট বানানোর মূল উদ্দেশ্য কী?

Perfect Answer:

বিশ্ববিদ্যালয়ের ম্যানুয়াল ইভেন্ট ম্যানেজমেন্ট—যেমন কাগজে রেজিস্ট্রেশন, অফলাইন টিকিটিং, ম্যানুয়াল অ্যাটেনডেন্স—এসবকে একটি আধুনিক, স্বয়ংক্রিয় ও data-driven সিস্টেম দিয়ে replace করা।
EVENTFLOW পুরো ইভেন্ট lifecycle-কে digitize করে:
Creation → Ticket → Payment → Attendance → Certificate → Recommendation।

📌 Question 2: Recommendation Engine কীভাবে কাজ করে?

Perfect Answer:

System user-এর **department, interests, hobbies, major, completed courses** থেকে একটি user profile vector বানায়।
Event description থেকে event vector বানায় (TF-IDF).
হিসাব করা হয় cosine similarity = কতটা মিল আছে user ও event-এর মধ্যে।
Top highest similarity score-গুলো user-কে সাজেস্ট করা হয়।

📌 Question 3: Cold Start Problem কী? আপনার system কীভাবে সমাধান করে?

Perfect Answer:

যখন নতুন user-এর কোনো data নেই তখন system recommendation দিতে পারে না।
আমরা সমাধান করেছি—

- Default trending events
- Department-based generic events

- User onboarding interests form
- University-wide popular events

📌 Question 4: কেন Supabase বেছে নিয়েছেন Firebase/MySQL নয়?

Perfect Answer:

Supabase PostgreSQL-based হওয়ায় relational data, ACID compliance এবং RLS security খুব strong।
Event system-এ users, events, tickets, attendance—সব relational nature-এর, তাই Supabase সবচেয়ে perfect.

📌 Question 5: QR Attendance কীভাবে নিশ্চিত করা হয় যে fraud হবে না?

Perfect Answer:

- Unique event-based QR code
- User-token + event-token validation
- Real-time check-in
- Duplicate scan blocking
- Admin log tracking

📌 Question 6: সবচেয়ে বড় technical challenge কী ছিল?

Perfect Answer:

Real-time ticket availability management এবং attendance syncing।
আমরা atomic operations ও Supabase RLS ব্যবহার করেছি যাতে data কখনো mismatch না হয়।

📌 Question 7: System-এর scalability কিভাবে অর্জন করেছেন?

Perfect Answer:

NGINX reverse proxy + stateless backend architecture ব্যবহার করেছি।
Need হলে horizontally multiple backend instance চালানো যাবে।
Supabase auto-scaling DB handle করে।

📌 Question 8: Future improvement কী কী?

Perfect Answer:

- ML-based collaborative filtering
- Offline attendance scanning
- Mobile app version
- Multi-language support
- AI-powered event summary

🟢 D. Board যদি জিজ্ঞেস করে—"This project is already available like Eventbrite, then why yours?"

Perfect Answer:

Eventbrite general-purpose; এটি academic institution-এর জন্য optimized নয়।
আমাদের system:
✓ University role hierarchy (Super Admin → Admin → User)
✓ Student-focused recommendation engine
✓ Certificate automation
✓ QR attendance built-in
✓ Bangladesh payment gateway (bKash/Nagad)

✓ Academic structuring (departments, clubs, courses)
These features **কোনো existing tool দেয় না**।

E. Bonus: ৫টি Hard Questions + Smart Answers

Q: আপনার system down হলে কি হবে?

A: Graceful error handling, retry queue, backup server support, এবং Supabase-এর redundancy রয়েছে।

Q: আপনার algorithm bias-free কি?

A: আমরা user interest ও department weight-balanced করি যাতে কোনো category dominate না করে।

Q: Security breach হলে কী করবেন?

A: JWT invalidation, role-locking, activity log audit, এবং forced password reset।

Q: Data privacy কীভাবে ensure করছেন?

A: Row Level Security, hashed credentials, OAuth-based access, এবং limited privilege-based DB access।

Q: আপনার system offline হলে attendance কিভাবে হবে?

A: ভবিষ্যৎ scope: offline QR caching + auto sync when online।

🎯 **Question–Answer Set (Very Important for Viva / Presentation)**

❓ Question 1: "এই সিস্টেমটা ব্যবহার করবে কারা? এবং কেন ব্যবহার করবে?"

✅ **Perfect Answer:**

EVENTFLOW মূলত তিন ধরনের ব্যবহারকারীর জন্য তৈরি—

1. Super Admin (University Authority)

তারা এই সিস্টেম ব্যবহার করবে কারণ—

- সব ক্লাব ও ইভেন্ট এক জায়গা থেকে মনিটর করতে পারবে
- Transparency বাড়ে
- Attendance, ticketing, certificate—সব ডেটা একসাথে পাওয়া যায়
- Decision-making সহজ হয় (which event is popular, student engagement etc.)

2. Club Admin / Event Organizers

তাদের ব্যবহারের কারণ—

- ইভেন্ট তৈরি, customization, টিকিট, notification—all in one place
- QR attendance—manual sheet-এর ঝামেলা নেই
- Certificate auto-generate হয়
- Analytics দেখে future event improve করা যায়

3. Students

তারা ব্যবহার করবে কারণ—

- Personalized event suggestions
- Campus-এর সব event এক জায়গায়

- Online ticketing
- QR চেক-ইন (সারি নেই, ঝামেলা নেই)
- Participation certificate portfolio-ready

👉 **Short summary:**
"Each group gets real value—admin gets control, organizers get automation, students get a smart campus experience."

❓ Question 2: "Users তোমাদের system সম্পর্কে জানবে কীভাবে?"

✅ **Perfect Answer:**

আমরা multi-level awareness strategy ব্যবহার করবো—

1. University-এর Official Communication

Registrar/IT Office থেকে:

- Email announcement
- Student portal notification
- University Facebook page

2. Orientation Program

1st-year students-দের orientation-এ system demos দেওয়া হবে।

3. Club Promotion

প্রতিটি club ইভেন্ট publish করার সময় বলবে—

"EventFlow থেকে ticket collect করুন।"

4. On-ground QR Posters

Campus-এর গুরুত্বপূর্ণ স্থানে QR code poster—scan করলেই EventFlow signup page।

5. Incentive Campaign

- প্রথম ৫টি ইভেন্টে অংশগ্রহণ করলে badge
- Leaderboard system
- Certificate tracking

👉 Board বুঝবে তুমি Marketing + Adoption strategy চিন্তা করেছো।

❓ Question 3: "System কি শুধু তোমাদের University-এর জন্য, নাকি Bangladesh-এর সব University-এর জন্য?"

✅ **Perfect Answer:**

আমাদের ভিত্তি নকশা (architecture) multi-university ready।

সুতরাং System:

- এখন **আমাদের University** দিয়ে শুরু
- ভবিষ্যতে **পুরো Bangladesh-এর সব University** adopt করতে পারে

Why scalable for all universities?

- Role system flexible
- Multi-tenant architecture possible
- Each university আলাদা data silo-তে manage করা যাবে

❓ Question 4: "যদি Bangladesh-এর সব University-এর জন্য বানানো হয়, তাহলে তারা system সম্পর্কে জানবে কীভাবে?"

✅ **Perfect Answer:**

We will use **institution-level onboarding strategy**:

1. University Partnerships

We will approach:

- IT Department
- Clubs + Event Coordination Body
- Administration
  একটি pilot program offer করব—

"Free 3-month trial for all events."

2. National Inter-University Hackathons / Tech Events

EventFlow sponsor হিসাবে:

- Students নিজেরাই system experience করবে
- Word-of-mouth awareness ছড়িয়ে যাবে

3. Social Media Marketing

- LinkedIn campaigns for authority
- Facebook/TikTok promotions for students

4. Success Case Studies

- আমাদের নিজের university-এর real success metrics দেখানো
- Attendance improvement
- Admin workload reduction

❓ Question 5: "Authority ছাড়া অন্য University-কে approach করবে কীভাবে?"

✅ **Perfect Answer:**

We will follow a structured B2B (Business to Institution) outreach:

1. Cold Email Outreach to University Admin

Subject: *Improve Your Event Management Using EventFlow*

2. Demo Presentation Meeting

একটি live demo দিয়ে দেখানো—

- কীভাবে automate হবে attendance
- কীভাবে recommendation increase করবে participation

3. Offer Free Pilot for Their Students

- ফ্রি ১ মাস
- তারা চাইলে dataset migrate করা যাবে

4. Students-এর সাথে Direct Engagement

- Inter-university competitions
- Club collaboration
- Ambassadors program

👉 বলবে:
"We onboard authority first, because only they can publish events."

❓ Question 6: "যখন দিনে ৫০–১০০টা event তৈরি হবে, তখন কি user-এর কাছে ১০০টা notification যাবে?"

🤯 **Dangerous Question — Many Students Fail!**

✅ **Perfect, Smart Answer:**

না, EVENTFLOW-এ user notification system **intelligent & controlled**।

User কখনোই ১০০টা notification পাবে না।

Why?

Notification types are categorized:

---

1. Priority Notification (Only for Registered Events)

যে ইভেন্টে user ticket করেছে—

- Reminder
- Venue change
- Time change

👉 এটাই mandatory notification।

---

2. Personalized Event Alerts

User-এর interest/department/major অনুযায়ী very few suggestions পাঠানো হবে।

👉 Maximum 1–2 per day.

---

3. Bulk Event Notifications → Not sent

যে events user-এর সাথে সম্পর্কিত নয়—
zero notifications.

---

4. Notification Digest Mode

User চাইলে daily summary পাবে:

"Today's top 5 recommended events for you."

👉 No spam
👉 No frustration

Short Perfect Answer:

“Our system uses smart notification grouping, filtering, and digest mode. Users only receive relevant and personalized notifications—not 100 random alerts.”

🔥 **BONUS Question Board জিজ্ঞেস করতে পারে:**

❓ **Q: “এই system সফল হবে কেন?”**

⭐ **Model Answer:**

Because EVENTFLOW directly solves the 6 biggest campus problems:

- Manual registration
- Paper attendance
- Scattered event information
- No centralized system
- No certificate automation
- No personalized event discovery

It saves time, reduces workload, increases participation, and creates a smart university environment.

❓ **Q: “System fail হলে fallback কী?”**

⭐ **Answer:**

- Offline attendance backup
- Cloud DB snapshots
- Manual export option
- Local cache

❓ **Q: "Students motivation কী system use করতে?"**

⭐ **Answer:**

- Easy ticketing
- Personalized discovery
- Certificates
- Event history
- One-stop solution

✅ Q1. তোমাদের সিস্টেম multi-university হলে data isolation কীভাবে নিশ্চিত করবে?

⭐ **Model Answer:**

আমরা **tenant-based architecture** ব্যবহার করবো, যেখানে প্রতিটি university আলাদা namespace/DB schema পাবে।
এর ফলে—

- এক বিশ্ববিদ্যালয়ের data অন্য university দেখতে পারবে না
- Role-based access + row-level security (Supabase RLS) দিয়ে sensitive data isolate থাকবে
- Multi-tenant model future scaling সহজ করে

"We treat each university as an isolated data tenant."

✅ **Q2. If your system scales to 10 lakh students, how will your backend handle recommendation load?**

⭐ **Model Answer:**

- Vector caching (store precomputed TF-IDF vectors)
- Incremental update (only update vectors when user interest changes)
- Offline batch processing using cron jobs
- Redis for low-latency retrieval
- Horizontal scaling (multiple backend nodes with NGINX load balancer)

"We remove real-time heavy computation and transform it into periodic, optimized background tasks."

✅ **Q3. তোমাদের System-এর সবচেয়ে weak point কোথায়? Accept it honestly.**

⭐ **Model Answer:**

Current version-এর সবচেয়ে দুর্বল দিক:

- Recommendation engine fully ML-based নয় (rule-based + TF-IDF-driven)
- Offline attendance বর্তমানে built-in নেই
- Student engagement long-term sustain করতে gamification প্রয়োজন

But আমরা এগুলো future roadmap-এ রেখেছি।

✅ **Q4. QR attendance জালিয়াতি কীভাবে ১০০% প্রতিরোধ করবে?**

⭐ **Model Answer:**

আমরা তিন-স্তরের validation ব্যবহার করি:

1. **Encrypted QR Token**
   QR-এর ভিতরে event_id + user_id hashed থাকে।
   Fake QR বানানো অসম্ভব।
2. **Single-scan rule**
   এক user একবার check-in করলে second scan reject হবে।
3. **Server cross-verification**
   Token decode করে backend ensures—

- event-এর সাথে user-এর ticket আছে কিনা
- user attendance already marked কিনা

Thus cheating nearly impossible.

✅ Q5. Ticket overbooking prevent কীভাবে করবে high concurrency situation-এ?

⭐ **Model Answer:**

Supabase-এর atomic transaction + row locking ব্যবহার করে:

- event capacity decrement + ticket creation একই operation-এ execute হবে
- যদি mid-way error আসে, পুরো operation rollback হবে

Thus:

"We guarantee consistency even when 100 users try simultaneously."

✅ Q6. যদি Admin ভুল করে event delete করে ফেলে? Then what?

⭐ **Model Answer:**

We have multiple layers of protection:

1. **Soft Delete System**
   Event immediately remove না হয়ে "Archived" এ যায়।
2. **Admin Recovery Console**
   Accidental deletion restore করা যায় ৭ দিনের মধ্যে।
3. **Audit Logs**
   কোথা থেকে কে delete করেছে track করা যায়।

Thus data loss prevent হয়।

✅ Q7. তোমাদের System GDPR/Privacy-Compliant কি? If yes, how?

⭐ **Model Answer:**

Yes, আমরা privacy principles follow করি:

- Minimal data storage
- Clear user consent
- Ability to delete account
- Sensitive information encrypted
- RLS ensure data isolation
- Audit logs for transparency

Even though Bangladesh GDPR enforce করে না, আমরা best practice follow করি for global standardization.

✅ Q8. তোমাদের Notification System কি user frustration তৈরি করতে পারে? How will you prevent spam?

⭐ **Model Answer:**

We implemented **Noise Filtering Logic**:

1. High-priority notifications → শুধুমাত্র registered events-এর জন্য
2. Personalized recommendations → max 2/day
3. Digest notification → Daily summary
4. Muted mode → User চাইলে notifications বন্ধ রাখতে পারবে

"We prefer relevance over volume. Our algorithm pushes only meaningful alerts."

✅ Q9. যদি Supabase server down হয়, তোমরা কী contingency plan রেখেছো?

⭐ **Model Answer:**

We use a three-stage fallback:

1. **Local cache backup** (recent event list, user session)
2. **Fail-safe offline attendance queue** (scan stored offline)
3. **Sync-on-reconnect**
   Server আসলেই pending operations execute করে।

This ensures continuous functionality even under outages.

✅ Q10. Bangladesh-এর সব বিশ্ববিদ্যালয়ে প্রয়োগ করতে গেলে legal approvals লাগবে কি?

⭐ **Model Answer:**

Yes, administrative approvals প্রয়োজন।
We will follow:

- IT cell clearance
- Data privacy documentation
- Student activity council approval

- MoU signing with each university

By presenting clear benefits + security documents, authorization easier হবে।

🔥 **EXTRA HARD QUESTIONS**

(এগুলো viva-তে বললে বোর্ড আপনার depth দেখে respect করবে)

❓ **Q11. What happens if a user tries to manipulate certificate IDs?**

⭐ **Model Answer:**

Certificate IDs hash-based এবং tamperproof।
Any mismatch server instantly invalidates the certificate.

❓ **Q12. তোমাদের system ক্লাব conflict কীভাবে handle করবে?**

⭐ **Model Answer:**

Same time-slot conflict হলে warning দেবে—
"Another event is already scheduled at this time."
Admins alternative slot receive করবে।

❓ **Q13. যদি এক user mistake করে ভুল department set করে, সাজেশন ভুল আসবে—How to fix?**

⭐ **Model Answer:**

User profile editable রাখা হয়েছে।
User anytime department update করলে new vector generate হবে।

❓ Q14. তোমাদের system কীভাবে student interest predict করবে?

⭐ **Model Answer:**

Not only user-input.
System watches:

- Clicks
- Bookmarks
- Past attendance
- Event category visit time

Gradually interest vector evolve করে।

❓ Q15. তোমরা কি ভাবো তোমাদের system eventually একটি full "Campus Ecosystem Platform" হতে পারে?

⭐ **Model Answer:**

Yes.
Future roadmap includes:

- Hostel management
- Club management automation
- Smart timetable
- Chat & collaboration
- Internship recommendation

অর্থাৎ EVENTFLOW একটি **complete student life management ecosystem** হতে পারে।