Staff Hackathon Feb 2026

Smart Admission & Discount Engine System

👥 Hackathon Note:

- Each participant will be working individually.
- Choose the technology/language/database as per your comfort/requirement.
- You can use any IDE, Documentation, online Tutorials, etc.
- Use of AI tools (ChatBots, Copilots, Vibe-coding, etc.) is **allowed**.

📌 Problem Statement

Design and implement a **Admission Management System** for an institute offering multiple courses and batches.

The system must allow students to register for available batches and apply eligible discounts through a **flexible, admin-configurable discount engine**.

🎯 Core Objective

Build a system that:

1. Allows student registration into course batches.
2. **one student can register for only one course at a time**.
3. Generates a unique **Registration ID** for each registration.
4. Implements a **flexible, extendable Discount System**.
5. Allows Admin to configure and assign discounts dynamically.

📑 Functional Requirements

1️⃣ Student Management

- Student can register using:
    - Name
    - Email (unique identifier)
    - Phone
- Same email may create multiple registrations over time.

2️⃣ Course & Batch Management

- Admin can:
    - Create Courses
    - Create Batches under Courses
    - Set batch fee amount
    - Define batch capacity, batch location/mode

3️⃣ Registration Rules

- On registration:
    - Student selects a batch.
    - Fee is calculated.
    - A single discount coupon may be applied.
    - Final payable amount is calculated and stored.
    - System generates a unique Registration ID.

💰 Discount System

🎯 Key Rules

1. A student can apply **only ONE discount per registration**.
2. Admin can:
   - Create new discount types.
   - Configure discount value (flat or percentage) and applicable dates (start date, end date).
   - Assign discounts to:
     - Specific batches
     - Specific students
     - Specific time windows
3. Discount amount is determined dynamically by admin configuration.
4. The system must be extendible to allow **future discount types without major refactoring**.

🧩 Supported Discount Types

Your system must support a flexible mechanism to implement discount types such as:

- Early Bird Discount (Date-based)
- Loyalty Discount (Based on previous completed courses)
- Individual Discount (Student-specific)
- Combo Batch Offer (Predefined batch combinations)
- Flat Amount or Percentage-Based Discount
- Group Discount (Based on number of students) -- Bonus implementation

⚠️ Student may apply only ONE discount at checkout.

🏗️ Design Expectations

Participants are expected to:

- Design a modular and extensible discount engine.

- Avoid hard-coded discount logic.
- Separate business logic from UI.
- Validate edge cases.

🧠 Key Technical Expectations

Your implementation should demonstrate:

- Clean code practices
- SOLID principles (as applicable)
- Proper data modeling
- Validation and error handling
- Extensibility of discount mechanism
- Clear separation of concerns
- Smart use of AI tools

🚫 Constraints

- Only one discount can be applied per registration.
- Discount values (percentage or flat) and dates must be configurable by Admin.
- Discount logic should not be hard-coded.

📦 Deliverables

- Source Code
- Short README explaining:
  - Architecture

    - Discount strategy
    - Assumptions
    - How to run
- Short demo presentation

Mandatory and Optional features

Mandatory Features

- Student registration
- Course & batch creation
- Student registration
- Flexible discount engine
- Registration/Payments report

Optional Features

- UI polish
- Advanced group discount logic
- Concurrency handling
- Admin dashboard Full UI