

Event Management System — MVP Specification

1. Business Requirements (10)

- 1 User registration & login — Organizers and participants can create accounts and log in.
- 2 Create events — Organizers can create new events with title, date, location, and capacity.
- 3 Browse events — Participants can view all upcoming events and filter by date or category.
- 4 Register for event — Participants can register for available events if seats remain.
- 5 Cancel registration — Participants can cancel their registration before the event date.
- 6 Manage attendees — Organizers can view and manage their event participants list.
- 7 Update event details — Organizers can modify event info (time, location, description).
- 8 Notifications — Notify users on successful registration or cancellation.
- 9 Admin dashboard — Admins can view all users and events; manage categories.
- 10 Reports — Generate summary reports (registrations per event, attendance stats).

2. MVP Features (5)

- **User & Role Management** — Basic authentication with JWT; roles for organizer, participant, and admin.
- **Event CRUD Operations** — Organizers create, read, update, delete events. Includes transactional updates for event and capacity management.
- **Event Registration Flow** — Participants register or cancel registration; transactional handling to ensure seat counts remain consistent.
- **Search & Filter** — Public API to list or filter events by date, category, or keyword.
- **Admin & Reports** — Admin access for viewing reports, user statistics, and event summaries.

3. Data Model Summary

Entities: User, Event, Registration, Category, Notification, ReportLog.

Relations: User→Event (1–N for organizer), Event→Registration (1–N), User→Registration (1–N), Event→Category (M–N).

4. Transaction & AOP Notes

- Use `@Transactional` for registration and cancellation operations to maintain accurate seat counts.
- Apply `@Aspect` for performance logging, event updates, and audit actions.
- Use `@ControllerAdvice` for consistent error responses (validation, conflicts, or not-found).

5. Example Endpoints

- POST /api/v1/auth/register
- POST /api/v1/auth/login
- POST /api/v1/events
- GET /api/v1/events
- GET /api/v1/events/{id}
- POST /api/v1/events/{id}/register
- DELETE /api/v1/events/{id}/cancel
- PUT /api/v1/events/{id}
- DELETE /api/v1/events/{id}
- GET /api/v1/admin/reports