Proposal for Enhancing Event Functionality with Chapter Location and Analytics Dashboard

Overview

This proposal integrates chapter features into the event functionality, aiming to enhance the user and organizer experience through location-based features and a dedicated analytics dashboard. These features address current challenges and provide insights into user behavior and event performance.

Background

Currently, chapters in the app lack geographic location data. Users have to infer chapter locations from names (e.g., "SF" for San Francisco), which makes it difficult to discover events in nearby areas. Additionally, the event admin interface lacks cross-event insights, leaving organizers with limited data to optimize event planning and performance.

Proposal

This proposal aims to:

1. Add geographic location data to the chapter model.

2. Introduce an Analytics Dashboard to provide event organizers with key performance insights.

Features to be Implemented

Chapter Location Data

- Instead of inferring location from chapter names, we'll add specific location fields (city, state/region, country, coordinates) to the chapter model.

- This will enable:

  o Nearby event discovery

  o City-specific chapters, especially for large metroplexes

  o Better event data for organizers

  o "Event Passport" feature for tracking prior events

Analytics Dashboard

- A lightweight dashboard to help organizers track performance metrics across events.

- **Phase 1**: MVP includes:

  - Overview cards with basic KPIs (total events, total demos, total attendees, average feedback)

  - Chapter comparison table

  - Recent events list

Technical Design

1. **Chapter Model Update**

   - Add location fields (city, state, country, coordinates) to the chapter model.

   - Update the admin UI for chapter location management.

2. **Analytics Dashboard**

   - Leverage existing data from the Event, Submission, Demo, Attendee, Feedback, Vote, Award, and Chapter models.

   - Implement the following:

     - **Overview Cards** for KPIs (total events, total demos, total attendees, average feedback)

     - **Chapter Comparison Table** to compare event performance by chapter

     - **Recent Events List** for quick insights on recent event statistics

Implementation Plan

1. **Chapter Location**

   - Update the database schema and run migrations.

   - Tag all chapters with location data.

- Update the admin UI for location management.

- Implement the "Event Passport" page for attendees and "Organizer Demographics" page for event data visualization.

2. **Analytics Dashboard**

- **Backend**: Create a new analytics.ts router for fetching event data and metrics.

- **Frontend**: Build /admin/analytics page with stat cards, chapter comparison table, and recent events list.

Success Metrics

1. **Chapter Location**

- Increased event attendance from non-local chapters.

- Positive feedback on the event discovery and "Event Passport" features.

2. **Analytics Dashboard**

- Increased use of the analytics dashboard by event organizers.

- Positive feedback from organizers about data insights and decision-making improvements.

Risks & Unknowns

- **Chapter Location**: Will users find value in the ability to see events in nearby chapters?

- **Analytics Dashboard**: Do organizers currently need this level of cross-event insights, or are their manual tools (e.g., Google Sheets) sufficient?

Future Directions (If MVP Succeeds)

1. **For Chapter Location**:

- Expand features to include time-based event tracking, allowing attendees to view past events visited based on location.

2. **For Analytics Dashboard**:
    - o After validating the MVP, consider adding:
        - ▪ Time-series trends
        - ▪ Detailed winner analysis
        - ▪ Advanced filtering and export options

Conclusion

This proposal combines two key features to improve both the user and organizer experience: enhanced chapter location data for better event discovery and a lightweight analytics dashboard to provide valuable insights into event performance. The phased implementation approach ensures quick feedback and iteration, allowing for scaling based on user needs.