Google Developer Groups

On Campus • Indian Institute of Information Technology Kalyani

Winter of Code 5.0

Proposal

for the Project

“ORYCON — OSC Internal Event Management Tool”

Under

OPEN SOURCE CHANDIGARH

About Me

Name – Vishal Dubey
Email-Id - dubeyvishal714@gmail.com
GitHub Username - https://github.com/Vishal34sd
Country - India
Timezone - IST (UTC +5:30)
Primary Language - JavaScript / TypeScript
LinkedIn Profile Link -
https://www.linkedin.com/in/vishal-dubey-478541320/
Portfolio Link- https://new-portfolio-website-seven.vercel.app/

Synopsis

I am proposing to contribute to ORYCON — OSC Internal Event Management Tool, a scalable platform designed to streamline event organization, content creation, social media workflows, attendance tracking, and logistics management.

This project is meaningful because it solves real operational challenges faced by student communities when running events—especially when multiple teams collaborate under strict deadlines. ORYCON brings all critical workflows (attendance, teams, planning, tasks, content drafting, social media scheduling, logistics coordination) into one place.

My contribution will focus on developing core modules that directly improve event execution and team collaboration while ensuring the code remains clean, maintainable, and open-source friendly for future contributors.

Benefits to the Community

My contributions will help OSC and contributors/community in the following ways:

- **Smooth event execution: attendance automation + admin workflows**
- **Better collaboration: task tracking + structured team operations**
- **Improved security: RBAC prevents feature misuse across roles**
- **Higher adoption & easier contributions: better docs + setup guide**
- **Maintainability: modular code structure + clean APIs**

Overall, my work will make ORYCON more production-ready and easier to extend.

Project Plan

To contribute effectively to ORYCON, I will first understand the existing monorepo structure, Prisma schema, and role-based workflows used across teams. I will start by picking a small issue/cleanup PR to get familiar with the coding standards and contribution flow, then move to core feature development.

My approach will be:

- **Set up the project locally (client + server) and study current APIs, DB models, and UI flows**
- **Break the work into small, review-friendly pull requests (feature → integration → polish)**
- **Implement features with clean modular code, consistent API responses, and proper validations**
- **Add documentation updates (README, .env, API details) alongside development**
- **Test key flows manually + add minimal automated tests where needed (critical APIs)**

Tech Stack I'll Use-

- **Frontend: Next.js (React) + TypeScript**
- **Backend: Node.js + Express.js**
- **Database: MongoDB**

- **ORM/ODM: Prisma**
- **Monorepo Tools: Turborepo**
- **Containerization: Docker (for consistent dev setup)**
- **Testing (where applicable): Jest / Supertest (API testing)**
-

This plan ensures that my contributions are not only feature-focused but also stable, maintainable, and easy for future contributors to extend.

Milestones

Milestone	Tentative Date	KPI
Week 1	Day 1–7	Setup project (client + server), understand existing backend APIs + Prisma models, map UI flow for each role, fix minor issue/doc update, **1 PR merged**
Week 2	Day 8–14	**Auth + RBAC Frontend**: Login/Register UI, protected routes, role-based dashboard layout, session handling, connect with backend APIs, **1 PR merged**
Week 3	Day 15-21	**Attendance UI Phase-1**: Attendance dashboard page, QR display page, scan UI (camera / upload), connect scan API, handle duplicate prevention & error states, **1 PR merged**
Week 4	Day 22 - 28	**Attendance UI Phase-2 + Backend gaps**: walk-in entry UI, attendance history UI, export CSV button + backend fix if needed, live count view, **1 PR merged**
Week 5	Day 29-35	**Student Management UI (Admin)**: student list page, profile details page, search/filter/pagination, disciplinary action UI + required backend patch if missing, **1 PR merged**
Week 6	Day 36-42	**Organising Module UI + Final Polish**: event tasks + milestones UI, assign tasks UI, UI consistency, bug fixes, refactor reusable components, final docs + walkthrough demo, **final PR merged**

Deliverables

By the end of the contribution timeline, I will deliver the following:

- **Role-based Frontend Dashboards**
    - Dedicated dashboards/pages for Students, Team Admins, and team roles (Content/Social/Organising/Logistics)
    - Protected routes + role-based UI rendering integrated with RBAC
- **Authentication UI + Integration**
    - Login/registration UI (as per existing backend)
    - Forgot/reset password UI flows
    - Session handling and secure API integration
- **Attendance Management Frontend**
    - QR attendance UI (QR display + scan/check-in interface)
    - Real-time attendance updates and error handling (duplicate scan prevention, invalid QR, etc.)
    - Walk-in entry UI support (if required by flow)
    - Attendance history page + export option (CSV) integration
- **Student Management Dashboard (Team Admin)**
    - Student list view with search/filter/pagination

    - Student profile view linked to attendance
    - Disciplinary actions UI integrated with backend
- **Organising Team Workflow UI**
    - Task management UI (create/assign/update status)
    - Milestone tracking/checklist view for event planning
- **Backend Gap Fixes (If Needed)**
    - Minor backend patches required for frontend integration (validation, missing endpoints, export route fixes, response format consistency)
- **Documentation & Contributor Support**
    - Updated README and setup steps for smooth onboarding
    - API usage notes (frontend integration points, env variables)
    - Screenshots/GIFs of implemented flows for maintainers
- **Final Delivery**
    - Clean, review-friendly PRs with clear commit history
    - Final summary/demo walkthrough of completed features

Acknowledgement

I sincerely appreciate Open Source Chandigarh for creating an open and supportive environment for students to learn and contribute through Winter of Code. I’m grateful to the maintainers and mentors for their guidance and the effort they put into building meaningful open-source projects. I’m excited to contribute to ORYCON and will ensure consistent communication, quality pull requests, and reliable deliverables throughout the program.