Assignment / Tutorial Team:

24MX108- Kalaivanan R

24MX109 – Kannan M

24MX111 – Karthikeyan R

24MX217 – Mokesh C

DevOps Topologies for LOGIN (Event Website)

Goal: To develop and maintain the **LOGIN Website**, a web platform used for participant registration, event information, schedule updates, announcements, and sponsor showcases. Each year, the platform is updated by new student teams while retaining the base infrastructure such as hosting, domain mapping, and CI/CD pipelines.

Best Suited DevOps Topologies:

Type 1: Dev and Ops Collaboration (Unified Team)

In this model, the development and operations responsibilities are shared within a single team, allowing for rapid iterations, streamlined deployment, and effective ownership of both the application and its infrastructure.

Justification for Type 1

- **Development -** Student team builds UI, handles routing, data, and interactive features.
- **Operation -** CI/CD pipelines, environment configs, deployment scripts managed by Dev team.
- **Hosting -** Integrated with GitHub for auto-deploy via Vercel.
- **Faculty Involvement** - Limited to initial DNS/domain access support.
- **Agility & Learning** - Student team gains full-stack DevOps experience.

Topologies to Avoid for Small Projects:

- **Type 2 (Shared Ops):** Overhead of shared Ops isn't needed; student team manages all Ops tasks themselves.
- **Type 3 (Ops-as-IaaS):** No dedicated infrastructure team — faculty only assist with domain mapping.

- **Type 4 (DevOps-as-a-Service):** Outsourcing DevOps would remove hands-on learning for students.
- **Type 5 (Temporary DevOps Team:** Not a fresh setup — infra and practices are reused each year.
- **Type 6–9:** These reflect poor collaboration, silos, or misaligned roles — unsuitable for student-led academic projects.

Requirements:

Functional Requirements:

- **Participant Registration -** Event-specific forms with validation and backend integration.
- **Role-Based Dashboards -** Different views for participants, coordinators, and admins.
- **Announcement & Results System -** Live updates, result posting, and notifications.
- **Continuous Deployment** - Auto-deploy via GitHub commits using CI/CD pipelines.

Non-Functional Requirements:

- **High Availability -** Handled by Vercel's cloud platform and edge network.
- **HTTPS & Security -** SSL handled via Vercel; secure environment variables for tokens.
- **Responsiveness -** Optimized for all screen sizes using Tailwind CSS.
- **Performance** - CDN caching, image optimization, lazy loading.

Stakeholders:

- **Student Dev Team -** Develop, test, deploy, and manage full application lifecycle.
- **Faculty/IT Support** - Provide access to college domain for DNS configuration.
- **Event Coordinators** - Input event schedules, results, and updates.
- **Participants** - End users who register and access event-related information.

- **Sponsors** - Organizations whose branding is featured on the site.
- **Institutional Admin** - Oversight of content accuracy, branding, and institutional compliance

---

Knowledge Level of Team Members:

1. **Frontend:** Next.js, Tailwind CSS, Framer Motion.
2. **Backend:** API routes (Next.js), form handling, authentication logic, basic database/JSON usage.
3. **DevOps:** GitHub Actions, Vercel CLI, environment variables, version control.
4. **Monitoring & Logs:** Preview build logs, deploy history, GitHub Issues.
5. **Collaboration:** Git, GitHub Projects, code reviews, pair programming.

Summary:

The selected DevOps Topology for the LOGIN College Event Website is **Type 1 – Dev and Ops Collaboration**, where the student team takes full responsibility for both development and deployment. The project builds on existing infrastructure, reused each year by a new batch with only minimal support from faculty for domain-level tasks. This model is ideal for educational projects, as it promotes fast updates, hands-on learning, and smooth handover between teams. It allows students to experience the complete DevOps cycle while keeping operational dependency low.