**1. Web/Mobile Application Development: "Campus Connect - A Unified Platform for Student Engagement and Resource Management."**

**Problem Background:** In modern educational institutions, students often face fragmented access to campus resources, leading to inefficiencies in daily academic life. Information about events, class schedules, library availability, hostel management, and peer collaboration is scattered across multiple platforms (e.g., emails, WhatsApp groups, university portals, and physical notice boards). This results in missed opportunities, delayed communications, and reduced student engagement. During a 24-hour hackathon, participants are challenged to build a comprehensive web or mobile application that centralizes these resources, fostering a more connected and efficient campus ecosystem.

**Problem Statement:** Develop "Campus Connect," a responsive web or mobile application (or hybrid) that serves as a one-stop platform for students, faculty, and administrators at Shri Vaishnav Vidyapeeth Vishwavidyalaya. The app should address the silos in campus information by integrating features for real-time updates, collaboration, and resource optimization. Key challenges include handling user authentication securely, ensuring cross-platform compatibility (web for desktops and mobile for on-the-go access), and incorporating user-friendly interfaces to encourage adoption among tech-savvy and non-technical users alike.

**Objectives:**

- **Core Features:**
    - User profiles with role-based access (student, faculty, admin).
    - Real-time notifications for events, deadlines, and announcements using push notifications or in-app alerts.
    - Integrated calendar for class schedules, exams, and extracurricular activities with reminders.
    - Resource booking system (e.g., library books, lab equipment, or study rooms) with availability checks and reservations.
    - Social collaboration tools, such as forums, group chats, or project matchmaking for hackathons and group assignments.
- **Technical Requirements:**
    - Use modern frameworks like React.js/Vue.js for web, Flutter/React Native for mobile, or Progressive Web App (PWA) technology for hybrid deployment.
    - Implement backend services (e.g., Node.js, Firebase, or Django) for data storage, API integrations (e.g., Google Calendar API for syncing), and real-time features (e.g., WebSockets or Firebase Realtime Database).
    - Ensure data privacy compliance (e.g., basic GDPR-like principles) with secure authentication (OAuth or JWT).
    - Optimize for performance on low-bandwidth networks, common in campus settings.

- **Constraints and Scope for 24-Hour Hackathon:**
    - Focus on MVP (Minimum Viable Product) with at least 3-4 core features fully functional.
    - No need for full-scale deployment; a demo version with mock data is sufficient.
    - Incorporate accessibility features like dark mode, voice navigation, or multilingual support (English/Hindi) to cater to diverse users.
- **Expected Impact and Evaluation:**

  The solution should demonstrate how it reduces information overload and boosts productivity. Evaluation will consider innovation, user experience (UX/UI design), scalability, and integration of web/mobile best practices. Bonus for incorporating gamification elements, like badges for active participation, to enhance engagement.