**KEY DESIGN PATTERNS FOR A REAL-TIME NOTIFICATION SYSTEM FOR A COMMUNITY EVENT APP**

The research focuses on identifying the most effective design patterns for building a real-time notification system for a community event web application. The goal is to ensure instant updates, support multiple users registering simultaneously, and maintain system reliability.

The **Observer/Publish–Subscribe pattern** enables instant, real-time notifications.  
**Event Sourcing and CQRS patterns** manage simultaneous user actions and maintain a complete activity log.  
**Optimistic Concurrency Control** ensures that only one user can register for the last available spot without conflicts.  
The **Mediator pattern** coordinates communication between components such as registration, notification, and AI services.  
The **Strategy pattern** supports multiple notification delivery methods, such as email or push alerts.  
The **Singleton pattern** efficiently manages shared global resources like the notification dispatcher.  
Finally, the **Factory pattern** dynamically creates notification types based on user preferences, improving scalability and flexibility.

Collectively, these design patterns make the system responsive, consistent, and scalable under real-time conditions.