**취미 공유 및 이벤트 플랫폼: 공통 관심사를 통한 사람들과의 연결**

Step-by-step:

**1. Project Planning / 프로젝트 계획**

* **Define project goals and features**  
  프로젝트 목표 및 기능 정의
  + Clearly outline the objectives, key features, and target audience.  
    목표, 주요 기능, 대상 사용자에 대해 명확히 정의합니다.
* **Create wireframes and UI/UX design**  
  와이어프레임 및 UI/UX 디자인 생성
  + Develop wireframes to visualize the layout of the website and app.  
    웹사이트 및 앱 레이아웃을 시각화하기 위해 와이어프레임을 작성합니다.

**2. Set up the Development Environment / 개발 환경 설정**

* **Choose your tech stack**  
  기술 스택 선택
  + Finalize the frontend (React.js/Vue.js) and backend (Node.js/PHP), and select a database (MySQL/PostgreSQL).  
    프론트엔드(React.js/Vue.js), 백엔드(Node.js/PHP), 데이터베이스(MySQL/PostgreSQL)를 최종 결정합니다.
* **Set up version control (Git)**  
  버전 관리 설정 (Git)
  + Initialize a Git repository for collaboration and tracking.  
    협업 및 추적을 위해 Git 저장소를 초기화합니다.

**3. Backend Development / 백엔드 개발**

* **Set up the database**  
  데이터베이스 설정
  + Design database schema for users, events, hobbies, and RSVPs.  
    사용자, 이벤트, 취미, RSVP를 위한 데이터베이스 스키마를 설계합니다.
* **Develop user authentication**  
  사용자 인증 개발
  + Implement secure login/signup functionality using JWT or OAuth.  
    JWT 또는 OAuth를 사용하여 안전한 로그인/회원가입 기능을 구현합니다.
* **Create RESTful API for events and users**  
  이벤트 및 사용자에 대한 RESTful API 생성
  + Develop APIs for creating, retrieving, updating, and deleting events.  
    이벤트 생성, 조회, 수정 및 삭제를 위한 API를 개발합니다.
* **Implement notification system**  
  알림 시스템 구현
  + Set up email or in-app notifications for event reminders and updates.  
    이메일 또는 앱 내 알림을 통해 이벤트 알림 및 업데이트 시스템을 설정합니다.

**4. Frontend Development / 프론트엔드 개발**

* **Build the user interface**  
  사용자 인터페이스 구축
  + Design and code the homepage, user dashboard, and event creation page.  
    홈페이지, 사용자 대시보드 및 이벤트 생성 페이지를 디자인하고 코딩합니다.
* **Integrate map for event locations**  
  이벤트 위치를 위한 지도 통합
  + Use Google Maps API or OpenStreetMap to display event locations.  
    Google Maps API 또는 OpenStreetMap을 사용하여 이벤트 위치를 표시합니다.
* **Implement event search and filters**  
  이벤트 검색 및 필터 기능 구현
  + Add search functionality for events based on location, hobby, and date.  
    위치, 취미 및 날짜를 기준으로 이벤트 검색 기능을 추가합니다.
* **Add RSVP functionality**  
  RSVP 기능 추가
  + Allow users to confirm attendance and manage RSVPs for events.  
    사용자가 이벤트 참석을 확인하고 RSVP를 관리할 수 있도록 합니다.

**5. Testing and Debugging / 테스트 및 디버깅**

* **Unit and integration testing**  
  단위 및 통합 테스트
  + Test individual components and their interaction within the platform.  
    플랫폼 내에서 개별 구성 요소 및 상호작용을 테스트합니다.
* **User acceptance testing**  
  사용자 수용 테스트
  + Ensure the platform works as expected for end users.  
    플랫폼이 최종 사용자에게 예상대로 작동하는지 확인합니다.

**6. Deployment / 배포**

* **Set up hosting environment**  
  호스팅 환경 설정
  + Choose a cloud service (e.g., AWS, Heroku) for hosting the application.  
    애플리케이션 호스팅을 위한 클라우드 서비스(AWS, Heroku 등)를 선택합니다.
* **Deploy the application**  
  애플리케이션 배포
  + Deploy the frontend and backend to the cloud hosting environment.  
    프론트엔드 및 백엔드를 클라우드 호스팅 환경에 배포합니다.