PRD: Initial Web Version Social Platform

(Note: This is the initial web version PRD, aligned with my PRD Writing Principles.)

# 1. Background

For students and young people, there is a strong need to make new friends, organize gatherings, and discover nearby dining and entertainment. Existing platforms (e.g., WeChat, Momo, Facebook) are either overly complex or lack the campus scenario.  
Our goal: build a lightweight web-based social platform (MVP) to validate core needs.

# 2. Product Goals

1. Allow users to register and complete their profile within 5 minutes.  
2. Enable location-based “Discover Nearby People” feature.  
3. Provide basic social features: add friends and chat.  
4. Offer nearby dining & entertainment recommendations to support offline meetups.  
5. Lay the foundation for future expansion into a mobile app.

## Success Metrics

- Registration completion rate ≥ 80%  
- Average of ≥ 2 new friend connections in the first week  
- Chat message delivery rate ≥ 99%  
- Recommendation click-through rate ≥ 30%

# 3. Personas

- Persona A: Freshman  
 - Scenario: Just arrived on campus, wants to quickly make friends.  
 - Goal: Find peers in the same major/dorm.  
  
- Persona B: Event Organizer  
 - Scenario: Wants to organize group meals or gatherings.  
 - Goal: Meet people and find recommended nearby places.

# 4. User Goals & Tasks

- Goal for A: Quickly meet classmates → Tasks: Register → Browse nearby people → Send friend request.  
- Goal for B: Organize gatherings → Tasks: Discover nearby people → Create chat/group → View recommendations → Meet.

# 5. Product Principles

1. Safety over efficiency: Protect user privacy > fast matching.  
2. Core actions ≤ 3 steps: Adding friends, chatting, and viewing recommendations must be simple.  
3. Privacy by default: Only minimal required info is public; the rest remains anonymous.  
4. Chat experience like IM: Smooth and reliable, similar to WeChat.  
5. Follow standard user habits: Align with common social product conventions.

# 6. Assumptions

- Users are willing to socialize via a web app (needs validation).  
- IP-based location is sufficient for “nearby” discovery (needs validation).  
- Text chat alone is enough for initial communication (needs user testing).

# 7. Features

## 7.1 Registration & Login

- Email registration and login.  
- Required fields: nickname, interests, school/city.

## 7.2 User Profile

- Display avatar, interests, basic info.  
- User can edit, with strict default privacy settings.

## 7.3 Discover Nearby People

- List people based on IP location.  
- Filter by interests.

## 7.4 Friends & Chat

- Send/accept friend requests.  
- After confirmation, start text chat.

## 7.5 Recommendations

- Pull dining/entertainment places via APIs (Yelp/Google Maps).  
- Show name, rating, and distance.

# 8. Release Standards

- Performance: End-to-end chat latency ≤ 300ms (P95).  
- Reliability: Message delivery rate ≥ 99%.  
- Scalability: Support up to 1000 concurrent users.  
- Security: User privacy not public by default; sensitive fields require opt-in.  
- Usability: Registration + adding a friend must be completed within 5 minutes.

# 9. Prioritization

- Must-have: Registration/login, discover nearby people, friend requests, chat.  
- High-want: Recommendations, interest filters.  
- Nice-to-have: Group chat, advanced recommendation algorithm, UI enhancements.

# 10. Risks & Limitations

- IP-based location may be inaccurate, leading to broad ranges.  
- Chat only supports text; lacks multimedia experience.  
- Web-only version may limit user retention.

# 11. Traceability Example

- Anonymous chat → Supports goal: Safety & privacy protection.  
- Recommendations → Supports goal: Organize offline gatherings.  
- Interest filters → Supports goal: Efficient matching.