CS407 Project Proposal

Title: Knot

Liye Fu, Email: lfu49@wisc.edu, GitHub Login Id: liyefu-a
Xiangyi Peng, Emails: xpeng83@wisc.edu, GitHub Login Id: Tardirisu
Enhui Zhao, Email: ezhao37@wisc.edu, GitHub Login Id: Enhui000
Shenwei Zhang, Email: szhang829@wisc.edu, GitHub Login Id: zsw0715

Introduce Our App:

In the current social media environment, the information flow is linear, fragmented, and disconnected from our physical world. We cannot intuitively see the connection between our friends' daily lives and specific locations. For close friends, there is a lack of a shared, geographically-based private space to leave behind contextual "digital footprints" - whether it's a joke in the cafeteria, a request to study in the library, or a funny emoji on the Mendota Lake.

Since youngsters are not satisfied with current social media interactions, they are finding ways to create functional and small platforms to express themselves. In this way, our application Knot is aiming at transforming the map into an interactive canvas exclusively for you and your close friends. It is not a traditional social network but a location-based asynchronous social space. Users can leave three types of tags on the real-world map:

- ✧ Comments: Make comments or record information about the location.
- ✧ Requests: Initiate a specific invitation or question, and reply directly in the tag with a sequential line.
- ✧ Stickers: Express your mood with fun stickers and mark the location.

All interactions are only visible to friends, forming a private shared map diary. Its core innovation lies in deeply integrating social interaction with geographical location, making communication more immersive and ceremonial.

Our target users are young people who have small, stable, and close social circles, especially college student groups, couples, and friends who often gather offline, and who are not satisfied with the superficial interactions offered by traditional social media like Instagram or TikTok and yearn for a more private, context-rich and closer-to-real-life communication method.

Our Competition:

1. Our competitors:

Zenly or Snapchat: Focuses on the presentation of real-time locations, concentrating on "social dynamic maps". Our product is a collaborative map for friends or a location diary. We not only pay attention to the real-time status of "people” but also consider the history and collaborative value of "locations".

大众点评(dianping) or yelp: Providing location-based reviews, but the content is public and intended for a broad audience, lacking the privacy among friends and the lighthearted, interactive fun with real-time connection.
(Zenly): https://zenly.com
(Snapchat, p.s.: similar to Snap Map but more): https://www.snapchat.com
(Yelp): https://www.yelp.com
(大众点评 dianping): https://www.dianping.com

2. Our unique advantages and niches

The unique advantage of Knot lies in pioneering the "map-based asynchronous social" market segment. Compared with competitors, our differences are as follows:

- ✧ We not only share locations, but also turn the map itself into an interactive social platform, supporting various interaction forms such as comments, requests, and stickers, allowing content to be accumulated.
- ✧ All content is shared only among friends, creating a safe and trustworthy communication environment.
- ✧ Communication is strongly related to specific locations, making invitations, sharing, and recording more contextually relevant. More specifically, the "location" scenario is automatically provided. People don't need to ask, "where to go", but clearly know which store it is and can even see the surrounding environment.

Main Modules of Our App:

1. **Frontend: (Kotlin)**
    - a. Screens: including Chat Message, Friend Management, User Management, Login/Register Screen, Main Interactive Map
    - b. WebSocket： Establish and maintain long connections with the backend Netty Server to receive push messages such as real-time location updates, new tag notifications, and response requests.
    - c. Map Android SDK to make an interactive map. MapBox is a good option here: https://docs.mapbox.com/android/maps/guides/

2. **Backend: (Java / Spring Boot & Netty)**
    - a. RESTful API (Spring Boot Controllers)
        - a) Handle all non-real-time, request-response mode HTTP(S) requests.
        - b) Including: user registration/login/logout, obtaining/refreshing JWT Token, friend relationship management (search/add/delete/set permissions), addition/deletion/modification/search of tags (non-real-time part, such as loading historical tags), and user profile management.
    - b. Netty Server Module:
        - a) A TCP long connection server built based on the Netty framework.
        - b) Core responsibilities:
            - i. Manage all online users' TCP connections (Channels).
            - ii. Receive real-time data from clients (mainly location updates, tag creation/reply notifications).
            - iii. Route real-time messages to the target user's Channel based on business logic (such as friend relationships, permission settings).
            - iv. Handle heartbeat packets to maintain connections and detect client disconnection.
    - c. JWT + Redis
        - a) JWT (Access Token / Refresh Token): The REST API uses JWT Access Token for authentication. A Refresh Token mechanism is provided to update the Access Token.
        - b) Redis:
            - i. Store the Refresh Token (which is associated with the user ID and is used to refresh the Access Token).
            - ii. Store the online status of users and the mapping relationship between Netty Channels (userID -> Channel ID)
    - d. MySQL
        - a) Persistent Storage: All data that needs to be saved for a long time, has complex relationships, and requires transaction guarantees.
            - i. User Information (users)
            - ii. Friendship Relationships (friendships)
            - iii. Tag Information (tags: type, content, location coordinates, creator, creation time)

Mobile Innovation:

Knot’ s innovation is based on smartphone sensors and its mobility characterization to achieve a unique and novel experience. Unlike desktop applications, Knot thrives on mobility, where users can develop and improve their friendship by sharing mutual locations and thoughts about them.

1. GPS-driven interaction: Use mobile devices with sensors such as GPS, cameras, etc. to achieve location-related social functions, with all tags, requests, questions, and stickers related to coordinates in the real world. Utilize the location information of the device, provide social interactions associated with the current location.
2. Real-time capability: Achieve real-time location and share chat messages only with close friends.
3. Camera integration: Attach photos or videos to map tags for users.

What We Need to Test Our App:

We plan to test Knot primarily on devices based on the Android platform. Among our group members, we currently have access to three Android phones, which will allow us to test key features such as GPS, camera integration, and live tag notifications.

Other ways we will take to test our app are:

1. Use the Android Studio emulator on laptops for development and debugging.
2. Test with the two real devices to ensure normal functionality in authentic mobile conditions and interactive functionalities among multiple devices.
3. Conduct real-world group trials to validate stability, usability, and user experience for further improvements.