SeniorConnect System

System Requirement Specification

GruFamily Project Team

Nanyang Technological University

50 Nanyang Avenue

Singapore 639798

22/09/2015

Revision History

Name	Date	A*MD	Reason For Changes	Version
Wang Siqi	10/09/2015	A, M	Draft system requirement specification	V1.1
Mao Huiqi	15/09/2015	A,M	Add in non-function requirements and interface requirements	V1.2
Ma Xiaoxue	18/09/2015	A	Add in functional requirements	V1.3
Mao Huiqi	20/09/2015	M	Resolve the issues and problems uncovered during System Requirement Review Meeting. Add-in several sections and modified some of the requirements as suggested in the meeting.	V2.1
Mao Huiqi	20/09/2015	M	When SW SQA team is developing the test case, it was found that the requirements in SRS are not atomic enough. Break down large requirement items into smaller ones and add in missing specifications according to use case.	V2.2
Ma Xiaoxue	21/09/2015	A	Add in activity diagrams	V2.3
Wang Siqi	22/09/2015	M	Finalize use case	V3

*A - Added M - Modified D – Deleted

SeniorConnect System

System Requirement Specification

22/09/2015

Software Requirement Specification Approvals:

LI YISHAN 22/09/2015

SQA Manager **Date**

刘熠坤 22/09/2015

Project Manager **Date**

Contents

1. Problem Statement

Population aging and social isolation among the elderly are becoming increasingly severe problems in nowadays society, especially in developed countries. According to World Population Aging Report published by United Nations in 2013, the global share of older people (aged 60 years or over) increased from 9.2 per cent in 1990 to 11.7 per cent in 2013 and will continue to grow as a proportion of the world population, reaching 21.1 percent by 2050. Globally, 40 per cent of older persons aged 60 years or over live independently, that is to say, alone or with their spouse only. As countries develop and their populations continue to age, living alone or with a spouse only will likely become much more common among older people in the future.

Researches have shown that old people with fewer social lives are more likely to die even if they are happy in their solitude. This suggests the danger of social isolation among the elderly. In order to maintain physical and mental health, seniors could connect with their family and friends frequently as well as participate in various interesting activities to enhance their social connectedness.

Social Networking Services (SNS) could be an effective and efficient solution. According to a two-year project funded by European Union, training elderly to use social media will improve cognitive capacity, increase a sense of self-competence and could have a beneficial overall impact on mental health and well-being (University of Exeter, 2014).

We, the Gru Family team proposes to develop a social network software, SeniorConnect, for the elderly, in order to satisfy their emotional, social, mental and learning needs and enrich their life. The ultimate goal of this software is to assist elderly people to communicate with each other easily and to get involved in social activities actively, which will lead them to healthier and happier life.

2. Overview

2.1. Background

Social isolation is becoming a more and more severe problem among seniors in nowadays society. Most elderly often lack the opportunity to communicate with their children and old friends, and they are unable to catch up with the changes in their surroundings. Therefore, special attention should be devoted to help the elderly maintain their existing connections as well as enrich their life.

There are some platforms existing in the market targeting on friendship establishment, family interaction and social connectedness. However, most of them are not user-friendly to and thus not widely used by the elderly. It would be helpful to develop a social platform especially catering to needs of the elderly.

Specifically, SeniorConnect is a social media application which provides a platform to engage the elderly in social life by allowing chatting, sharing and joining events, and to help the elderly manage their life for daily activities. It will be designed and released as a mobile application so that elderly people can access to it anytime and anywhere. Our design emphasizes on convenient voice channel, simple GUI with enlarged fronts, accessible functionalities and security, to provide a reliable and effective social platform for the elderly.

The project is scheduled to be carried out from August 2015 to June 2016 by the 6-memebr team, the Gru Family, with an estimated budget of S$147,070.

2.2. Overall Description

Driven by the preceding demands, our group proposes a mobile phone application to primarily facilitate and to further promote social connectedness of the elderly. The application is designed to achieve the following objectives.

a. The application should allow users to make voice calls, video calls and send voice messages to other users as communication channels. The application will not include text messaging because it is not an easy to use function for most elderly.

b. The application should allow users to join communities categorized by hobbies and other topics. The community should provide online learning courses for group members to take. The application should also allow community members to receive notifications of news updates and invitations for events organized by communities.

c. The application should allow users to share their daily life events and experiences with friends, by providing a platform of information sharing and interactive communication.

The application is designed to be encouraging and motivating for the elderly in achieving those objectives. When taking online courses posted by community core members in the form of online videos and blogs, the application will encourage the elderly to complete the course by showing progress and motivate the learning by periodic reminders. The posted life events will be viewable by the friends and thus the elderly will be able to communicate and socialize under the posts by writing comments.

Additionally, the application also allows users to play braining-training games. The games are selected based on the effectiveness of stimulating different parts of brain and their level of popularity.

Taking the advantage of being a social media platform for the elderly and family, the application will also provide helpful tools for the everyday life. For example, in future, family members could set reminder to inform the elderly of the events they registered and the things they need to do such as taking medicine, as the elderly are weak in memory.

3. Investigation & Analysis Methodology

3.1. System Investigation

There are many existing systems in the market focus on social connectedness. However, almost none of them targets at the elderly. Most of them are therefore neither user friendly to nor widely used by the elderly, because of the intensive text input required, small font, assumption of familiarity with technology, and lack of features targeting at the elderly's special needs. Therefore, the proposed system will have to resolve those issues, by providing a novice and innovative platform for the elderly.

3.2. Analysis Methodology

3.2.1. Feasibility Study and Requirements Elicitation

The first thing to do is to identify the needs of targeted user group, the elderly. The elderly might have different social behaviors, which makes it difficult to abstract customer needs. Therefore, a **top-down approach** should be adopted, where people's needs are categorize first and more detailed areas, such as functionalities and interface designs, will be considered later.

In order to perform detailed analysis, the following should be carried out:

- Preliminary discussion with seniors and caregivers on how the elderly socialize in daily routines.
- Discussion and observations to discover the elderly's special needs due to physical and mental status.
- Survey on the connectedness of the elderly with family and friends.
- Detailed discussion on functionalities and interface with caregivers.

3.2.2. System analysis and requirements specification

A **spiral approach** will be adopted to identify and generate satisfactory specifications.

The user requirements gathered from the surveys and interviews will be used to decide on the key functionalities to be included in the product. A low fidelity prototype will be drafted to facilitate the illustration of the specifications. Short interviews with researchers in elderly research labs in Nanyang Technological University will also be conducted, to analyse and refine specifications defined previously. A high fidelity interface prototype will be generated based on the analysis and volunteers will be invited to simulate the usage of the product. Their feedback will be gathered to identify any potential improvement.

With the first draft version of mock-up, the product will be distributed among more seniors for investigation purpose. 50 users with age range from 50 to 70 will be reached out and their feedback will be collected in order for us to modify and consolidate the design. This will be done several times and in each iteration, an identified possible improvement will be evaluated. For example, if they feel better with a shortcut to send messages to a specific person, we will consider adding this feature.

This process will be repeated till conditions listed below are satisfied:

1. Over 80% of our volunteers feel satisfied with the functionality and find our GUI friendly.
2. Over 50% of volunteers show their interest in downloading and trying out the application.

With this spiral approach, we are confident in the identified requirements and specifications. Catering to the special needs of our target users, our application will definitely win a good market share.

3.2.3. System requirement documentation

The proposed system requirements and functionalities will to be documented using UML. Documentation includes the following for the core functionalities:

- Use case diagram
- Sequence diagram for complicated core functionalities

3.2.4. Prototyping

A fast prototype will be implemented for the system. The prototype will be a working example of the system for demonstration purpose, and will also be used in gather user's feedback for improvement in the spiral approach. It will include the following:

- A mobile application on Android and iOS
- A working server for data storage and business logic

4. Constraints

4.1. Hardware

The proposed system requires a robust Linux server in order to be functional.

4.2. Project Schedule

The delivery of system should be completed within one year. However, given the time and effort required for modification and improvement in the spiral approach, the first prototype should be developed within two months.

4.3. User Feedback

There is difficulty in getting user feedback as none of the team could evaluate the software from a user's perspective. User feedback from the elderly in the society may be misleading, as individual feedback is highly subject to the person's technology literacy, personal relationship, and cultural background.

5. Operational Requirements

5.1. Tutorial and Help Hotline

As most of our targeted user are not technology-savvy, a tutorial will be necessary for them to even start using the proposed system. The tutorial should be continuously evolving so as to cater to needs of the elderly based on feedback.

A help hotline should also be set up to help the targeted user with technical problems using our system. It will also play a key role in getting the elderly used to the system. People in charge of the hotline should be well educated in technology, familiar with the application, and preferable able to communicate with elderly on topics of technology in a manner that is easy to understand.

5.2. Service Maintenance

The service should be well maintained so that its uptime satisfies non-function requirement. Also, it should cater to different versions of clients when the system is released to the public.

5.3. Data Security

The data should be periodically backed up so that minimal amount of data are affected in the case of data loss.

5.4. Privacy Concern

The system is a social network application and therefore involves a lot of personal information. As such, strict control over database access should be implemented so that data are not accessed by unauthorized person.

6. Functional Requirements

6.1. Registration

System should provide log in function to SC system. Registration primarily consists of entering a unique phone number for verification and creating a password.

1. SC system must allow user to tap SC system button to start the application followed by the ‘New User’ button at login page.
2. SC system should direct user to the registration page and prompts out an input form for new user registration.
3. SC system should allow user to input phone number and password, and the conformation password, followed by submitting the form.
   3.1. SC system should allow user to select the cancel option.
   3.2. SC system should return the user to the login page without the user being logged in and any information entered has been erased.
4. SC system must validate and submit the input form for authentication if the entered information is valid.
   4.1. If user does not provide valid information, e.g. invalid phone number, too simple password, wrong reentered password etc., system must display information with appropriate message to correct invalid information.
   4.2. SC system should ask user to re-enters information, back to 3.
5. SC system should allow user to authenticate the submitted phone number and directs user to the main page if the phone number is authenticated.

6.2. Login/Logout

6.2.1. Login

System must provide log in function to SC system. SC system should allow user to use his/her phone number and password to log in to SC system.

1. SC system should allow user to tap SC system button to start the application.
2. System should prompt out an input form for user to use phone number and password to log in.
3. SC system should ask user to input phone number and password, and then SC system should allow user to submit the form.
4. SC system must authenticate the user with the given information and allows user to log in if the details are valid.
   4.1. If user does not provide correct login details, system should prompt out an error message to show user that either the phone number is not registered or the password is incorrect
   4.2. Back to 2.

6.2.2. Logout

System should provide log out function to SC system.

1. SC system should allow user to tap corresponding button to log out.
2. System should show log out is successful.

3. System should prompt out an input form for user to use phone number and password to log in later.

6.3. Chat

6.3.1. One-to-one chat

6.3.1.1. Message

6.3.1.1.1. Send voice message

SC system should allow user to send a voice message in a chat. The voice message is first recorded and sent out when the recording is done. SC system should allow the voice message to be sent in a chat with another user, or be sent in a group chat with several users.

1. SC system should allow user to activate voice recording function by tapping “record a voice message” button.
2. SC system should let user speak out the message.
3. SC system should allow user to tap “stop recording and send” button to send the voice message in a chat with a friend or in a group chat.
4. SC system must process the voice message and store it in database and then send out to the recipient user .
    4.1. If SC system fails to process the recorded voice message or fails to store it in database, the system shall display an error message and suggests the user to record again.
    4.2. If SC system fails to send the voice message to the recipient user due to network problem, the system shall show error and potential reasons to the user explaining that it has encountered a network error.
5. SC system must display the new voice message on the chat page as a sent voice message if the sending is successful.

6.3.1.1.2. Listen to a voice message

SC system should allow the user to listen to a voice message sent/received by him/her.

1. SC system should allow the user to tap a specific voice message to play it.
2. System should play the voice message via native system speaker and at the same time shows the message is being played.
    2.1. SC system should allow the user to tap the voice message again to stop playing it.
        2.1.1. SC system should stop playing the message and stop showing that the voice message is being played.
    2.2. If the user taps on another voice message when there is a voice message being played, SC system shall stop playing the current voice message and play the selected voice message automatically.

6.3.1.1.3. View chat history

SC system should allow user to view voice message chat history with another user or with several users in a group chat. SC system should display past voice message in chronological order, with the most recent messages showing first.

1. SC system should allow user to visit the chat list page.
2. SC system should allow user to tap on a chat and enters the chat page.

    2.1.SC system should allow user to scroll down the chat list page to find the friend.
    2.2.Back to 2.
3. SC system should list out the most recent 10 voice messages in chronological order.
4. SC system should allow user to scroll down to view earlier chat history
    4.1.SC system should allow user to proceed to listen to a voice message that is listed in the chat history
    4.2.Back to 4.

6.3.1.2. Call

6.3.1.2.1. Voice call a friend

SC system should allow user to make voice call with a friend. SC system should allow user to initiate a voice call and wait for friend's response to start voice communication.

1. SC system should allow user to visit a chat page of a friend.
2. SC system should allow user to tap on the "voice call" button to initiate a voice call with a friend.
3. SC system should send the invitation to user's friend and display a waiting page to wait for the response.
4. SC system shall wait the invited friend to accept the voice call invitation by tapping the "accept" button.
    4.1.SC system shall receive declining response of the voice call request.
        4.1.1. SC system should terminate the voice call invitation session.
        4.1.2. SC system must notify the user that the request has been declined.
        4.1.3. SC system should return to the chat page.
    4.2.If SC system does not receive response for more than 120 seconds, the system should treat the the session as a timeout.
        4.2.1. SC system should terminate the voice call invitation session.
        4.2.2. SC system must notify the user about the timeout of the voice call invitation.
        4.2.3. SC system should return to the chat page.
5. System shall receive confirmation response and establish a voice communication channel between two users.
    5.1.If SC system encounters some low quality connectivity or disconnection for the voice call, the system should ends the voice call.
    5.2.SC system must notify the user about the disconnection of the voice call.
    5.3.SC system should direct the user to the chat page.
6. SC system should allow user to tap on a corresponding end button to end the voice call.

6.3.1.2.2. Video call a friend

SC system should allow user to to make video call with a friend. SC system should allow user to initiate a video call and wait for friend's response to start video communication.

1. SC system should allow user to visit a chat page of a friend.

2. SC system should allow user to tap on the “video call” button to initiate a video call with a friend.
3. SC system should send the invitation to user’s friend and displays a waiting page to wait for the response.
4. SC system must wait for the invited friend to accept the video call invitation by tapping the “accept” button.
5. SC system shall receive confirmation response and establish a video communication channel between two users.
    - 5.1. SC system shall receive declining response of the video call request.
        - 5.1.1. SC system should terminate the video call invitation session.
        - 5.1.2. SC system must notify the user that the request has been declined.
        - 5.1.3. SC system should direct the user to the chat page.
    - 5.2. If SC system does not receive response from the invited user for more than 120 seconds, the system should treat the session as a timeout.
        - 5.2.1. SC system should terminate the video call invitation session.
        - 5.2.2. SC system must notify the user about the timeout of the video call invitation.
        - 5.2.3. SC system should direct the user to the chat page.
6. SC system should allow the user to video chat with a friend.
    - 6.1. If SC system encounters some low quality connectivity or disconnection for the video call, SC system should end the video call session.
    - 6.2. SC system should terminate the video call.
    - 6.3. SC system must notify the user about the disconnection of the video call.
    - 6.4. SC system should direct the user to the chat page.
7. SC system should allow the user to end the video call by tapping on a corresponding end button.

6.3.2. Group chat

6.3.2.1. Create a new group chat

SC system should allow the user to create a group chat with two or more friends. SC system should create chat successfully only when the user adds two or more friends in the chat and a valid name is given.

1. SC system should allow the user to visit the chat list page.
2. SC system should allow the user to tap the “Create a new group chat” button to initiate creating a new group chat.
3. SC system must prompts form for user to input general information of the group chat.
    - 3.1. SC system should allow the user to select the cancel option.
    - 3.2. The system should return the user to the chat list page without the group chat being created and any information entered has been erased.
4. SC system should ask the user to input the name of the group chat.
    - 4.1. If the user does not input a group chat name, SC system should name the group chat using the usernames of users in this group chat.
5. SC system should allow the user to select at least 2 friends to be added in this group chat by scrolling down and up the friend list.

5.1. If the user only selects one friend to be added in the group chat, SC system should disable the submit button for creating the group.

6. SC system should allow the user to submit the form.
7. SC system should create new group chat with the given name and acknowledge the successful creation of the group by displaying the new group chat in the associated users' chat list pages.

6.3.2.2. Add friends to a group chat

SC system should allow the user to add one or more friends into an existing group chat. SC system should allow the friends being added to a group chat to receive messages from the group chat.

1. SC system should allow the user to visit the group chat page.
2. SC system should allow the user to tap on the "add friends" button to add new members to group chat.
3. SC system must display a list of user's friends for selection to be added to the group chat.
   3.1. SC system should allow the user to select the cancel option.
   3.2. The system should return the user to the group chat page without the new member(s) being added and any information entered has been erased.
4. SC system should allow the user to select one or more friends and confirms the action.
   4.1. SC system should allow the user to select the cancel option.
   4.2. The system should return the user to the group chat page without the new member(s) being added and any information entered has been erased.
5. SC system shall update the member list of this group chat.
   5.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
6. SC system should allow the new member to receive chat messages from the group chat and SC system should display this group chat in the new member's chat list when first chat message is received.
7. SC system should direct the user back to the group chat page and notify the joining of new members in the group chat.

6.3.2.3. Quit a group chat

SC system should allow the user to quit a group chat. SC system should delete the group chat from the user's chat list and the system will no longer send chat messages from this group chat to the user.

1. SC system should allow the user to activate quitting a group chat by tapping on the "quit" button.
2. SC system should delete the user from the group chat member list.
   2.1. If after the user is deleted from a group chat member list and there are only 2 users left in a group chat, SC system shall keep the group chat although there are less than 3 members in the chat.

2.2. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.

3. SC system should delete the group chat from the user’s chat list.
4. SC system should return the user back to the chat list page.
5. SC system should delete the user from the group chat member list.

6.4. Moments

6.4.1. Share

6.4.1.1. Share photo(s) in moments

SC system should allow the user to share photo in moments. SC system should allow the user to upload a photo from mobile phone photo library or take a picture via mobile phone camera. SC system should allow up to 9 photos in every moment.

1. SC system should allow the user to tap the “New Photo” button located at the top of “moments” page.
2. System should prompt out a dialog for user to choose either uploading from mobile phone photo library or taking a new photo via mobile phone camera.
   2.1. SC system should allow the user to select the cancel option.
      2.1.1. Back to 9.
3. SC system should allow the user to choose to take a new photo.
   3.1. SC system should allow the user to choose to upload existing photo(s) from mobile phone photo library.
      3.1.1. System should display a gallery of existing photos in mobile phone for user to choose.
      3.1.2. SC system should allow the user to taps photo(s) which he/she intends to upload. SC system should allow the user to choose up to nine photos in a moment.
      3.1.3. SC system should allow the user to tap corresponding button to confirm the selection.
      3.1.4. Back to 6.
   3.2. SC system should allow the user to select the cancel option.
      3.2.1. Back to 9.
4. System should prompt the interface for taking photo provided by the native mobile phone system.
   4.1. SC system should allow the user to select the cancel option.
      4.1.1. Back to 9.
5. SC system should allow the user to tap corresponding button to take a new photo.
   5.1. SC system should allow the user to select the cancel option.
      5.1.1. Back to 9.
6. System should prompt a page where photo(s) already selected or taken is/are displayed. System should also provide a “+” sign, by taping which system shall allow user to add up to 9 photos in a moment.
   6.1. SC system should allow the user to tap “+” sign to choose more photos from mobile phone photo library for uploading.
      6.1.1. Back to 3.1.
   6.2. SC system should allow the user to select the cancel option.
      6.2.1. Back to 9.

7. SC system should allow the user to tap “upload” button to share all photos as one moment.
8. System should prompt a dialog informing the user that the upload is successful.
9. System should direct user back to main page of “moments” to view the uploaded photo(s).

6.4.1.2. View moments shared by friends

SC system should allow the user to view photos shared by friends in “moments”.

1. SC system should allow the user to tap the “moments” button from navigation bar.
2. System should prompt out a list of moments shared by friends sorted by posting time.
   2.1. If the user and none of the user’s friends have shared any photo, SC system should display “No moments, Start sharing” in the page.
3. In each moment, SC system should display the user’s icon is at the top left corner, followed by the user’s name at top. SC system should display user’s photo(s) below.
4. Following every moment, SC system should allow the user to tap a big button to like the moment if he/she has not liked it.
   4.1. If the user has already liked the moment, the system should provide user a button to unlike the moment.
   4.2. Back to 5.
5. SC system should allow the user to tap a specific photo to view more details.
6. SC system should allow the user to tap the current viewing photo to exit detailed mode.
   6.1. SC system should allow the user to swipe right or left to view next or earlier photo(s) correspondingly.
   6.2. Back to 6.
7. SC system should allow the user to scroll down to view moments shared by friends earlier.

6.4.2. Like

6.4.2.1. Like moments shared by friends

SC system should allow the user to like photo(s) shared by friends in “moments” page.

1. SC system should allow the user to tap “like” button under a certain moment.
2. System should change the “like” button under the moment to “unlike” button.
   2.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.

6.4.2.2. Unlike moments shared by friends

SC system should allow the user to unlike photo(s) shared by friends in “moments” page.

1. SC system should allow the user to tap “unlike” button under a certain moment.
2. System should change the “unlike” button under the moment to “like” button.

2.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.

6.5. Community and Events

6.5.1. Community

6.5.1.1. View community list

SC system should allow the user to view list of communities available in SC system. SC system should display communities user have joined as a member first followed by other communities. SC system should list all communities in alphabetic order.

1. SC system should allow the user to tap "community" button in navigation bar to enter community list page.
2. SC system should display list of communities available in SC system. SC system displays communities user have joined as a member first followed by other communities. SC system should list all communities in alphabetic order.
   2.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
3. SC system should allow the user to scroll down to view more communities.

6.5.1.2. Join new community

SC system should allow the user to join a new community. SC system should only allow the user to join communities available on the community list that they are not a member of. SC system should allow communities to send event updates to the user after he/she joining the community.

1. SC system should allow the user to tap "community" button in navigation bar to enter community list page.
2. SC system should allow the user to tap corresponding community avatar in community list page to enter a community.
   2.1. If the user is unable to find the community avatar, SC system should allow the user to scroll down to view more communities.
   2.2. Back to 2.
3. SC system should allow the user to tap "join community" button on the top of the community page that he/she wants to join.
4. SC system should add the community to user's community list in database and allows the user to receive event update of it. SC system should allow the user to join events created and managed by the community.
   4.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
5. SC system should show the newly joined community in front of other communities in community list.

6.5.1.3. Quit a community

SC system should allow the user to join a new community. SC system should only allow the user to join communities available on the community list that they are not a member of. SC system should allow communities to send event updates to the user after he/she joining the community.

1. SC system should allow the user to tap "community" button in navigation bar to enter community list page.
2. SC system should allow the user to tap corresponding community avatar in community list page to enter a community.
   2.1. If the user is unable to find the community avatar, SC system should allow the user to scroll down to view more communities.
   2.2. Back to 2.
3. SC system should allow the user to tap "quit community" button on the top of the community page that he/she wants to quit.
4. SC system should remove the community from user's community list in database and SC system should disallow the user to receive event update of it.
   4.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
5. SC system should remove the user from all upcoming events of the quit community by updating the corresponding database tables. SC system should also remove all upcoming events from the user's calendar of joined events. SC system should not allow the user is to join events created and managed by the community.
   5.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
6. SC system should remove the newly quit community from the joined community list and show it in alphabetic order together with other disjoined communities.

6.5.2. Events

6.5.2.1. Receive community event notification

SC system should allow the user to receive information about new published event by any joined communities in SC system.

1. SC System should display a red dot on top right corner of the "community" button in navigation bar.
2. SC system should allow the user to tap "community" button to enter community list page.
3. SC system should display list of communities available in SC system. SC system should display communities user have joined as a member first followed by other communities. SC system should list all communities in alphabetic order.
4. If there is any event update in the communities user have joined, SC system shall display a red dot at the bottom of the group name.

5. SC system should allow the user to tap corresponding community avatar in community list page to enter a community page.
6. SC system should remove the red dot of communities viewed by the user.
7. SC system should remove the red dot on top right corner of the community button in the navigation bar if no community has unread event updates.
    7.1. SC system should keep the red dot on top right corner of the community button in the navigation bar if users exits community page before finishes reading all event updates.
8. SC system should allow the user to scroll down to view more communities.

6.5.2.2. View community event list

SC system should allow the user to view all events organized and published by the communities in SC system. SC system should only show events of date no later than current date in chronological order by default. SC system should allow the user to scroll upward to view earlier events of selected community.

1. SC system should allow the user to tap “community” button to enter community list page.
    1.1. If the user is currently at the home page of one community. SC system should allow the user to tap “back” button at the top left corner of the community home page to exit to community list page.
    1.2. Back to 2.
2. SC system should display list of communities available in SC system. SC system should display communities user have joined as a member first followed by other communities. SC system should list all communities in alphabetic order.
3. If there is any event update in the communities user have joined, SC system shall display a red dot at the bottom of the group name.
4. SC system should allow the user to tap corresponding community avatar in community list page to enter a community page.
    4.1. SC system should display “no upcoming events” in community home page if there is no published event of date no later than current date.
    4.2. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
5. SC system should display all events of date no later than current date in chronological order of the selected community.
6. SC system should allow the user to scroll upward to view earlier events organized and published by the selected community.

6.5.2.3. Join an event

SC system should allow the user to join an event organized and published by joined communities. SC system should not allow the user to to join events of disjoined communities.

1. SC system should allow the user to tap community button in navigation bar to enter community list page.

2. SC system should display list of communities available in SC system. SC system should display communities user have joined as a member first followed by other communities. SC system should list all communities in alphabetic order.
3. SC system should allow the user to tap corresponding community avatar in community list page to enter a community page.
4. SC system should display all events of date no later than current date in chronological order of the selected community.
   - 4.1. SC system should allow the user to scroll upward to view earlier events organized and published by the selected community.
     - 4.1.1. SC system should allow the user to tap the grey “ended” button under events of date earlier than current date.
     - 4.1.2. SC system must pop out a message box stating “event has already ended”.
     - 4.1.3. Back to 5.
   - 4.2. SC system should display “no upcoming events” in community home page if there is no published event of date no later than current date.
5. SC system should allow the user to view all events and select the one he/she wants to join by tapping the “click to join” button under the event details.
   - 5.1. If the user is not a member of the selected community, SC system must pop out a message box stating “Unable to join member exclusive community events. Please join the community first!”
6. SC system should change the green “click to join” button into red “joined, click to drop” button.
   - 6.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
7. SC system should register the user as a participant of the selected event by updating any data associated with the event and the user in database.
8. SC system should add the events into the user’s calendar of joined events list on the corresponding date.

6.5.2.4. Drop an event

SC system should allow the user to drop an event organized and published by joined communities. SC system should not allow the user to to drop events he/she has joined before before the expiration date.

1. SC system should allow the user to tap community button in navigation bar to enter community list page.
9. SC system should display list of communities available in SC system. SC system should display communities user have joined as a member first followed by other communities. SC system should list all communities in alphabetic order.
2. SC system should allow the user to tap corresponding community avatar in community list page to enter a community page.
3. SC system should display all events of date no later than current date in chronological order of the selected community. For every event the user has joined and has not expired, SC system must display a red “joined, click to drop” button.

4. SC system should allow the user to view all events and select the one he/she wants to drop by tapping the “joined, click to drop” button under the event details.
   4.1. SC system should allow the user to scroll upward to view earlier events organized and published by the selected community.
      4.1.1. SC system should allow the user to tap the grey “ended” button under events of date earlier than current date.
      4.1.2. SC system must pop out a message box stating “event has already ended”.
      4.1.3. Back to 5.
   4.2. SC system should display “no upcoming events” in community home page if there is no published event of date no later than current date.
5. SC system should change the red “joined, click to drop” button to green “click to join”.
6. SC system should remove the user from participants list of the selected event by updating or removing any data associated with the event and the user in database.
   6.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.
7. SC system should remove the events from the user’s calendar of joined events list on the corresponding date.

6.5.2.5. View joined event

SC system should allow the user to view all events he/she has joined. SC system should display events of the current month in a calendar in chronological order by default. SC system should highlight current date. SC system should allow the user to view joined events sorted according to year, month, week and day.

1. SC system should allow the user to tap “calendar” button in navigation bar.
2. SC system should display events of current month that the user has joined in chronological order. SC system should highlight current date.
3. SC system should allow the user to select “week” option on top of the calendar.
   3.1. SC system should allow the user to select “day” option on top of the calendar.
      3.1.1. SC system should display events of current day that the user has joined in chronological order. SC system should highlight the current time.
   3.2. SC system should allow the user to select “year” option on top of the calendar.
      3.2.1. SC system should display events of current year that the user has joined in chronological order. SC system should highlight the current year.
   3.3. SC system should allow the user to select “month” option on top of the calendar.

        3.3.1. SC system should display events of current month that the user has joined in chronological order. SC system should highlight the current month.

    3.4. SC system should allow the user to select “next” or “former” option on top of the calendar.

        3.4.1. SC system should display events of next or former month that the user has joined in chronological order.

4. SC system should display events of current week that the user has joined in chronological order.

6.6. Friend

6.6.1. View list of friends

SC system should allow the user to view his/her friend list. SC system should list friends in alphabetic order by default. SC system should allow the user to scroll down to view more friends.

1. SC system should allow the user to tap contact list button in navigation bar.
2. SC system should display the user’s friends listed in alphabetic.
    2.1. SC system should display “no friend” in friend list page if there is no friend of the user.
3. SC system should allow the user to scroll down to view more friends.

6.6.2. Send a friend request

SC system should allow the user to send friend request to people in contact list of his/her native mobile phone.

1. SC system should allow the user to tap friend list button in navigation bar.
2. SC system should allow the user to tap “add friend” button on the top of the friend list page.
3. SC system must fetch contact list from the user’s native mobile phone and displays names in alphabetic order. SC system should display a green “add” button following people’s name on the right end of each row.
4. SC system should allow the user to send friend request to a friend he/she wants to add by tapping the “add” button.
5. SC system should send a friend request to the corresponding user.
    5.1. If the user has already added the target user into friend list, SC system should direct the user to chat page with the corresponding user.
6. SC system should allow the target user receives a friend request upon which he/she can perform accept or decline action.

6.6.3. Accept a friend request

SC system should allow the user to accept a received friend request and add the user initiated the friend request into his/her friend list. The SC system should also add current user into friend list of the user who initiated the friend request.

1. SC system should allow the user to tap “friend request” button on top of chat list page to view received friend requests.

2. SC system should show a list of friends requests. System should display a green "accept" button and a red "decline" button on the right hand of every friend request.

   2.1. SC system should display no friend request if there is no unsolved friend request.

3. SC system should allow the user to tap "accept" button to accept friend request.
4. SC system should remove the selected request from list and add the user initiated the friend request and the user accepted the friend request into each other's friend list.

   4.1. If the system is unable to process the request due to network problem, the system should show error and potential reasons to the user explaining that it has encountered a network error.

5. SC system should direct the user to chat page of the newly added friend.

6.6.4. Decline a friend request

SC system should allow the user to decline a received friend request and add the user initiated the friend request into his/her friend list. The SC system should also add current user into friend list of the user who initiated the friend request.

1. SC system should allow the user to tap "friend request" button on top of chat list page to view received friend requests.
2. SC system should show a list of friends requests. System should display a green "accept" button and a red "decline" button on the right hand of every friend request.

   2.1. SC system should display no friend request if there is no unsolved friend request.

3. SC system should allow the user to tap "decline" button to accept friend request.
4. SC system should remove the selected request from list and add the user initiated the friend request and the user accepted the friend request into each other's friend list.

   4.2. If the system is unable to process the request due to network problem, the system should show error and potential reasons to the user explaining that it has encountered a network error.

5. SC system should direct the user to chat page of the newly added friend.

6.7. Profile

6.7.1. View self profile

SC system should allow the user to to view his/her profile image, name, phone number, statistical analysis of daily active hour since registered.

1. SC system should allow the user to tap corresponding button to view his/her self profile.
2. System should prompt out a page for user to view his/her profile image, name, phone number, and statistical analysis of daily active hour since registered.
3. SC system should allow the user to tap a specific date in the statistic diagram to view the number of active hours of selected date.

3.1. If the system is unable to process the request due to network problem, the system must show error and potential reasons to the user explaining that it has encountered a network error.

4. System should show the statistic diagram for active hours.

6.7.2. View own photos

SC system should allow the user to to view his/her own shared photos.

1. SC system should allow the user to tap corresponding button to view own photos.
2. System should prompt out the shared photos sorted by date.
3. SC system should allow the user to scroll down to view photos shared earlier.
   - 3.1. If the system is unable to process the request due to network problem, the system should display no photo. The system should also show error and potential reasons to the user explaining that it has encountered a network error.
4. (Optional) SC system should allow the user to tap a specific photo to view more details.
   - 4.1. SC system should allow the user to swipe right to view next photo.
     - 4.1.1. Back to 3.3.
   - 4.2. SC system should allow the user to tap the current viewing photo to exit detailed mode.

7. Non-functional Requirements

7.1. Performance

- When activated, the system should be completely available within 15 seconds.
- The system shall be able to register user within 5 seconds after user authenticates his/her phone number.
- The system shall be able to send the secret code to user within 1 minute provided that the phone number is valid and not registered.
- The system shall be able to log the user in within 30 seconds provided that the input matches the generated secret code.
- The system shall be able to log the user in within 3 seconds provided that the user phone number matches the password.
- The successfully sent messages shall display within 1 second after the user taps “stop recording and send” button.
- If the system fails to process/send the recorded message, an error message shall display within 10 seconds.
- The chat page and history shall display within 1 second.
- The previous chat history shall display within 1 seconds after the user scrolls down to view
- The system shall be able to start playing a voice message within 1 second after the user taps to play the voice message.
- The voice call connection shall be established within 5 seconds after receive the confirmation response.
- The video call connection shall be established within 10 seconds after receiving the confirmation response.
- The system shall direct the user back to the chat page within 1 second if the voice/video call request is declined by the invited user, or due to poor network connection.
- During video chat, each user is able to see a real-time video of the other user. The latency of video is within 500ms.
- The system shall display the new group chat in the chat list page within 1 second after creation.
- After adding a friend to a group chat, the system shall return to the group chat page and send notifications within 3 seconds.
- After quitting a group, SC system shall return to the chat list page within 1 second.
- The system shall be able to upload photos at a speed around 500KB/s, given that the user’s mobile network supports such upload speed.
- The system shall be able to provide the profile without internet connection.
- The system shall be able to show the most recent 10 moments within 3 seconds.
- The request to like/unlike a moment should be completed within 1 second.
- The system shall be able to provide user profile and statistical diagram of the user’s usage rate within 3 seconds.
- The system shall be able to provide the number of active hours of selected date within 1 second.

- The system shall be able to log the user out within 3 seconds.
- The system shall display the list of communities within 3 seconds.
- The system should display list of events organized by the community in less than 3 seconds.
- The user should be able to receive event update in less than 3 seconds after the new event is published.
- The system should display the list of upcoming events of selected community in chronological order in less than 5 seconds.

7.2. Security

- The login information shall include username and password. The password consists of combination of at least 6 alphabets, numbers or special characters.
- The system shall be able to encrypt user's credentials and save encrypted data into database
- The system shall protect user's profile information from unauthorized access to prevent the leakage of user's privacy.
- Only the professional system administrators shall be authorized to complete community management operations. Eg: creating new community, publish new event details, etc.

7.3. Usability

- 80% of first-time users must be able to add friends, start a chat with a friend and send a voice message within 2 minutes of starting out.
- 80% of first-time users shall be able to join new community, view community event list and join an event within 5 minutes.
- The system should follow normal flow of activities.
- The application must function well both on Android and IOS mobile platforms.

7.4. Consistency

- In case of potential loss of connection between the client and the server, any clients' information or progress is lost.
- The system shall be able to verify the input information before updating it into database, e.g. unique and valid phone number, complex password.

7.5. Reliability

- The downtime of system due to yearly maintenance and contingency maintenance should be less than one day.
- The system must be able to run continuously for 24 hours a day and 7 days a week.
- System must go through regular security check to protect the private information of the users.
- The system should reboot automatically in the event of failure within 1 minute.

7.6. Correctness

- The user shall receive latest and accurate information of all events of joined communities.
- SC system shall update the calendar correctly when user joins/drops an event or when user quits a community.
- The statistical analysis shown at profile page shall include detailed information and statistics, such as date and number of active hours/day.

7.7. Maintainability

- The system server shall be capable of backing up the data.
- The SC system shall keep a log of all the errors.
- The system should be operating at a low maintenance level – yearly maintenance. The occurrence of situations where contingency maintenance is needed should be less than 2 times per 10,000 requests processed by the server.
- The downtime of system due to yearly maintenance and contingency maintenance should be less than one day.

7.8. Packaging

- The system shall be published in Google Play and Apple App stores, which will be installed automatically when user choose to install it.

8. Interface Requirements

The designed user-friendly Graphical User Interface should fulfil Shneiderman's 8 golden rules of interface design.

8.1. Strive for Consistency

The GUI should strive for consistency. For similar situations, for example, user join and drop an event, there should be consistent sequence of actions. For example, Tap “community” button → Choose a Community → View event list of the Community → Join/Drop an Event. There should also be consistent visual layout, such as fonts, button colors, backgrounds and icon styles.

8.2. Universal Usability

The GUI should cater to universal usability. The designed SC system should be able to function correctly in both Android and IOS mobile platforms. Plastic design should be executed, so that the future developed software application could adapt to various display resolutions all around the world. Plus, the adopted GUI should support novice users by displaying all main functions directly after logging into the application.

8.3. Offer Informative Feedback

The GUI should offer informative feedback for every user action. For instance, the color of buttons should change accordingly when user joins/drops an event. The type of feedback should vary according to the frequency of tasks. For common tasks, like receiving community event notification, modest feedback should be provided. However, for uncommon errors, like network connection failure, substantial feedback should be provided.

8.4. Permit Easy Reversal of Actions

The GUI should permit easy reversal of actions. For example, if the user joins/drops an event unintentionally, he should be able to cancel it.

8.5. Support Internal Locus of Control

The GUI should support internal locus of control. There should be rapid response of interface to ensure that the elderly won’t be annoyed by the system. Acausality should be avoided, making users initiators rather than responders. Users should obtain necessary ability easily. There should not be any tedious, unexpected or irrelevant actions so as to relax users and build satisfaction.

8.6. Reduce Short-term Memory Load

The GUI should reduce users’ short-term memory load by keeping display simple and consolidating display. The GUI should be consistent, thus making the software easy to be understood. It should also be in accord with the rule of thumb meaning 7+/-2 actions should be required for the completion of each common task.

8.7.Prevent Errors

The GUI should prevent errors as much as possible. For example, input should be by voice message instead of typing, reducing the chance for errors when typing. In case of errors, there should be simple, constructive and specific messages to help fix current error and reduce similar errors, for example the error message when failing to connect to the server. Information, screens and tabs should also be organized in a logical and consistent way with distinctive buttons and commands. Large buttons should also be implemented to help to restrain the error rate.

9. Input Requirements

The system requires the following kinds of data to be functional.

9.1. User Identification

A user must be able to identify by others and the system. In the proposed system, hand phone number is considered as the identification for user.

9.2. User Generated Content

The content generated by users is the core of this social network application. It includes users' chat message, moments, and other action records that will be available to others.

9.3. User Preferences

The application should be customized based on user's preference, such as font size, interface color scheme, and notification setting.

10. Process Requirements

The following are among the inherent requirements that the SC system must be able to handle.

10.1. Data Validation

Data error from the user's end and from the back-end database-processing end must be gracefully handled. There will be data validation and error-handling routines as part of the SC system.

10.2. Performance

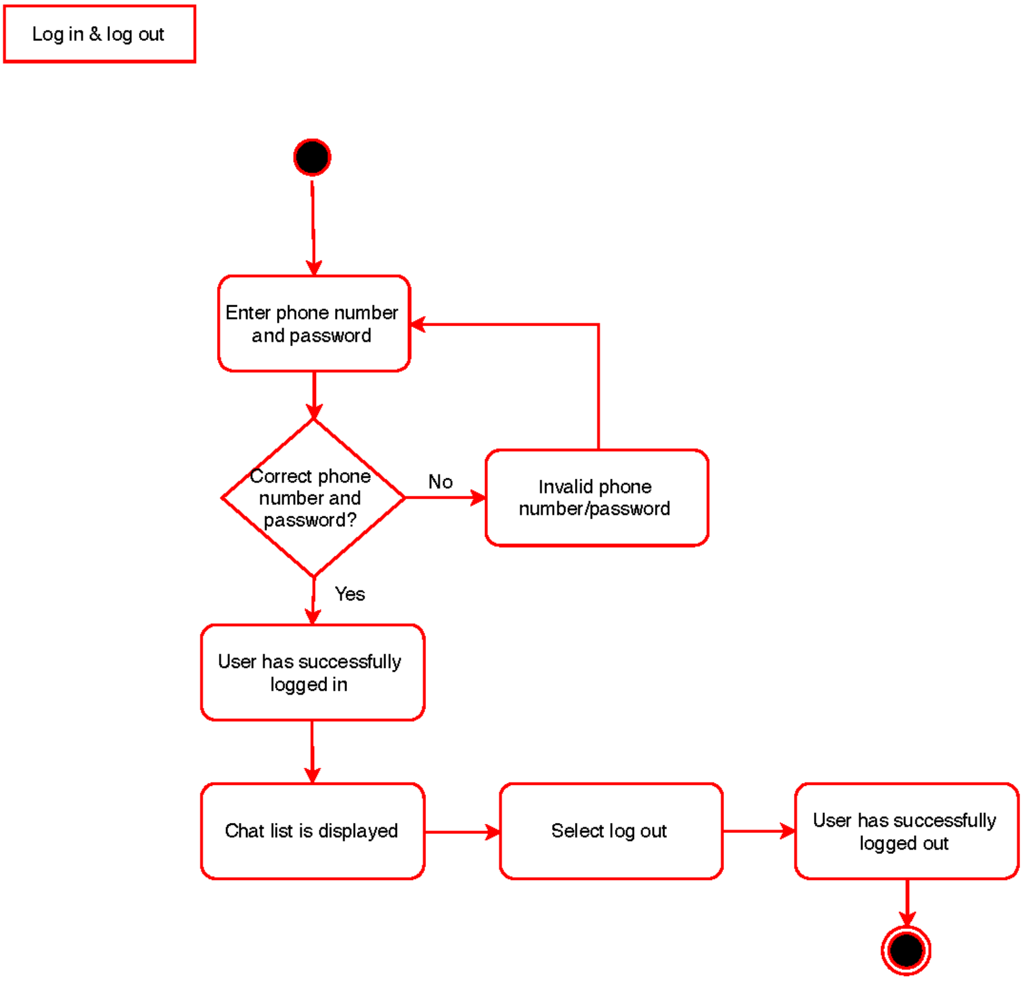
The system must be able to run continuously for 24 hours a day and 7 days a week.

System must go through regular security check to protect the private information of the users.

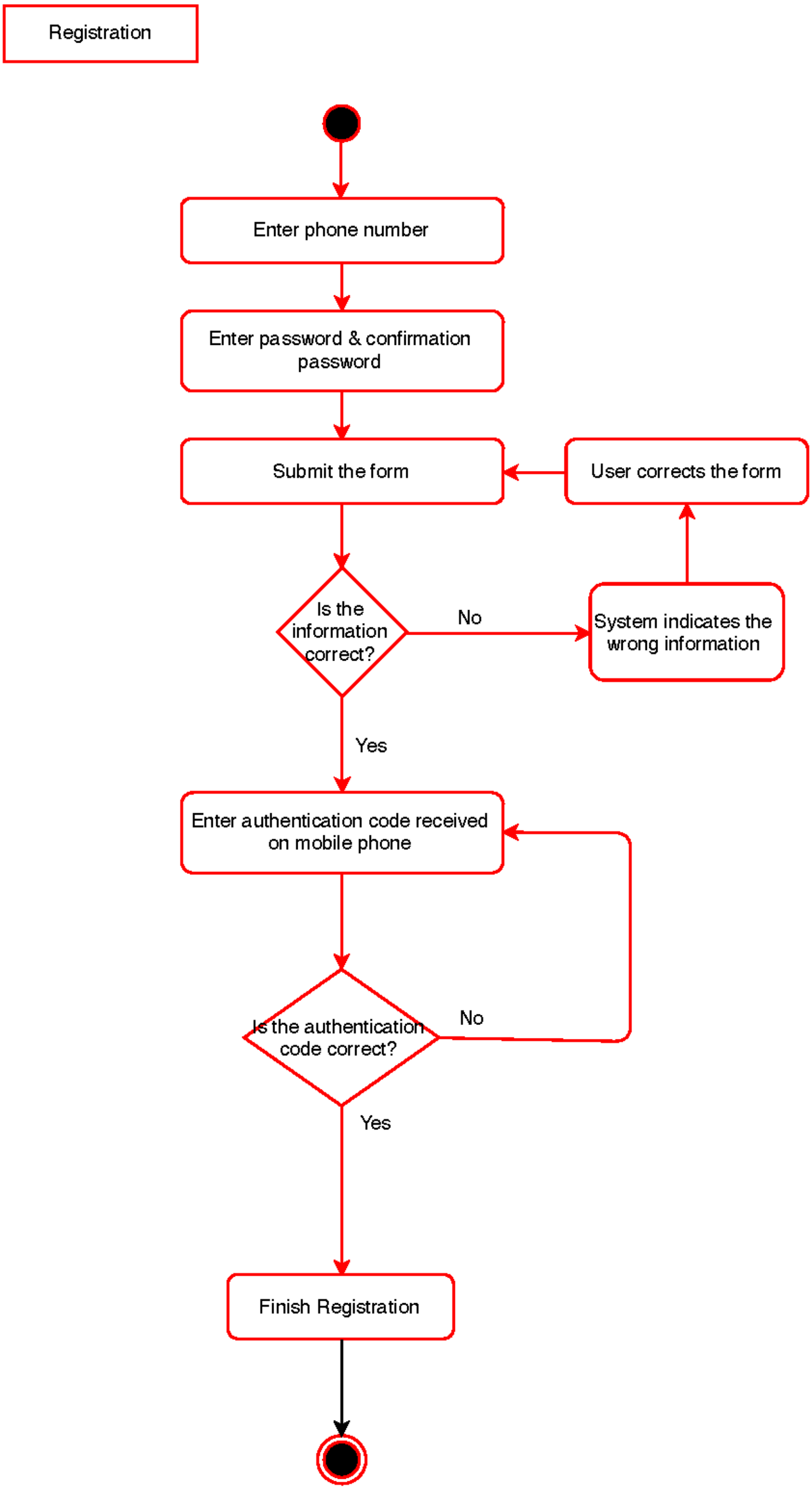
10.3. Activity Diagram

The following activity diagrams depicts the process break down in details. They work as good complements to use cases as they provide a visual picture of the steps SC system undertakes to deliver outcomes and the procedural logic required in the process. The focus of activity diagrams is the action sequence of execution and the conditions that trigger or guard those actions.

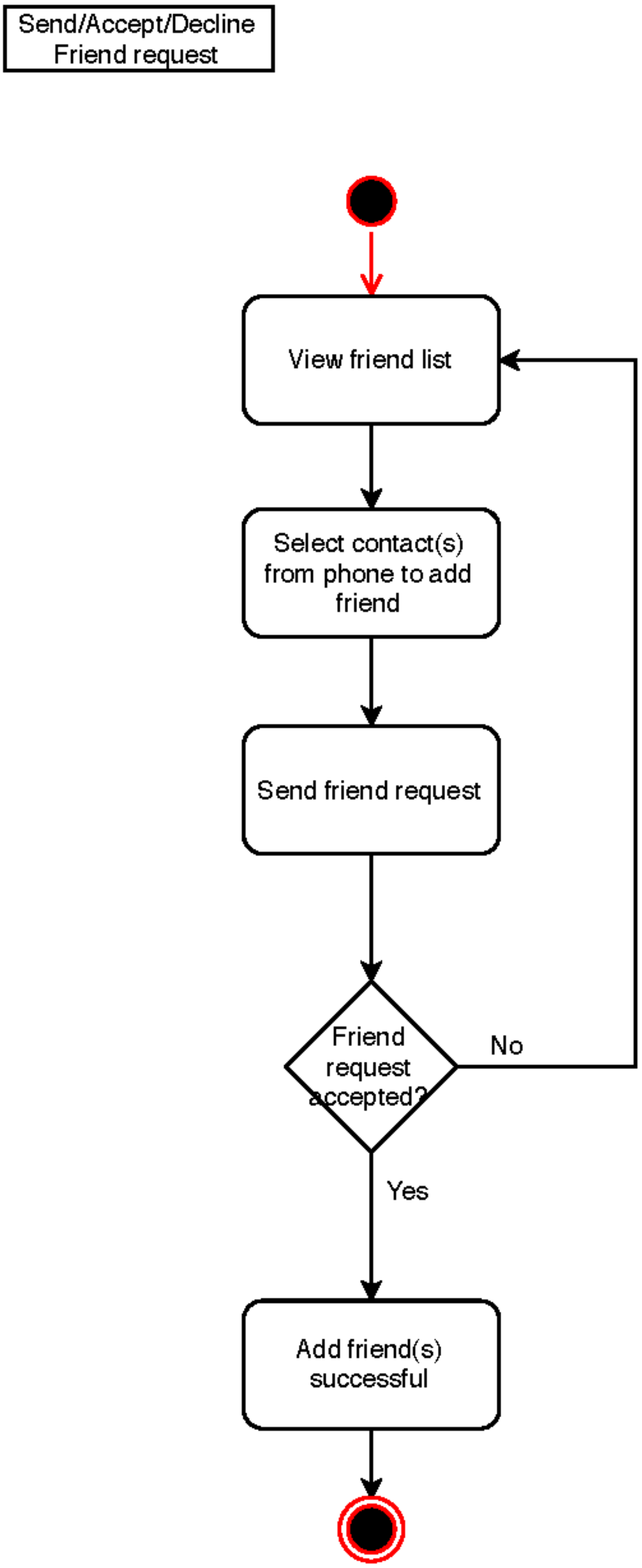
10.3.1. Login/Logout


Log in & log out
Enter phone number and password
Correct phone number and password?
No
Invalid phone number/password
Yes
User has successfully logged in
Chat list is displayed
Select log out
User has successfully logged out

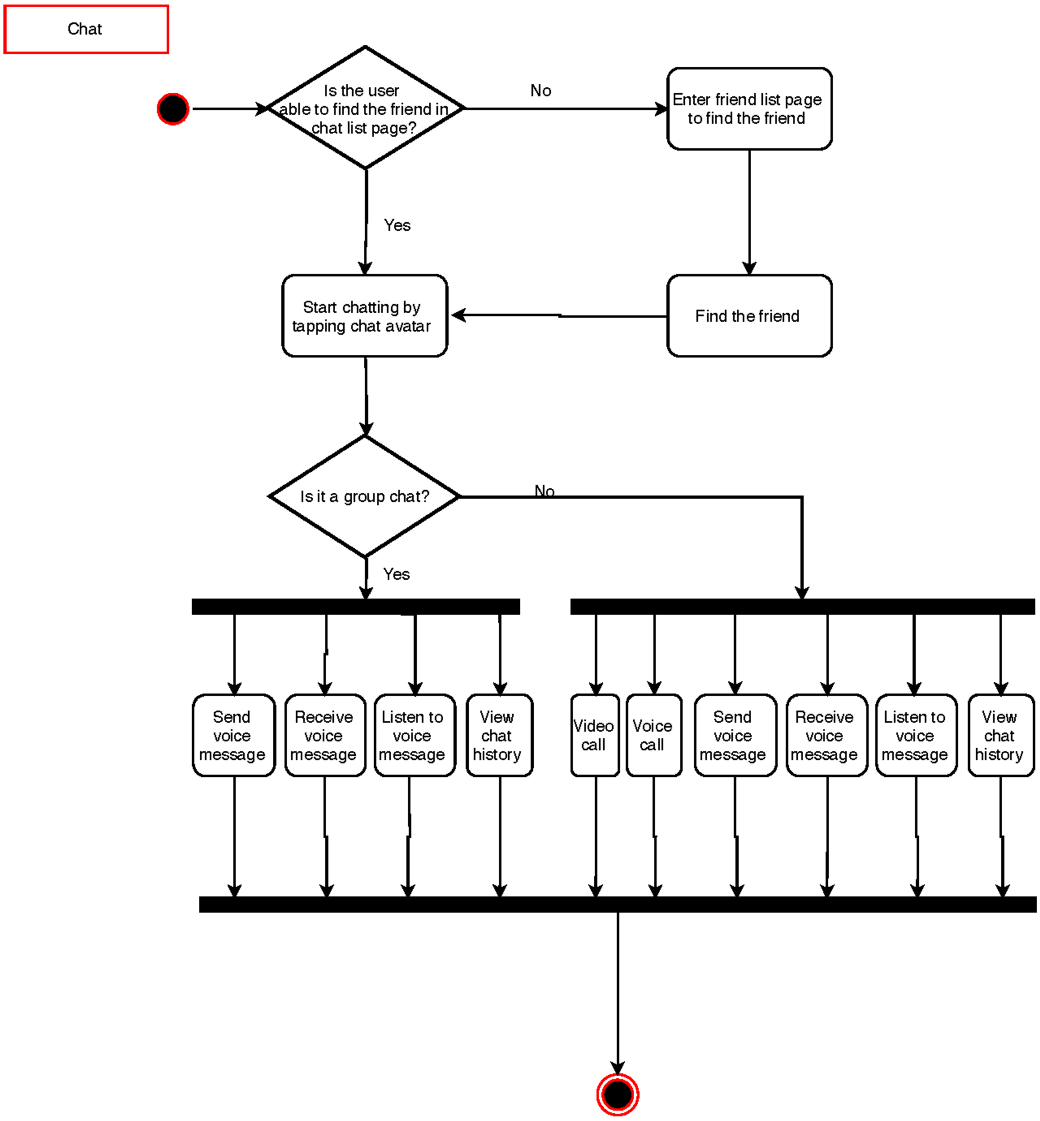
10.3.2. User Registration


Registration
Enter phone number
Enter password & confirmation password
Submit the form
User corrects the form
Is the information correct?
No
System indicates the wrong information
Yes
Enter authentication code received on mobile phone
Is the authentication code correct?
No
Yes
Finish Registration

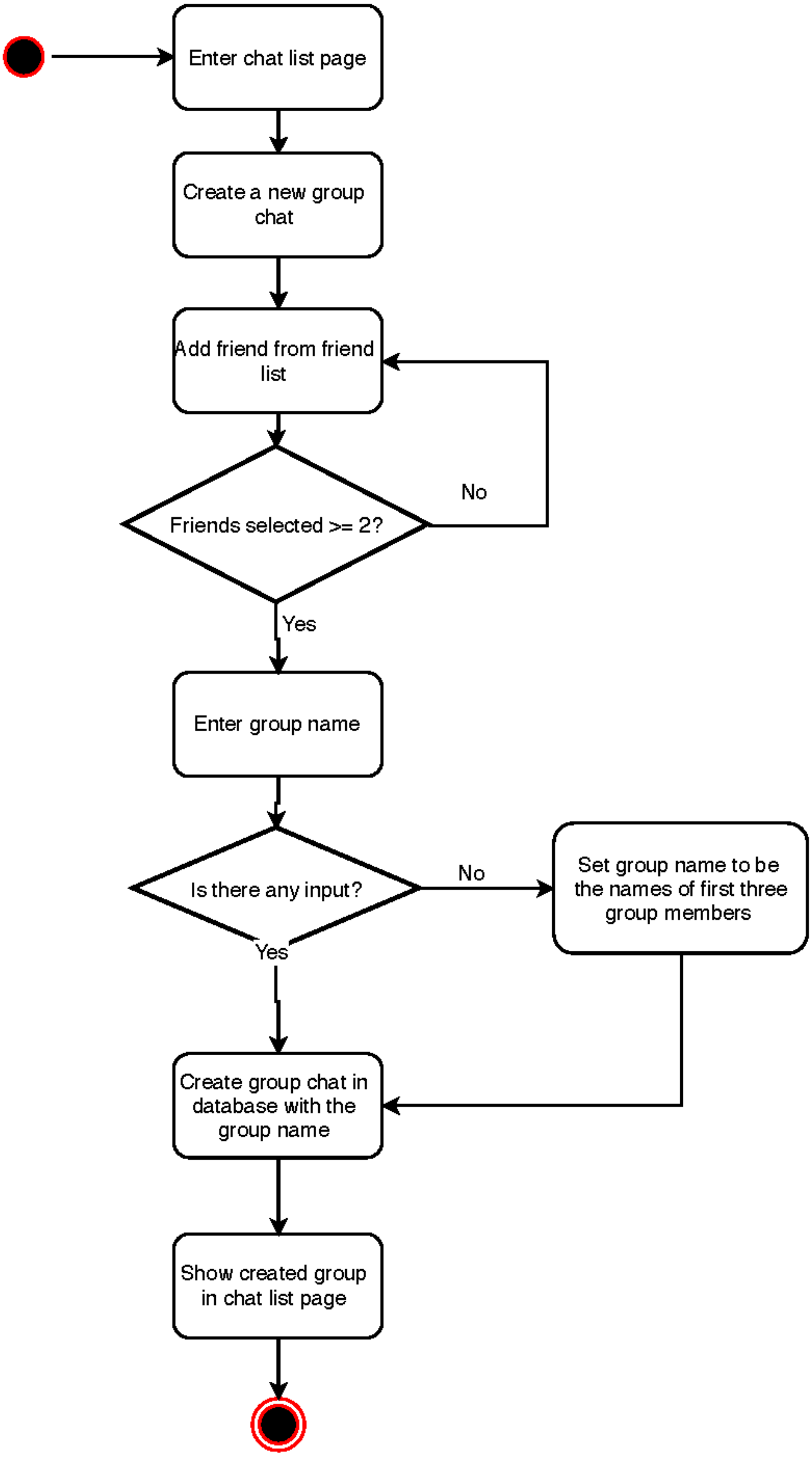
10.3.3. Send/Accept/Decline friend request



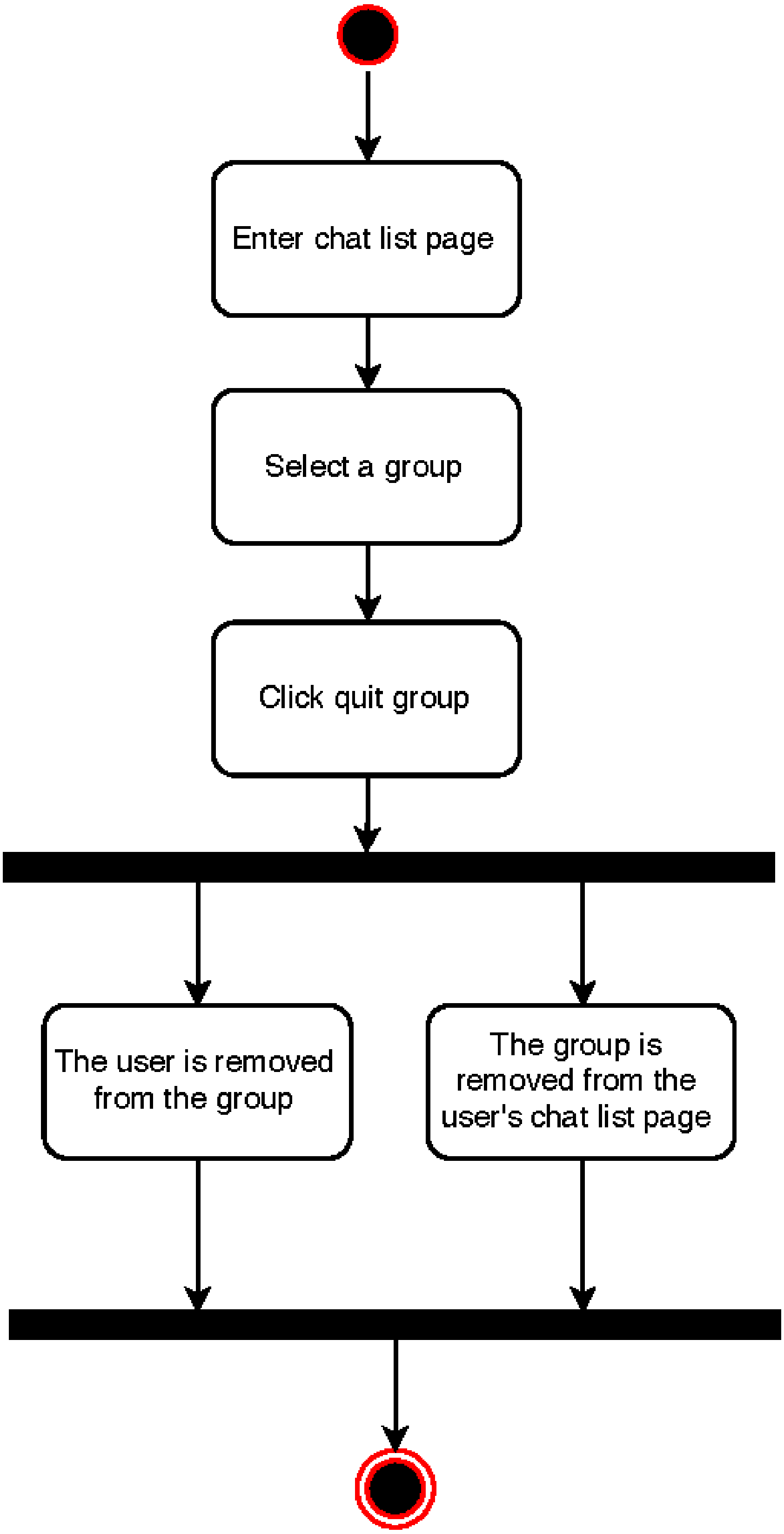
10.3.4. Chat


Chat
Is the user able to find the friend in chat list page?
No
Enter friend list page to find the friend
Yes
Start chatting by tapping chat avatar
Find the friend
Is it a group chat?
No
Yes
Send voice message
Receive voice message
Listen to voice message
View chat history
Video call
Voice call
Send voice message
Receive voice message
Listen to voice message
View chat history

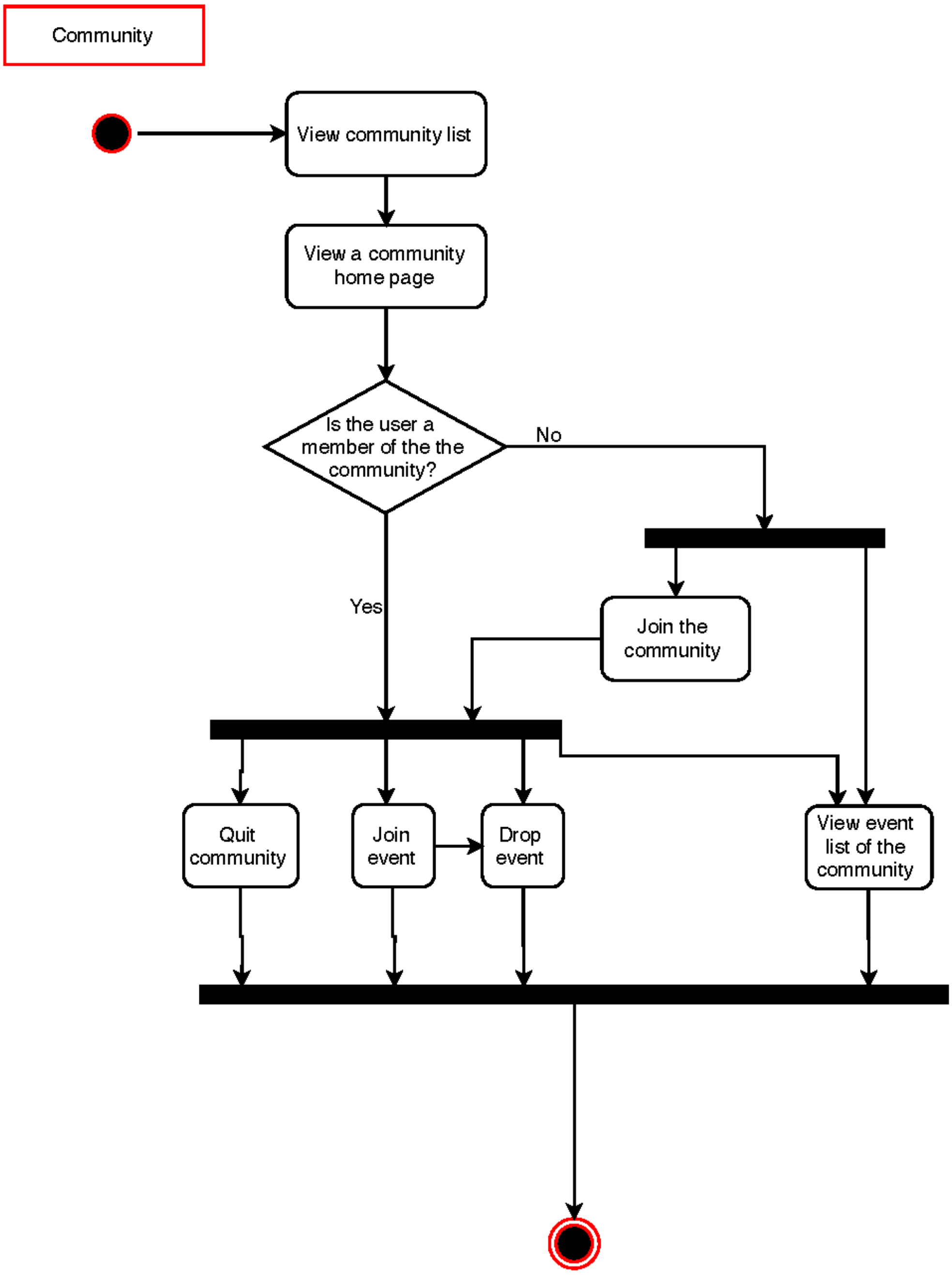
10.3.5. Create a group chat


Enter chat list page
Create a new group chat
Add friend from friend list
Friends selected >= 2?
No
Yes
Enter group name
Is there any input?
No
Set group name to be the names of first three group members
Yes
Create group chat in database with the group name
Show created group in chat list page

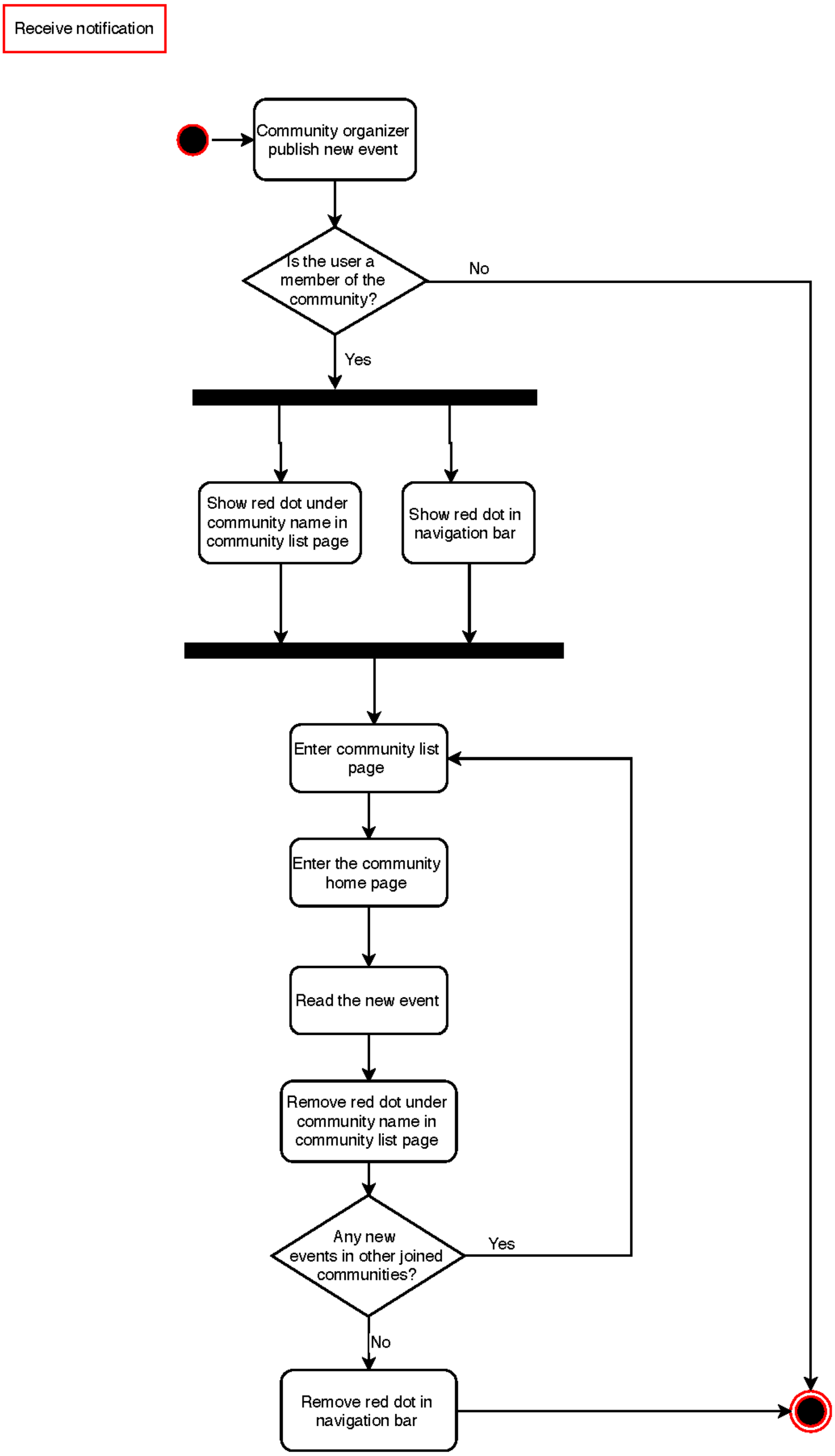
10.3.6. Quit a group chat


Enter chat list page
Select a group
Click quit group
The user is removed from the group
The group is removed from the user's chat list page

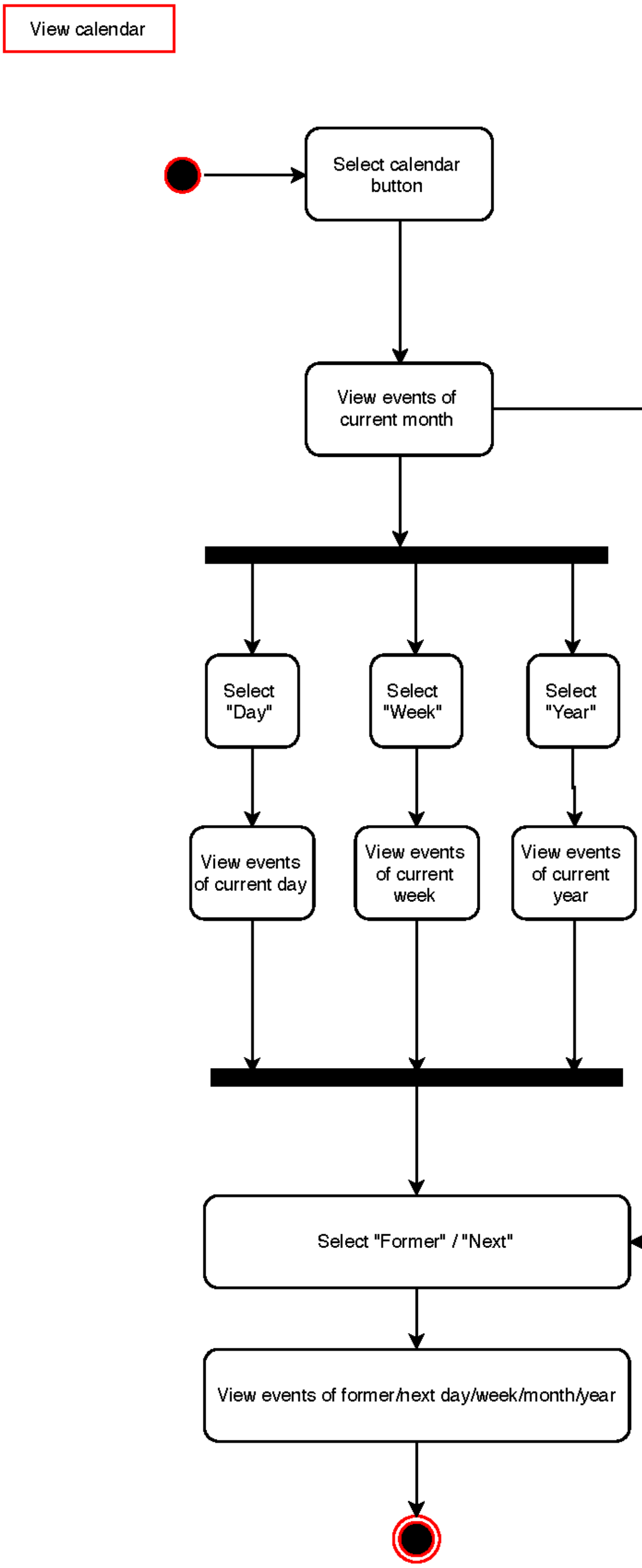
10.3.7. Join/View/Quit community & Join/Drop event


Community
View community list
View a community home page
Is the user a member of the the community?
No
Yes
Join the community
Quit community
Join event
Drop event
View event list of the community

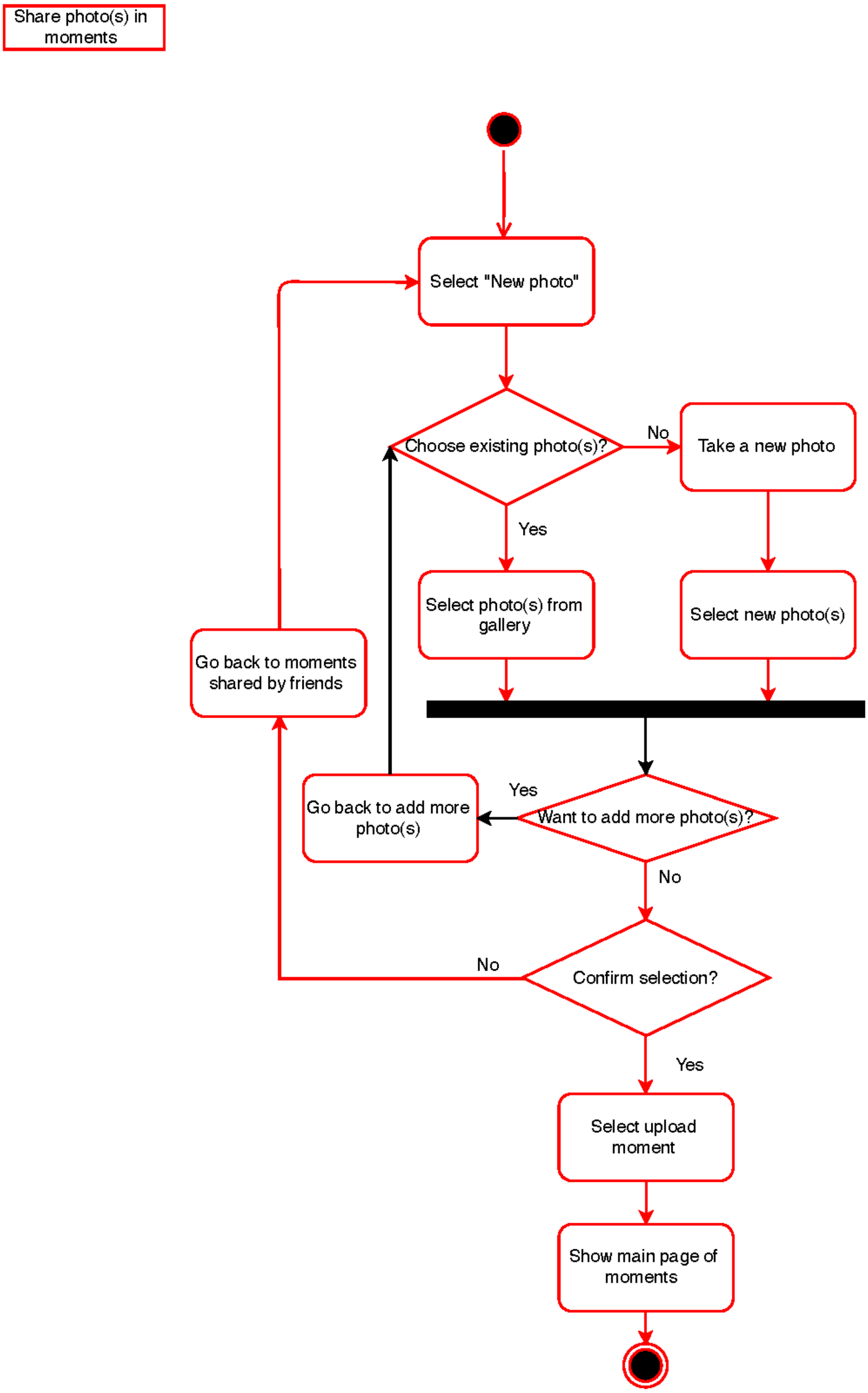
10.3.8. Receive event notification of joined communities


Receive notification
Community organizer publish new event
Is the user a member of the community?
No
Yes
Show red dot under community name in community list page
Show red dot in navigation bar
Enter community list page
Enter the community home page
Read the new event
Remove red dot under community name in community list page
Any new events in other joined communities?
Yes
No
Remove red dot in navigation bar

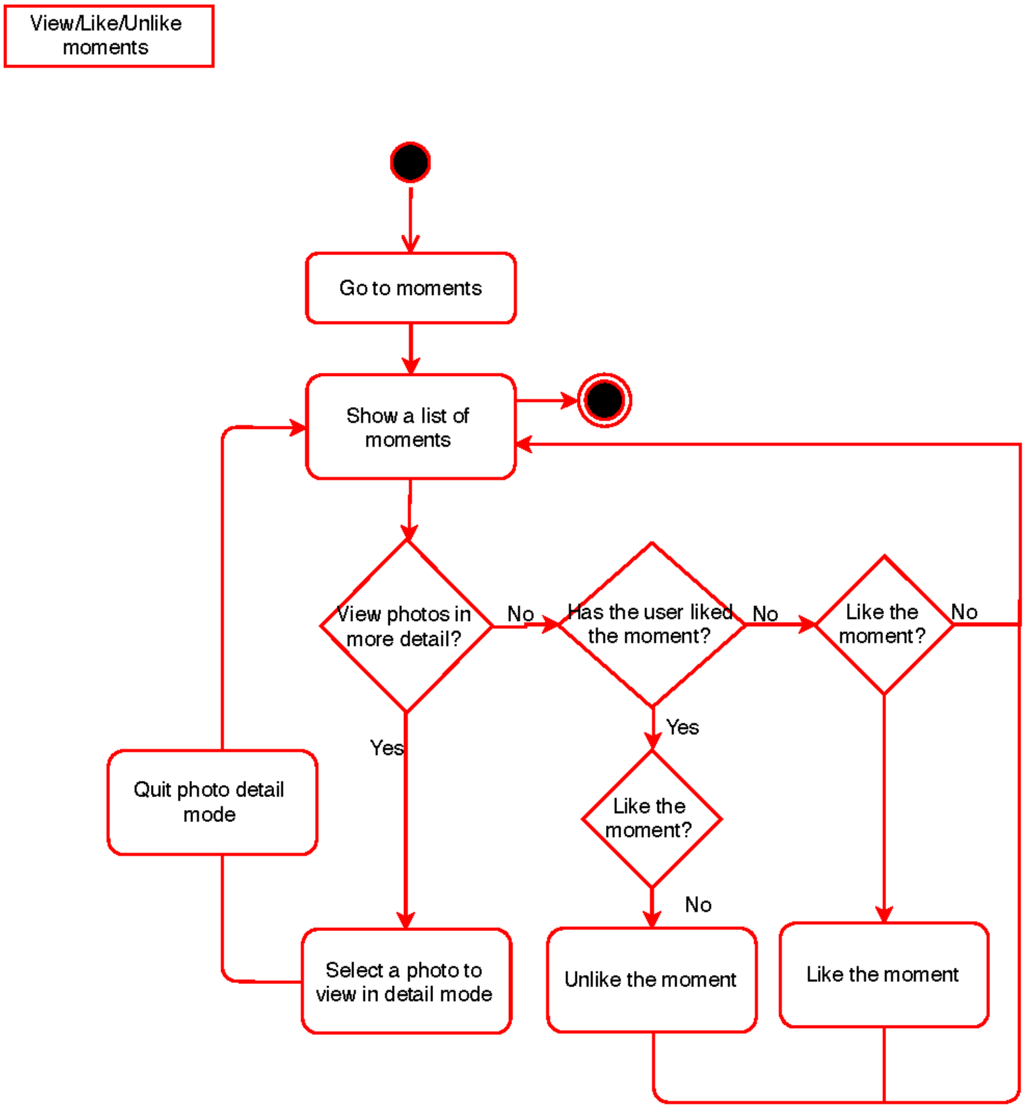
10.3.9. View joined event in calendar


View calendar
Select calendar button
View events of current month
Select "Day"
Select "Week"
Select "Year"
View events of current day
View events of current week
View events of current year
Select "Former" / "Next"
View events of former/next day/week/month/year

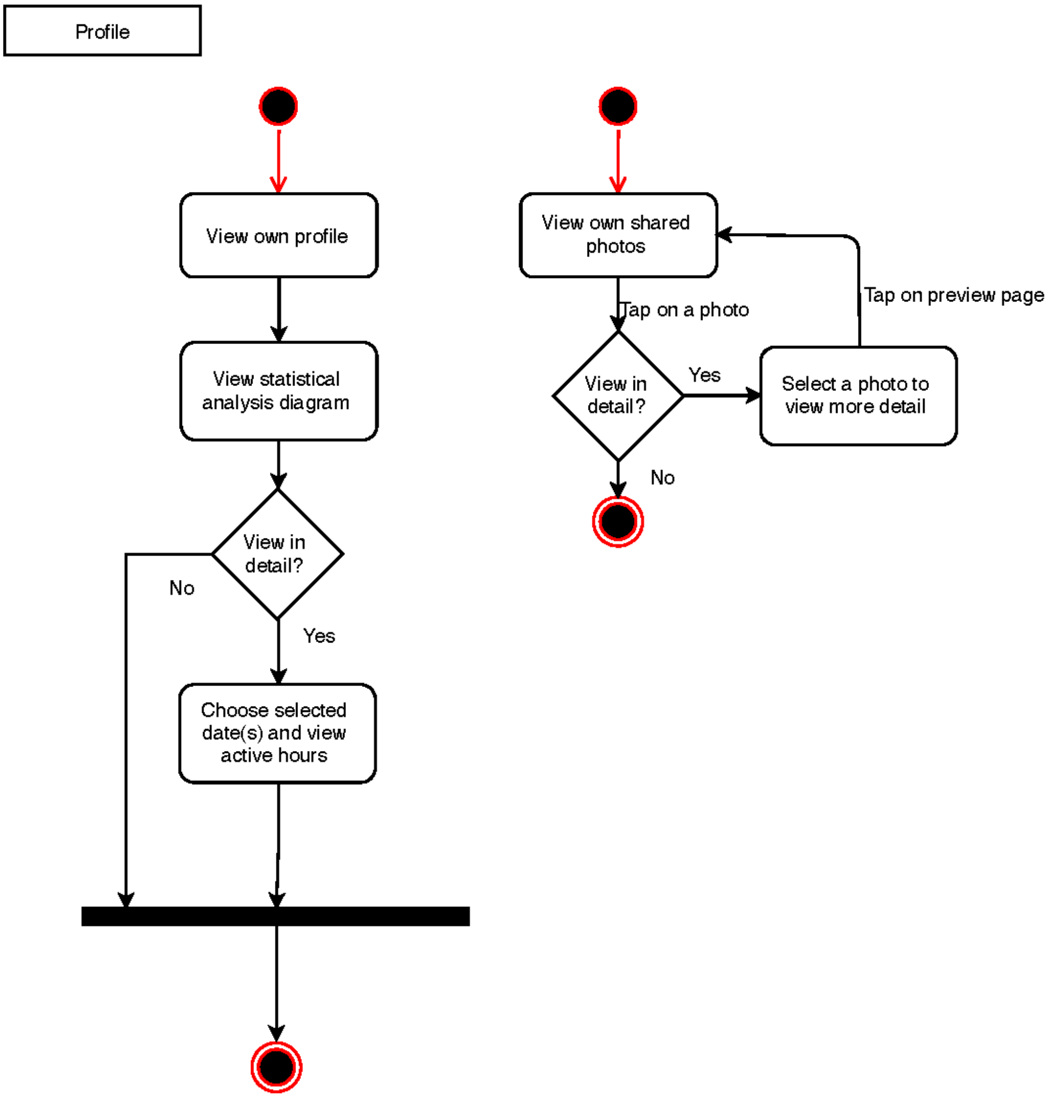
10.3.10. Share photo(s) in moments


Share photo(s) in moments
Select "New photo"
Choose existing photo(s)?
No
Take a new photo
Yes
Select photo(s) from gallery
Select new photo(s)
Go back to moments shared by friends
Go back to add more photo(s)
Yes
Want to add more photo(s)?
No
No
Confirm selection?
Yes
Select upload moment
Show main page of moments

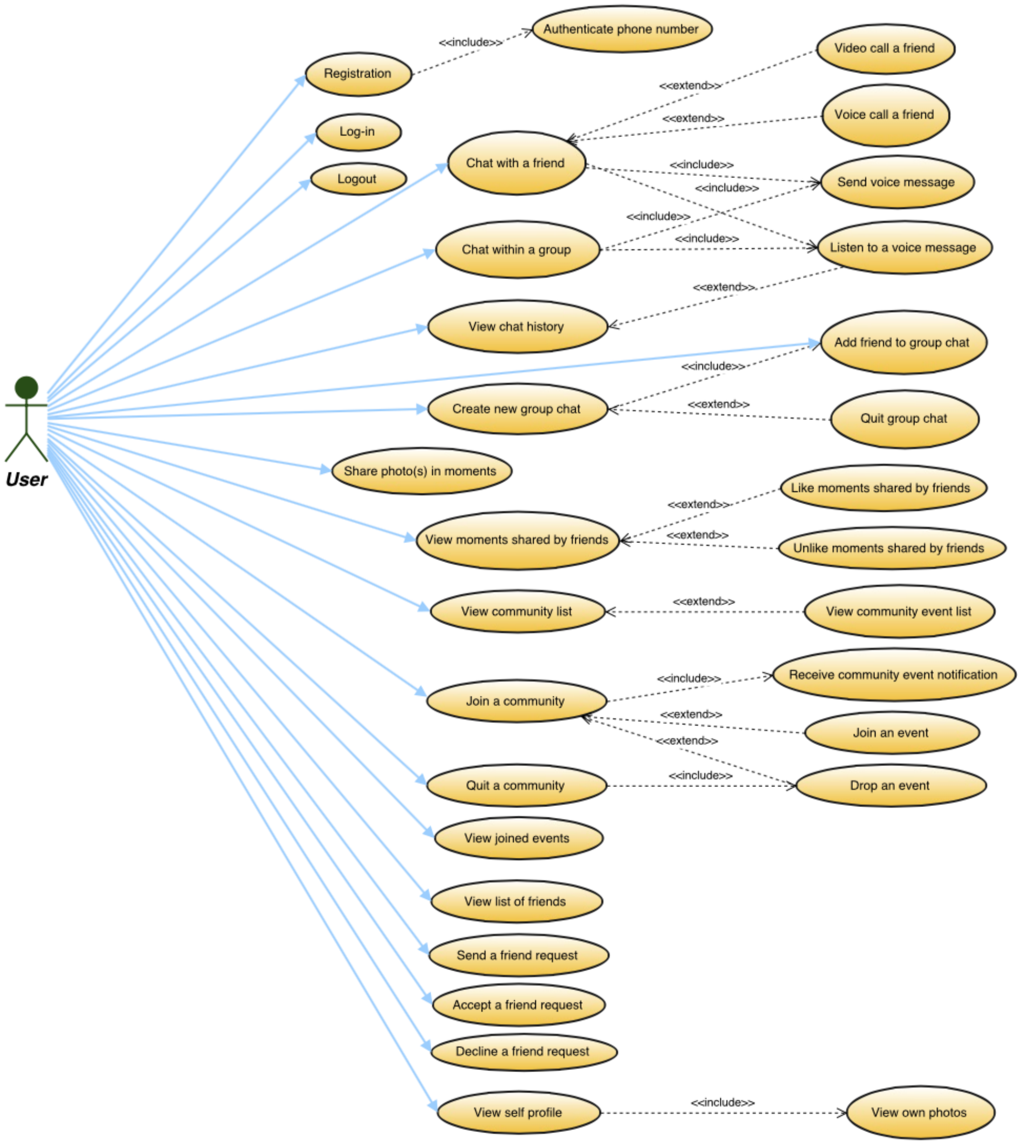
10.3.11. View/Like/Unlike moments



10.3.12. View own profile & shared photos



10.4. Use Case Diagram



11. Output Requirements

The output of the system is extremely simple: it returns a user interface to display user generated contents based on current page and user's request. It includes chat message, moments, communities, and other action records.

12. Hardware Requirements

The system requires the following hardware:

1. Mobile client device
2. A physical server
3. Production support system (hotline, back-up servers and database)
4. Network connection

13. Software Requirements

The system requires the following software:

1. Android (version 4.4 or above) system running on a mobile device
2. iOS (version 7 or above) system running on a mobile device
3. Linux or Windows operating system

14. Deployment Requirements

14.1. Client Application

Client application should be deployed on Android platform and iOS platform. The update of application should usually take place together, unless the update is to resolve a platform-specific problem.

The application should be released via Google Play Store and Apple AppStore.

14.2. Server Application

Server application deployment includes deployment of server-side business logic and database. The deployment of database should only happen in the initial deployment or in the case where an update of database schema is made. The deployment of server side business logic can happen no often than client application release, so as to provide maximum compatibility.