

Iteration 4 - Team A1 Requirements Proposal

Team ID/Name: ____A1____

TA Name: _Dimitris Sarlis____

Team members: Jiyu Shi (jiyus), Pan Li (panli), ConShang Lv (congshal),
Guangyu Chen (guangyuc), Jin Gu (jingu)

Within your team, each use case needs to be different and independent:

There should be **NO overlaps or dependencies between use cases.**

Make sure that your new use case do not rely on the internet or cellular network.

CLONE THIS DOCUMENT FOR YOUR TEAM IN [THE STUDENT DELIVERABLES FOLDER FOR SE PROJECT - ITERATION 4-5](#)

Check the following pages for the template for each use case and student...

Student 1	Proposal	Approved
Name: Jiyu Shi	Step 1: Use Case Name: Change Profile Use Case Brief Description: This use case allows the citizens to change the password and other information of the citizen's profile. In that case, the citizen needs to provide correct old password to the system to verify. Any information provided by the citizen in the profile page will be stored in the database once the citizen click the save and submit button. The User can also provide what kind of skills he has in order to help others in emergence.	Y Value: 3
	Faculty/TA Comments: It seems like an easy Use case to implement. If you allow for editing the whole profile (username, email, full name) then you can get 3 points.	
	Step 2: UI Mockups Link: https://drive.google.com/file/d/0B92804F8iyghUWVGa1JaYWNazEU/view?usp=sharing	Y Value*: 0.75
	Faculty/TA Comments: I approve only if you make the add skills functionality more clear. Is it a dropdown list that users can select from? They just enter some text in that textbox? Try to make it more clear. I think that an add skill button that will display a dropdown list to choose a skill is a good idea. Then the selected skills are displayed in the profile page. (fixed by jiyu , with new link and skills text window is replaced by a drop down list)	
	Step 3: Elaboration - Use Case Specification See below	Y Value*: 0.8
	Faculty/TA Comments:	

*Value point might be revisited as the target user has a better understanding of the functionality.

Use Case Specification:Change Profile

Participating Actors

The use case is initiated by a Citizen.

Brief Description

This use case allows the citizens to change the password and other information of the citizen's profile. In that case, the citizen needs to provide correct old password to the system to verify. Any information provided by the citizen in the profile page will be stored in the database once the citizen click the save and submit button. The User can also provide what kind of skills he has in order to help others in emergence.

Assumption

The Citizen is logged into the system.

Flow of Events

Basic Flow

1. The use case starts when the citizen elects to his profile page.
2. The system directs to his profile page and listed all the previous message stores in the database like user name, email address, first name, last name, skills.
3. The citizen can change anything he/she wants, including the user name.
4. The system will check if the user name is valid or already existed according to the **Valid User Name Rule**, and the email address is valid or not according to the **Email Format Rule**, and the page will be dynamic updated to show the citizen if the email and username is valid or not.
5. The citizen click the submit and save button.
6. The system updates the citizen information to reflect the changed profile and the fact if succeeded or failed.

Alternative Flows

- A1. In Step 2-6, the citizen can click any button in the navigation bar. the system direct to that page and the use case ends.
- A2. In Step 3, if the citizen click the change password button, the System will popup an alert and the citizen can choose to save his information or discard what he has done in the profile page or just cancel.if cancel , The system will direct to profile page otherwise the system will direct to a new page for the citizen to change password. The citizen provides his old password ,then type new password to confirm. The citizen can click confirm button and if the old password matches the password in database, the citizen will get a successful password change message and return to the profile page, otherwise will stay on this page with an error message. The citizen can click back button at any time and system will direct to previous profile page.

Rules

- **Valid User Name Rule:** Usernames are provided by users and should be at least 3 character long. They should be different from the list of banned usernames provided at: <http://blog.postbit.com/reserved-username-list.html>.
- **Email Format Rule:** The Email are provided by users and should at least contains a symbol "@", and should has form like "[xxx@xxx.xxx](#)" or "[xxx@xxx.xxx](#)".

Student 2	Proposal	Approved
Name: Pan Li	Step 1: Use Case Name: Delete Individual Message Use Case Brief Description: Use Case Brief Description: This use case allows citizen to delete the messages that he/she sent in chat publicly and chat privately functionalities. Once the message deleted by the sender, cannot display anymore. (Only the sender can see delete button behind each message, and once click on delete button, there will be a pop window to confirm and delete that message from database.) Additional feature: citizen can delete the whole conversations in chat privately functionalities.	Y Value: 3
	Faculty/TA Comments: Approved. Maybe you can add a feature of multiple deletion of messages. That would be cool.	
	Step 2: UI Mockups Link: (Updated) https://www.fluidui.com:443/editor/live/preview/p_9f8xFWZA7LJ5qqwdtbPf8GgN3zg4JdAB	Y Value*: 0.6
	Faculty/TA Comments: Approved. I think you are not supporting the multiple deletion feature. You can add this probably.	
	Step 3: Elaboration - Use Case Specification See below (Updated)	Y Value*: 0.8
	Faculty/TA Comments:	

*Value point might be revisited as the target user has a better understanding of the functionality.

Use Case Specification: Delete Message and Announcement

Participating Actors

The use case is initiated by a Citizen (In future implement, "Delete Announcement" can only be done by Monitor).

Brief Description

This use case allows the citizens to delete one or more messages in public chat conversation or private chat conversation and delete multiple announcements. In that case, the citizen can select to delete one or more messages that he/she sent. As for the announcement, the citizen (Monitor, in future iterations) can select a range of dates to delete multiple announcements. All of the selected messages and announcements will be deleted from the database once the citizen click the confirm button and the operation is irrevocable.

Assumption

The Citizen is logged into the system.

Flow of Events

Basic Flow

1. The use case starts when the citizen elects to click on “Delete Message” button in chat public/private conversation.
2. The system will re-display all messages and, if the sender of message is current citizen, there will be a clickable checkbox appears in the front.
3. The Citizen clicks on checkbox if the citizen wants to delete the message and clicks on “Delete All” button.
4. The system will display a pop window to confirm whether the citizen wants to delete those messages.
5. The Citizen clicks on “Yes, delete it” button.
6. The system will remove that message from database and it will no longer show up in the chat room.

Alternative Flows

- A1. In Step 3-6, the Citizen can click the “cancel” button and the pop-up window disappears and nothing changes.
- A2. In Step 3-4, if the Citizen does not choose any message to delete and clicks on “Delete All” button, the system will display a pop window to warning prompt that “You must choose at least one message to delete!”
- A3. In Step 1-6, if the Citizen (Monitor in future iteration) is in Post Announcement page and want to manage the expiration announcement (according to the timeliness of announcement), the Citizen also can click on “Delete Announcement” button to delete the multiple announcements by time range. There will display a form and the Citizen can chose the date using datepicker. Once the Citizen clicks on “Delete All” button, (If there is no message in that date range, the system will display a error alert.) the system will display a pop-up hint box to ask Citizen whether confirm the deletion of those announcement. After the Citizen clicks on “confirm” button, the particular messages in that date range do not exist anymore.

Rules

- The checkbox will only appear next to each chat message for the message sender, which means, only the sender has the permission to delete the message from the database by clicking checkbox in public/private chat room.
- The date range picker will be clickable when it is prior to today. And the end date must be greater than start date.

Student 3	Proposal	Approved
Name: Guangyu Chen	Step 1: Use Case Name: Real-time Emergency Notification Use Case Brief Description: We have already implemented public chatting and post announcement in previous iteration. However, they cannot draw enough attention. For instance, when users are on other pages. He cannot see the update of public message and announcement. I propose to add a new feature to push real time notification to users. It will give users real-time prompts and sound notifications. Notification sound may varies due to the content of notification which means there are multiple types of notification. For instance, an alert means emergency and an exciting sound may indicate good news, such as "Help is coming!". In iteration 4, this feature is available for every user. After iteration 5, it can only be accessed by administer.	Y Value: 3
	Faculty/TA Comments: Approved. Sounds like a really interesting feature.	
	Step 2: UI Mockups Link: https://drive.google.com/file/d/0B8CzYnzheOK_SE9HSDNfc0ZWdmc/view?usp=sharing	Y Value*: 0.75
	Faculty/TA Comments: Approved. Good job.	
	Step 3: Elaboration - Use Case Specification See below	Y Value*: 1
	Faculty/TA Comments:	

*Value point might be revisited as the target user has a better understanding of the functionality.

Use Case Specification: Instant Message

Participating Actors

In iteration 4, this feature is available for every user. After iteration 5, it can only be accessed by administer.

Brief Description

This feather allows user to send real-time notification to all members in community. For other user, all members will get real-time prompts and sound notifications no matter which page they are currently in . Notification sound may varies due to the content of notification which means there are multiple types of notification. For instance, an alert means emergency and an exciting sound may indicate good news, such as "Help is coming!".

Assumption

The Citizen has logged into the system. In iteration 5, the citizen has have administered permissions.

Flow of Events

Basic Flow

1. The use case starts when user clicks on the "instance message" button on menu bar.
2. The system will redirect the page to "instance message" page.

3. The sender will type in the content of message and choose the type of notification.
4. The system will send the notification to every online citizens.
5. The receivers(all citizens) will receive the notification with content and sound and a confirm button.
6. The receiver will click on the confirm button and the notification button will disappear.

Alternative Flows

- A1 In 1-3, sender can cancel this notification by click on other buttons on the menu bar. Sender will then be redirect to other user cases. The unfinished notification will be abandoned automatically.
- A2 In 6, receiver can click on make this notification disappear by click on the buttons on the menu bar. It will achieve the same goals as “confirm” button.

Rules

- **Short Message Rule - the message need to be short and concise. So the content of notification is limited to 20 characters.**

Student 4	Proposal	Approved
Name Jin Gu	Step 1: Use Case Name: chat in a group Use Case Brief Description: We have implemented chat privately. But it is only a one by one connections. If a user want to chat with several friends, the user need to chat with them one by one to tell the same story. Therefore, users need a function to chat with their friends in a group but not the whole community. Under this pattern, users can invite other online users to a group chat, and the users who have been invited can see a system message to notice that they have been invited and go to the group chat page to join a group chat.	Y Value: 3
	Faculty/TA Comments: Approved. Useful feature for an emergency social network.	
	Step 2: UI Mockups Link: https://drive.google.com/file/d/0B1MJGesjpMQ5UIhPazU5VklzRWs/view?usp=sharing	Y Value*: 0.75
	Faculty/TA Comments: Your mockups are incomplete. There is no screen that shows me how I can invite a user. For example, from the user list if I click someone's name I should get a prompt to begin a group chat. Furthermore, if I already am in some group chats, I should be able to add a user to an existing chat. You need to add this for your mockups to be complete. [FIXED]	
	Step 3: Elaboration - Use Case Specification See below	Y Value*: 1
	Faculty/TA Comments: Approved. Good job.	

*Value point might be revisited as the target user has a better understanding of the functionality.

Use Case Specification: Group Chat

Participating Actors

The use case is initiated by a Citizen.

Brief Description

The use case allows the Citizen to chat with several other Citizens instead of the whole community or only one Citizen.

Assumption

The Citizen is logged into the system.

Flow of Events

Basic Flow

1. The user case starts when the Citizen click the user names on User List page.
2. The system give a prompt to ask the Citizen whether the Citizen want to start a group chat.
3. The Citizen choose to start a group chat.

4. The system direct the current page to group chat page.
5. The Citizen selects other Citizens to join a group chat.
6. The System sends a system message to Citizens who have been invited
7. The Citizens join the group chat by clicking the Group Chat option and then send message.
8. The System display the message sent in a group chat.

Alternative Flows

- A1. At step 3, if the Citizen choose not to start a group chat, the user case ends.
- A2. At step 3, if no other Citizens is online, no one should be selected. The use case continues at step 3.
- A3. At step 5, the Citizens who have been invited can choose to joined to group chat later or ignore it. The use case stops or ends
- A4. At step 6, if no one posts a message, none is displayed. The use case continues at step 6.
- A5. Any time the Citizens click Go Back option, the use case ends.

Rules

- **Start Group Chat Rule:**If there is no other Citizen, the use case can not start.
- Join Group Chat Rule:**If the Citizen hasn't been invited, the Citizen can not join into the group chat.

Student 5	Proposal	Approved
Name: Congshan Lv	Step 1: Use Case Name: Send and View Post Use Case Brief Description: This use case allows user to send a text post about how he/she feels right now. All history posts are saved to the database and can be viewed in reverse chronological order. User can view other people's posts when clicking their name on the user list. The post also have timestamp feature and an notification to others. Each post would have a timestamp with it. There would be an notification on the navigation bar of the number of new posts.	Y Value: 3
	Faculty/TA Comments: Approved. I would prefer to see the timestamp of the post also, or even a notification that the user has changed their posts. This would give you full points.	
	Step 2: UI Mockups Link: https://drive.google.com/open?id=0B0KJ_lmO1tU4S0JtMUVtX0tmMjA New Link: https://drive.google.com/file/d/0B0KJ_lmO1tU4bnd5Q2ZsMkoxcDQ/view?usp=sharing	Y Value*: 0.75
	Faculty/TA Comments: Approved, although I didn't see if you have a notification when a user posts something new. Maybe you could add this screen also.	
	Step 3: Elaboration - Use Case Specification See below	Y Value*: 1
	Faculty/TA Comments:	

*Value point might be revisited as the target user has a better understanding of the functionality.

Use Case Specification: Send and View Post

Participating Actors

The use case is initiated by a Citizen.

Brief Description

This use case allows user to send a text post about how he/she feels right now. All history posts are saved to the database and can be viewed in reverse chronological order (latest one first). User can view other people's posts when clicking their name on the user list. The post also have timestamp feature and an notification to others. Each post would have a timestamp with it. There would be an notification on the navigation bar of the number of new posts. Whenever a new post is published, all the online users would receive a notification.

Assumption

The Citizen has logged into the system.

Flow of Events

Basic Flow

1. The use case starts when the Citizen elects to view the post page.
2. The system directs to the post page, which contains an input area to send new post and also a list of all the previous posts stored in the database. Each post would contain sender name, content, and timestamp.
3. The Citizen can choose to send a new post by typing in the input area, and then click the POST button.
4. The system stores and displays the post(together with sender name, timestamp) so others can see it. Other logged in Citizens would receive a notification about it.

Alternative Flows

- A1. In step 2, the Citizen can choose to view another person's posts by clicking on the post send by that person. The system would directs to a post page that only contains the posts from that person.
- A2. In step 2, the Citizen can choose to view his own posts. The system would directs to a post page that only contains the posts from himself.
- A3. In step 3, the Citizen can choose to cancel. The use case ends.

Rules

- **Valid Input Rule:** The content of the post cannot be empty.