

Shared To-Do List
CSE 5324-002
Coffee_Code - Group 6
Hannan Khan
Yanzhi Wang
Anum Farrukh Khan
Mohammed Furkhan Shaikh
Yu Zhao

Table of Contents

1. Change Log.....	1
2. Project Description.....	2-3
3. Overall Design Approach.....	4
4. Requirements.....	5
5. Use Case List.....	6
6. High Level Use Cases.....	7-8
7. Use Case Diagram.....	9
8. Requirements to Use Case Traceability Matrix.....	10
9. Increment Matrix.....	11
10. Domain Model Diagram.....	12
11. Expanded Use Cases (& User Interface Prototypes).....	13-29
12. Design Sequence Diagrams.....	30-42
13. Design Class Diagram.....	43
14. Task List.....	44

Our YouTube demonstration can be found at:

[ShareList](#)

Project Change Log

Release #	Change Description	Date	Affected Page(s)
Iteration 1	Original Release	03-12-2021	All
Iteration 2	Original Release	04-09-2021	All
Iteration 3	"Time Calculation" removed from project description and requirements, for clarification.	05-03-2021	2,3,5
	Project Description updated with clarification on wireless resources, (also, new line numbers).	05-03-2021	2,3
	Requirements updated to match project description.	05-03-2021	5

Project Description

1 **ShareList** - Since the start of the pandemic, many workplaces have switched to working from
2 home (working online). This can cause problems in collaboration within the teams as many places
3 are switching to working online for the first time. There already exist many different apps which
4 cater to only large organizations, have a steep learning curve, or are not available/accessible to
5 everyone. We would like to create a mobile app which is intuitive to use and accessible by
6 everyone. An app where a user can create a group of tasks, and decide which users will work on
7 it. This app can be used whether the group follows the SCRUM model, waterfall model, Agile, or
8 no model at all.

9 This to-do app is designed to allow users to create, delegate, and track tasks that are
10 shared amongst users in a group. The app needs the internet to connect with the database so all
11 users can stay synchronized. The main purpose of the app is to allow group members in a virtual
12 environment to have better collaboration. The app can be used by anyone wishing to
13 create/delegate tasks with others and not just for industry use. Whether you are a group of
14 students working on a semester project, or a family managing the chores, or an organization
15 managing their work on the go, this app is an efficient, fast, user-friendly and smooth task-
16 management tool for you!

17 *Group Admin:* Any user that has created a group is the admin of that group. The group
18 admin can add users to the group, assign who works on what tasks, promote users to admin
19 status, and add/remove/edit tasks.

20 *User:* Anyone who is part of a group, but is not admin. Can be assigned to tasks by the
21 admin. A user can mark a task as completed.

22 *Task:* Tasks can only be created by the group admins. All tasks in a group are visible to
23 all members within that group. Each task has users assigned to it.

24

25 Each task has properties such as:

- 26 • Start time- creation date/time of task.
- 27 • Completion time- when the task has been marked completed by all those
28 assigned to it.
- 29 • Due Date - the date the admin expects the task to be done by.
- 30 • Time spent - the time from the start time to completion time.
- 31 • Title, description, users assigned.

32 *Group:* "Groups" are collections of tasks that belong to the members of that group. Groups
33 can be created by any user of ShareList. When a user creates a group, they have admin status.
34 Members of the group include admins and users. Group admins can control who is
35 added/removed from the group. Tasks can be added by the group admins only. The whole
36 concept of groups is to have a specific goal that all tasks within that group are working towards.
37 Think of a group as a "project" where the members of the group work towards a certain goal,
38 accomplishing tasks along the way.

39

40 **Functions -**

- 41 • Login/Profile: Create/edit login credentials, manage sessions, etc.
- 42 • Add/Remove/Edit/Assign Tasks: Group Admins can add/remove tasks from the group,
43 edit who the task is assigned to, what the task contains, due date, start date, etc.
- 44 • Search tasks/groups/users: Search by keyword.

- 45 ● Mark Completed: Allows for users to mark a task as completed.
46 ● Create/Edit Group: Allows a user to create a group of unique users. The creator is initially
47 set as the group admin. The group admin can add additional admins.
48 ● Delete Group: Allows any admin of a group to delete the group. Deleting the group will
49 permanently delete the group and all related records.
50 ● Group Summary: Calculates and reports to the user a summary of the group, which
51 includes:
52 ○ Tasks completed - number of completed tasks.
53 ○ In-progress tasks - number of in-progress tasks.
54 ○ Tasks not completed – number of remaining tasks.

55
56 **Utilized Resources -**

- 57 ● Database: Used for storing information related to tasks, groups, users.
58 ● Wireless: Used for syncing between each user's device and the database. The syncing
59 action is handled by the Firebase Realtime Database System, not the ShareList System.

60
61 **Android Experience per Team Member -**

62 Hannan Khan - No prior experience with Android. I have implemented/developed simple
63 apps in Java when I had started out programming. In terms of app development, I mainly focus
64 on apps in Python, which I consider myself to be strong in. These apps consisted of both frontend
65 and backend work.

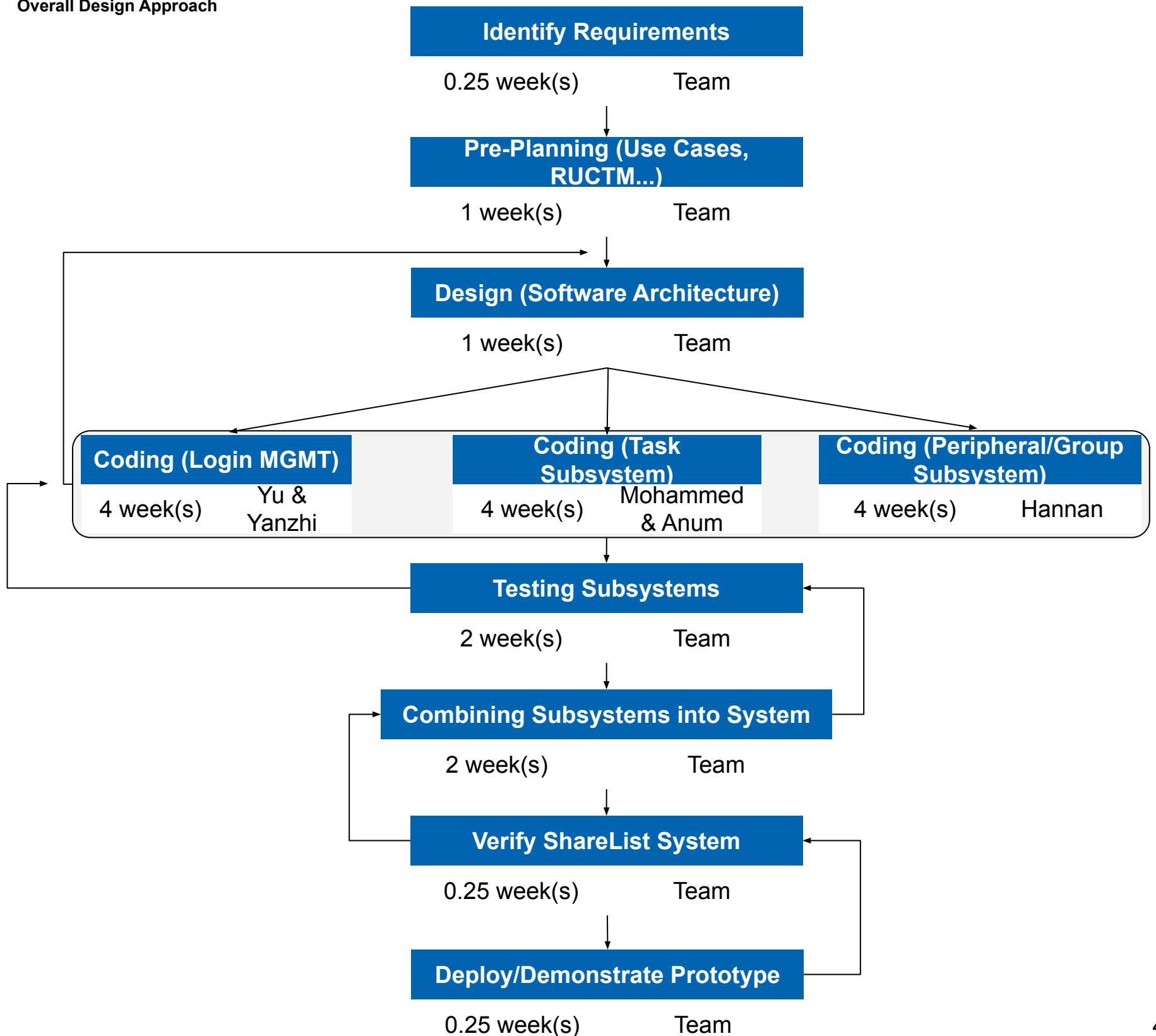
66 Yanzhi Wang - I do not have a lot of Android experience and I have not worked on any
67 Android project during the previous semester. The project will take the whole semester, so we will
68 have plenty of time to learn. I hope everyone in the team will learn something about Android, UML,
69 and the agile process by working together on the project.

70 Yu Zhao - Do not have any experience with Android development. I have some small full-
71 stack applications experience, and familiar with programming languages like Java, Python, and
72 JavaScript. I can work on both the frontend and backend.

73 Mohammed Furkhan Shaikh - Does not have prior experience with Android application
74 development. I have experience with programming languages like Java, C, Python. I'm familiar
75 with object-oriented concepts. I can work on the backend and implement functions for the app.
76 No experience with the formal design process.

77 Anum Farrukkh Khan - No prior experience with Android development. I am familiar with
78 python, java and javascript and have developed various applications using these languages. I
79 have developed a mobile application in react native as part of an academic project.

Overall Design Approach



Requirements

Req ID	Req Statement	Line (From Project Description)
R1	The system shall provide for authenticated login to the ShareList app.	
R1.1	Register.	41
R1.2	Login.	
R1.3	Log Off.	
R1.4	Reset password.	
R2	The system shall provide group admins the ability to:	
R2.1	Add tasks.	42,43
R2.2	Remove tasks.	
R2.3	Edit tasks.	
R2.4	Assign tasks.	
R2.5	Add users to group.	46,47
R2.6	Remove users from the group.	34,35
R2.7	Add new group admins to group.	18,47
R2.8	Delete group	48,49
R2.9	Edit group.	46,47
R3	The system shall provide all users the ability to:	
R3.1	Mark task as completed.	45
R3.2	Create groups.	46,47
R4	The system shall provide the ability to search by keywords.	44
R5	The system shall provide tasks the ability to:	
R5.1	Be visible to all members in that group.	22,23
R5.2	Have users assigned to it.	42,43
R5.3	Have assigned users mark it as completed.	45
R6	The system shall provide tasks the following properties: start time, completion time, due date, time spent, title, description, users assigned	26,27,28,29,30,31
R7	The system shall provide users the ability to create groups, and have group admin status when they do.	46,47
R8	The system shall provide a database used for storing information related to tasks, groups, users.	57
R9	The system shall provide wireless resources used for syncing between each user's device and the database in realtime.	58,59
R10	The system shall provide a group summary function, which will accomplish:	50
R10.1	Calculating and reporting the tasks completed.	52
R10.2	Calculating and reporting the in-progress tasks.	53
R10.3	Calculating and reporting the remaining tasks.	54

Use Case List

Use Case #	Use Case Name
UC1	Login
UC1.1	Register.
UC1.2	Login.
UC1.3	Log Off.
UC1.4	Reset password.
UC2	Manage task
UC2.1	Add tasks.
UC2.2	Remove tasks.
UC2.3	Edit tasks.
UC2.4	Assign tasks.
UC3	Manage group
UC3.1	Add users.
UC3.2	Remove users.
UC3.3	Add admins.
UC3.4	Delete groups
UC3.5	Edit groups.
UC4	Mark task as completed
UC5	Create groups.
UC6	Search by keywords.
UC7	Generate summary

High Level Use Cases

UC 1.1: Register

- TUCBW the user clicks on the 'Register' button.
- TUCEW the new user is registered successfully.

UC 1.2: Login

- TUCBW the user enters their details and clicks on the 'Login' button.
- TUCEW the user successfully logs in.

UC 1.3: Log Off

- TUCBW user clicks on the 'Logout' button.
- TUCEW user successfully logs out of the application.

UC 1.4: Reset Password

- TUCBW user clicks on the 'Forgot Password' button.
- TUCEW the password of the user is changed.

UC 2.1: Add Tasks

- TUCBW group admin clicks on the 'Add Task' button.
- TUCEW the group admin sees a confirmation "The task has been added" on the app.

UC 2.2: Remove Tasks

- TUCBW group admin clicks on the 'Remove task' button.
- TUCEW the group admin sees a confirmation of the task being removed.

UC 2.3: Edit Tasks

- TUCBW group admin clicks on the 'Assign User' button.
- TUCEW group admin sees the changes.

UC 2.4: Assign Tasks

- TUCBW the group admin clicks on the 'Update Assigned Users' button.
- TUCEW group admin sees the user is assigned to the task.

UC 3.1: Add users

- TUCBW admin clicks on the 'Add User' button.
- TUCEW group admin sees the user added to the group.

UC 3.2: Remove users

- TUCBW admin clicks on the 'Remove User' button.
- TUCEW group admin sees the user is no longer a member of the group.

UC 3.3: Add admins

- TUCBW group admin clicks on the 'Add Admin' button.
- TUCEW group admin sees the user is now an admin.

UC 3.4: Delete Group

- TUCBW group admin clicks on the 'Delete group' button.
- TUCEW the group, related tasks and users, are removed from the records.

UC 3.5: Edit Group

- TUCBW admin clicks on the 'Rename group' button.
- TUCEW group admin sees that the group name is updated.

UC 4: Mark task as completed

- TUCBW user clicks on the 'Mark Task As Completed' button.
- TUCEW the user sees the task as crossed off.

UC 5: Create Group

- TUCBW user clicks on the 'Create Group' button.
- TUCEW the user sees a confirmation "The group has been added".

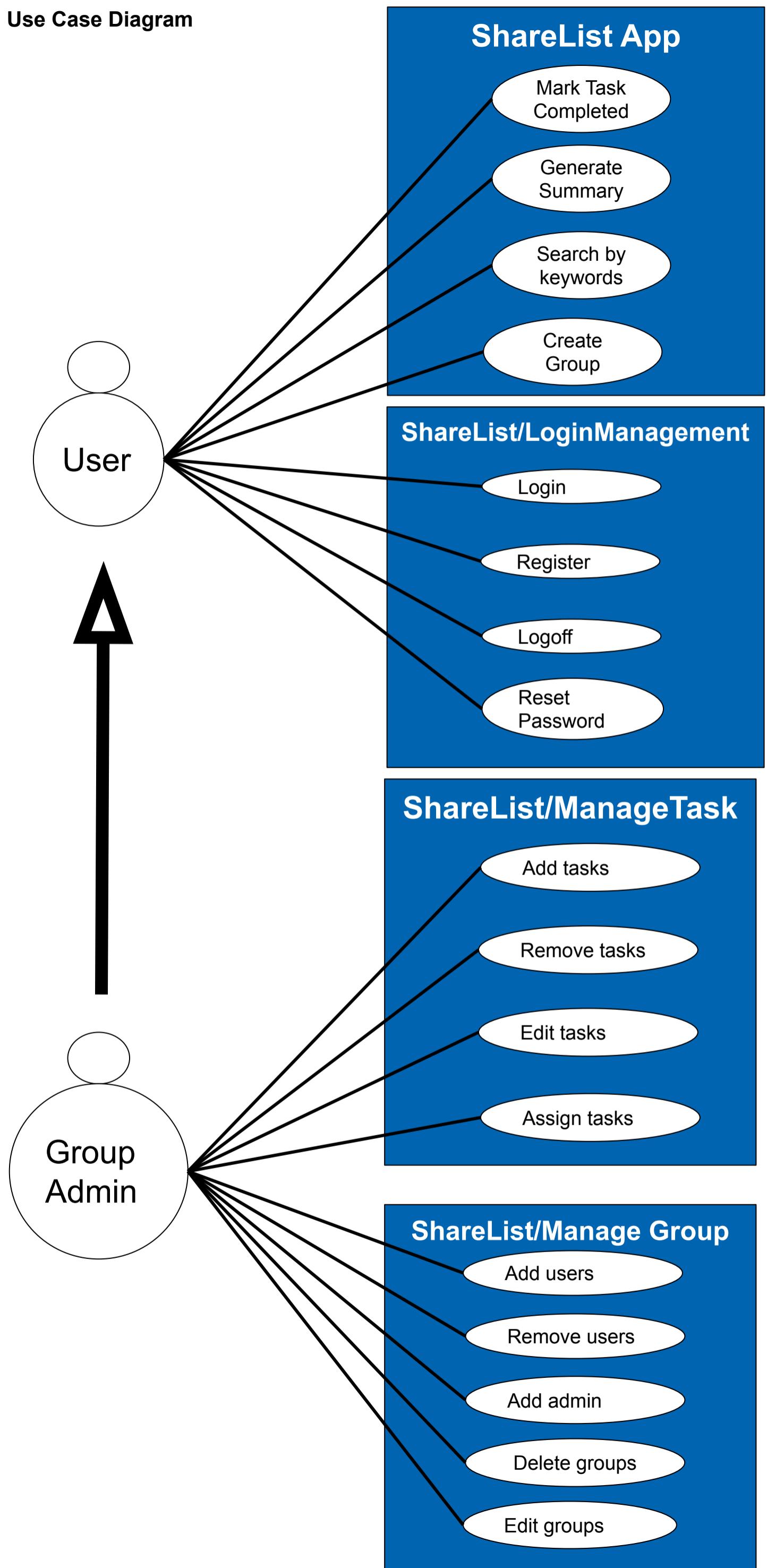
UC 6: Search by keywords

- TUCBW user clicks on the 'Search' button and enters the search keyword.
- TUCEW user sees the correct search results.

UC 7: Generate summary

- TUCBW user clicks on the 'Group Info' button.
- TUCEW summary of group is displayed to the user.

Use Case Diagram



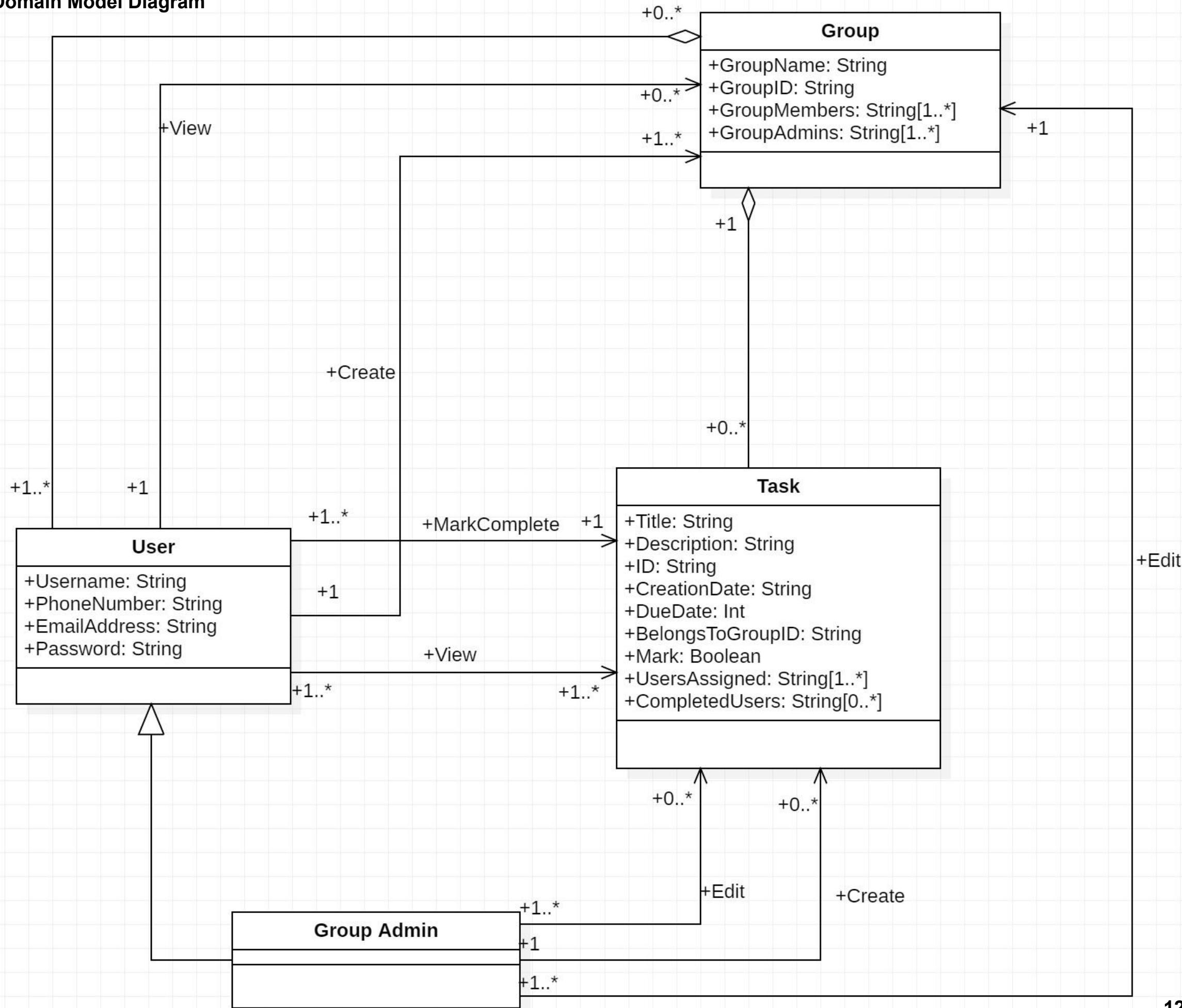
RUCTM

Req ID	Priority Weight	UC 1	UC 1.1	UC 1.2	UC 1.3	UC 1.4	UC 2	UC 2.1	UC 2.2	UC 2.3	UC 2.4	UC 3	UC 3.1	UC 3.2	UC 3.3	UC 3.4	UC 3.5	UC 4	UC 5	UC 6	UC 7	
R1	3	X																				
R1.1			X																			
R1.2				X																		
R1.3					X																	
R1.4						X																
R2	1							X														
R2.1									X													
R2.2										X												
R2.3											X											
R2.4												X										
R2.5													X									
R2.6														X								
R2.7															X							
R2.8																X						
R2.9																	X					
R3	3																					
R3.1																		X				
R3.2																			X			
R4	6																				X	
R5									X													
R5.1										X												
R5.2											X											
R5.3												X										
R6	1																					
R7		1															X					
R8		2																				
R9		4																				
R10	5																				X	
R10.1																					X	
R10.2																					X	
R10.3																					X	
Score		3	3	3	3	3	2	2	2	2	1	1	1	1	1	1	1	1	1	7	24	
Legend:		1 is highest priority. Work 1 First.																				

Increment Matrix

Use Case #	Priority	Effort (Person-weeks)	Depends on	Iteration 1 (03/12)	Iteration 2 (04/09)	Iteration 3 (05/03)
UC1	3	6	None			
UC1.1					6	
UC1.2					6	
UC1.3					6	
UC1.4						6
UC2	2	2	None			
UC2.1				2		
UC2.2						2
UC2.3					2	
UC2.4					3	
UC3	1	4	UC 1			
UC3.1					4	
UC3.2					4	
UC3.3						4
UC3.4						4
UC3.5						4
UC4		3	UC 2		3	
UC5		4	UC 3		4	
UC6	5	2	UC 2		2	
UC7	4	1				2
Total Effort		63		2	40	22
1 PW = 5 hrs				10	200	110

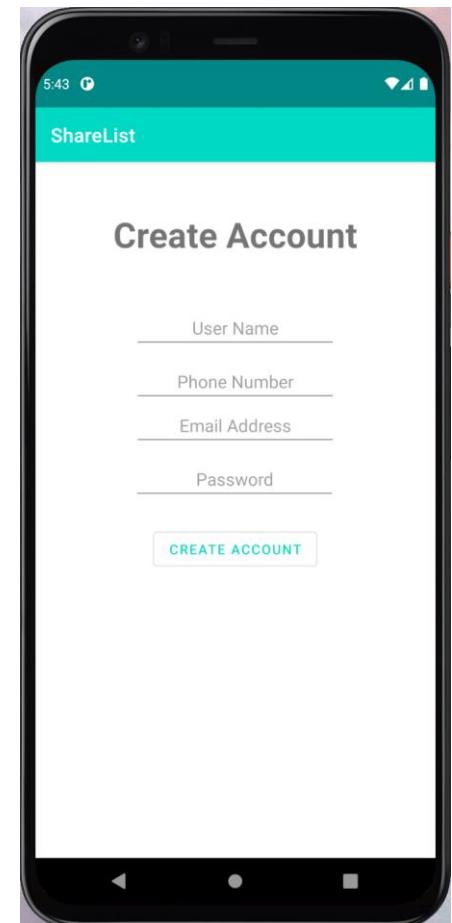
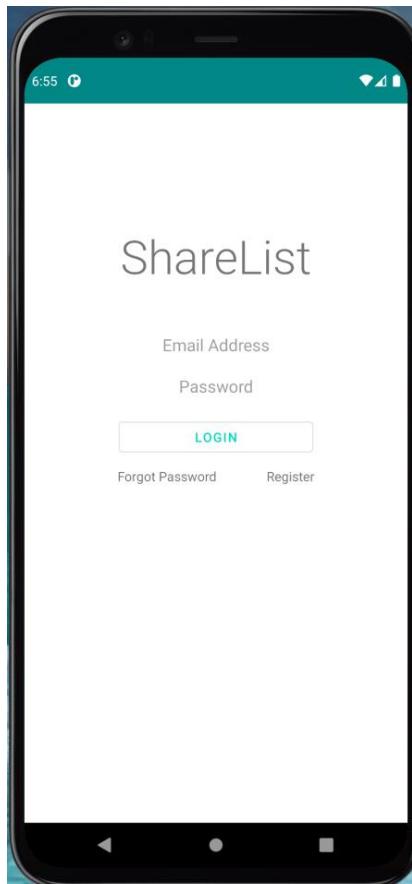
Domain Model Diagram



Expanded Use Cases with User Interface Prototypes

EUC1.1: Register

Actor: User	System: ShareList/LoginMGMT
	0. System displays the Login page.
1. TUCBW the user clicks on the 'Register' button.	
3. User enters credentials and clicks 'Create Account'.	2. System displays the Register Activity page.
5. TUCEW the new user is registered successfully.	*4. System checks if the information is okay, displays a confirmation, and takes the user back to the Login page.
Postcondition: The user's new account is stored in the database, ready for authentication.	

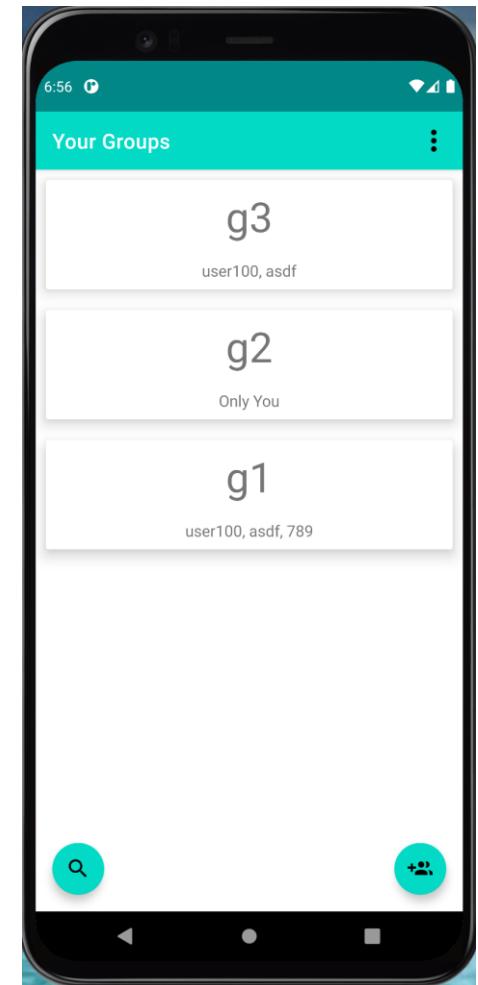
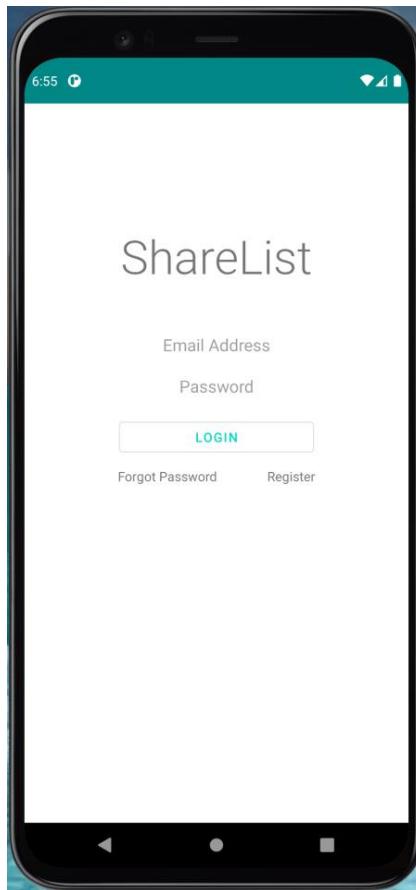


EUC1.2: Login

Precondition: This use case assumes that the user is seeing the Login page and is a registered user.

Actor: User	System: ShareList/LoginMGMT
1. TUCBW the user enters their details and clicks on the Login button.	0. System displays the Login page. *2. System checks the credentials entered by user: a) If the credentials are correct, the users 'Your Groups' page will be displayed. b) If incorrect, the system will display an "Incorrect email or password!" message.
3. TUCEW the user successfully logs in.	

Postcondition: The user will be able to access the features of the app.

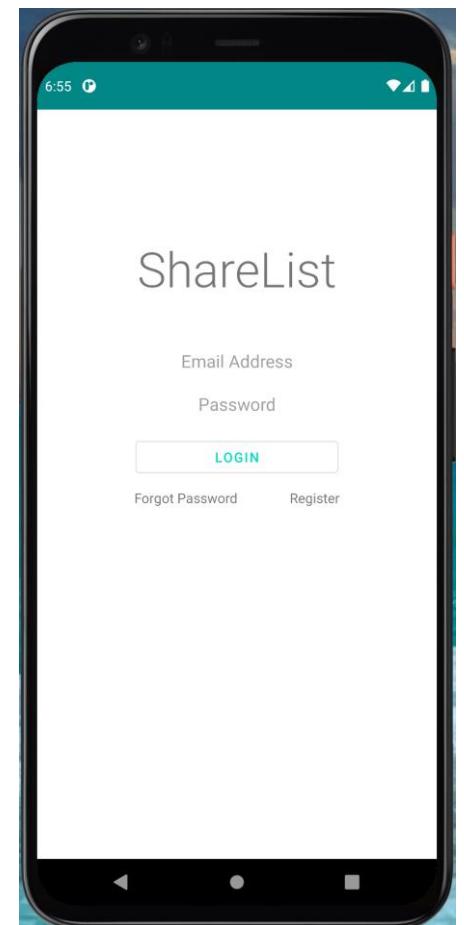
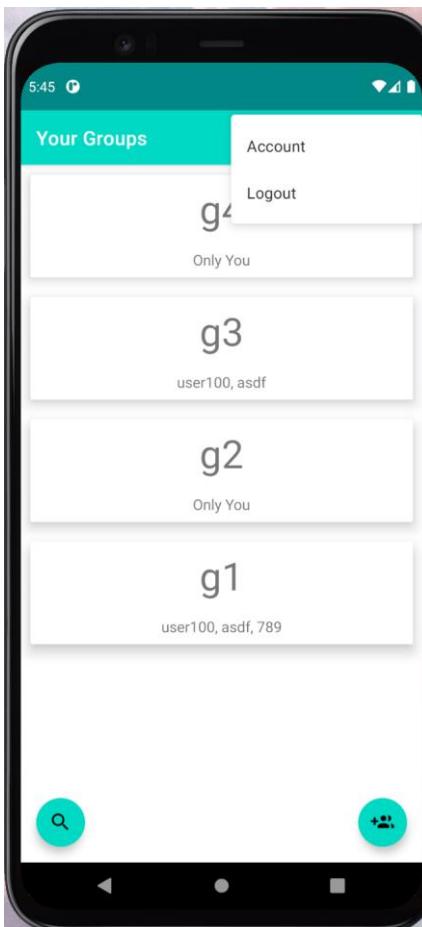


EUC1.3: Log Off

Precondition: This use case requires that the user is logged in and is on the Your Groups page.

Actor: User	System: ShareList/LoginMGMT
1. TUCBW user clicks on the Logout button.	0. System displays the Logout button on the corner menu.
3. TUCEW user successfully logs out of the application.	2. System displays the Login/Register page.

Postcondition: The user must be able to login again.

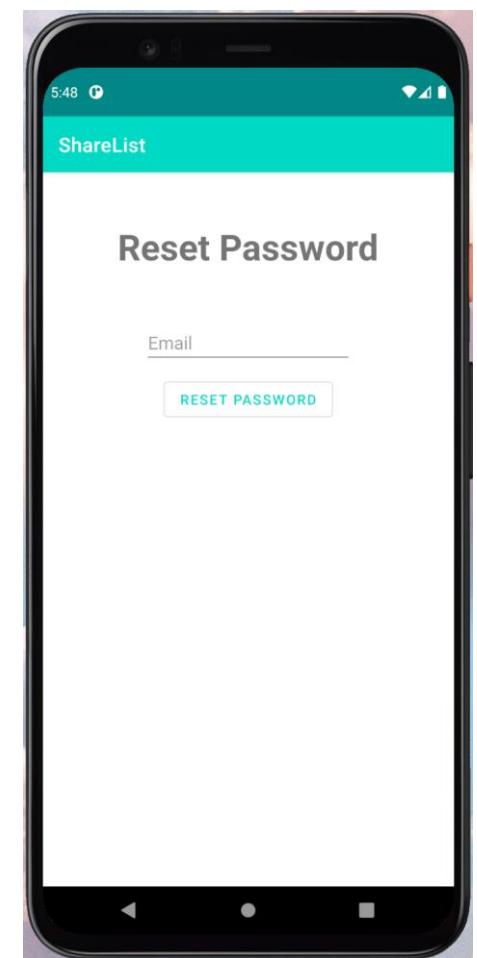
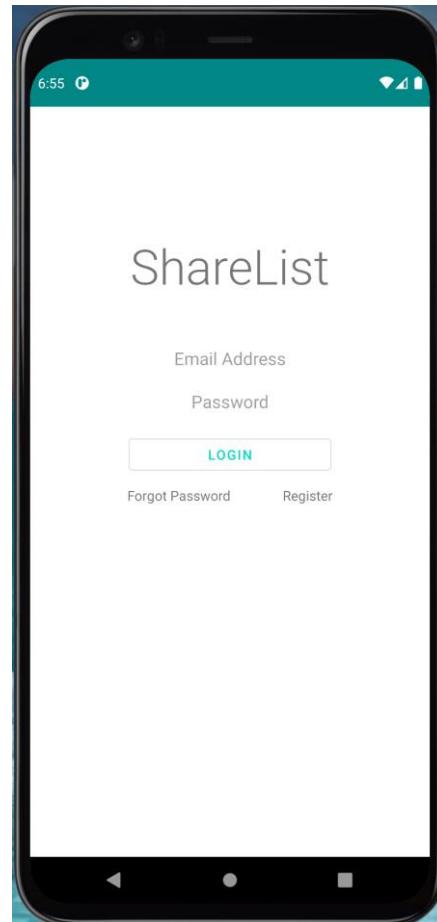


EUC1.4: Reset Password

Precondition: This use case requires that the user is already registered.

Actor: User	System: ShareList/LoginMGMT
1. TUCBW user clicks on the Forgot Password button.	0. System displays the Login/Register page.
3. The user enters the email id and clicks on the Reset Password button.	2. System prompts the user to enter the registered email id.
5. The user clicks on the link received in email and enters the new password.	4. System sends a link to the email to reset the password.
7. TUCEW the password of the user is changed.	6. System successfully updates the new password.

Postcondition: The user's new password is stored in the database.

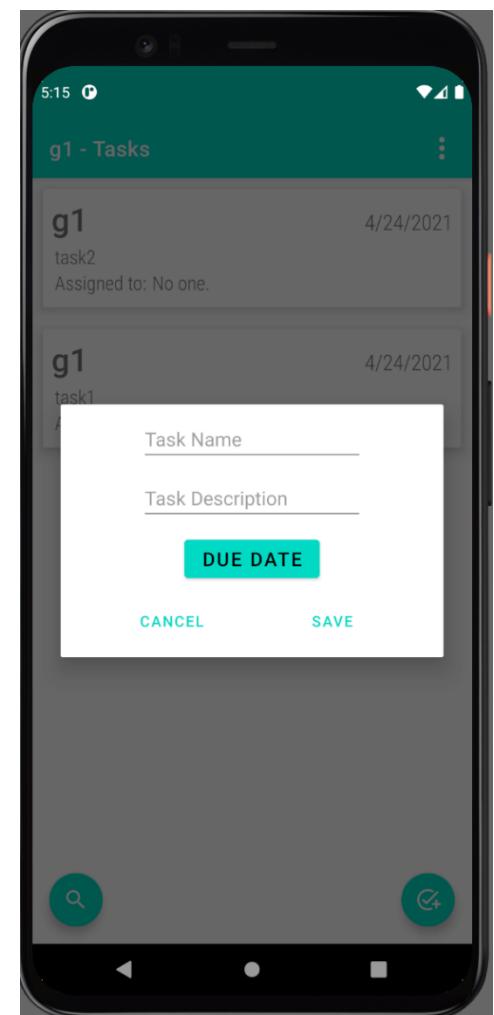
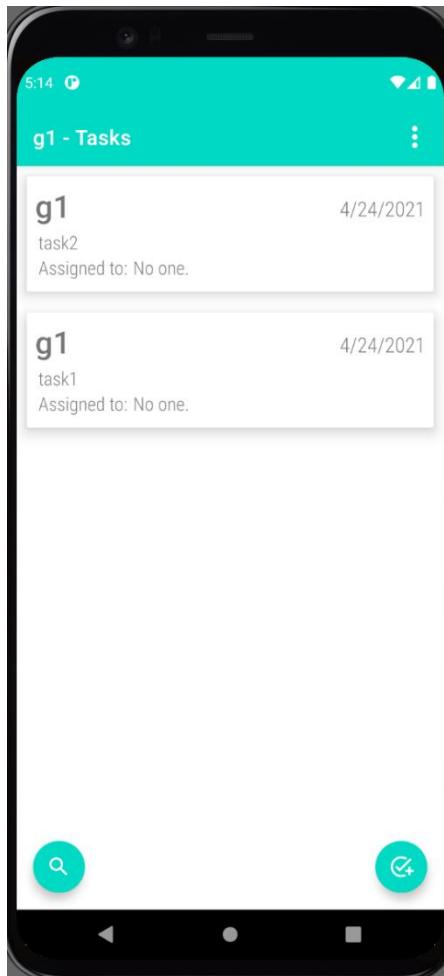


EUC2.1: Add Tasks

Precondition: This use case requires that the user is logged in and is an admin of the group.

Actor: Group Admin	System: ShareList/ManageTask
<p>1. TUCBW group admin clicks on the Add Task button.</p> <p>3. The group admin enters the information and clicks on the Save button.</p> <p>5. TUCEW the group admin sees a confirmation “The task has been added” on the app</p>	<p>0. System displays the Add task button.</p> <p>2. System prompts the group admin to enter the Task name, Task Description and Due date.</p> <p>*4. System checks for errors and processes the save request.</p>

Postcondition: The task is stored in the database.

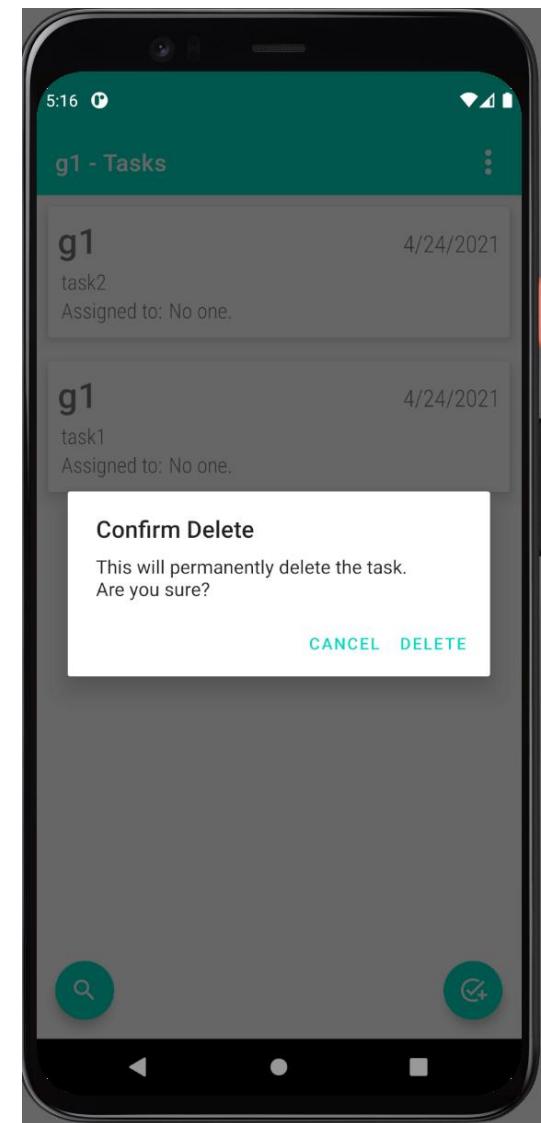
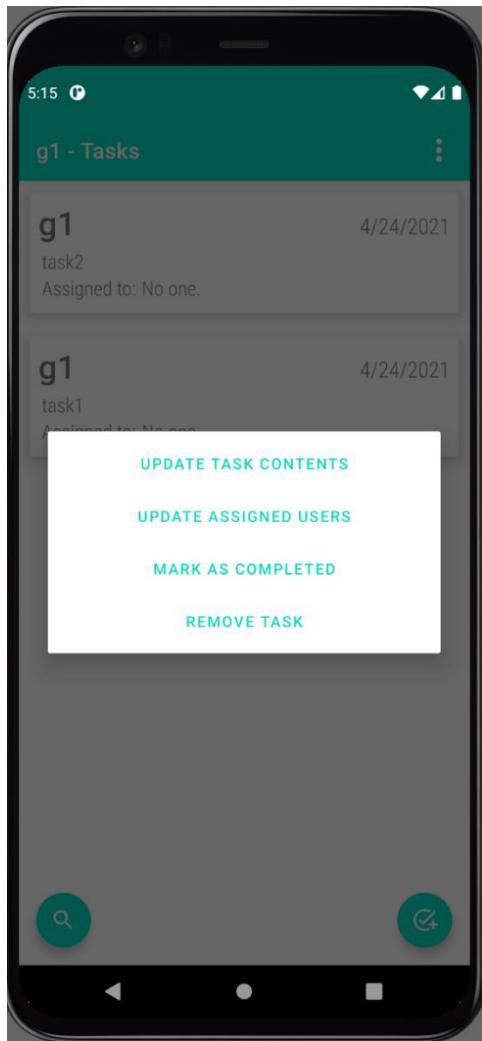


EUC2.2: Remove Tasks

Precondition: This use case requires that the user is an admin of the group and the task has been created.

Actor: Group Admin	System: ShareList/ManageTask
1. TUCBW group admin clicks on the Remove task button.	0. System displays the Remove task button.
3. The group admin confirms by clicking the "Confirm" button.	2. System prompts the group admin to confirm the remove task activity.
5. TUCEW the group admin sees a confirmation of the task being removed.	*4. System removes the task.

Postcondition: The task is removed in the database.

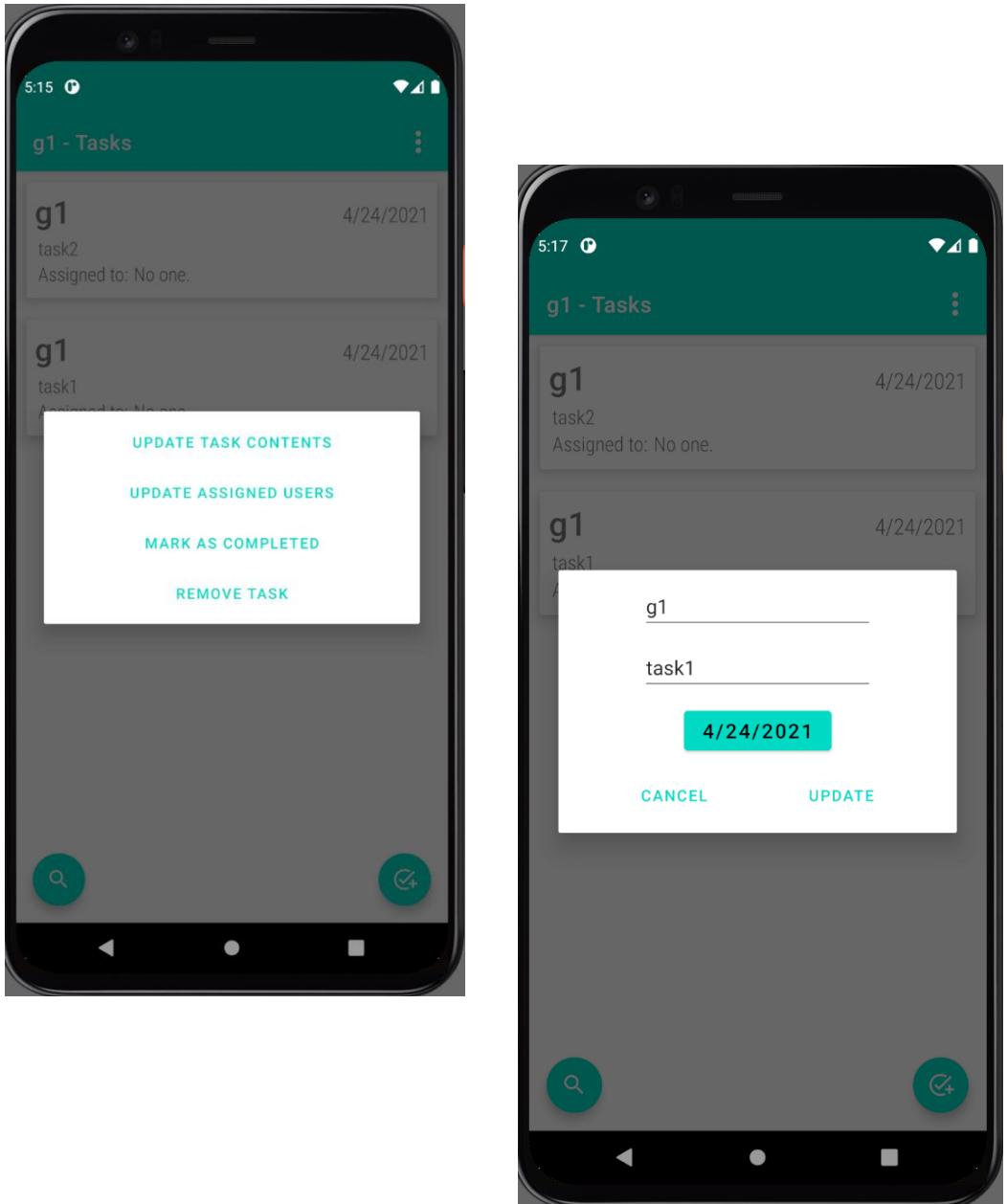


EUC2.3: Edit Tasks

Precondition: This use case requires that the task is already created.

Actor: Group Admin	System: ShareList/ManageTask
1. TUCBW group admin clicks on the 'Update Task Contents' button.	0. System displays the Update Task Contents button. *2. System provides the group admin the ability to edit name, description, due date and displays a message "The task has been updated".
3. TUCEW group admin sees the changes.	

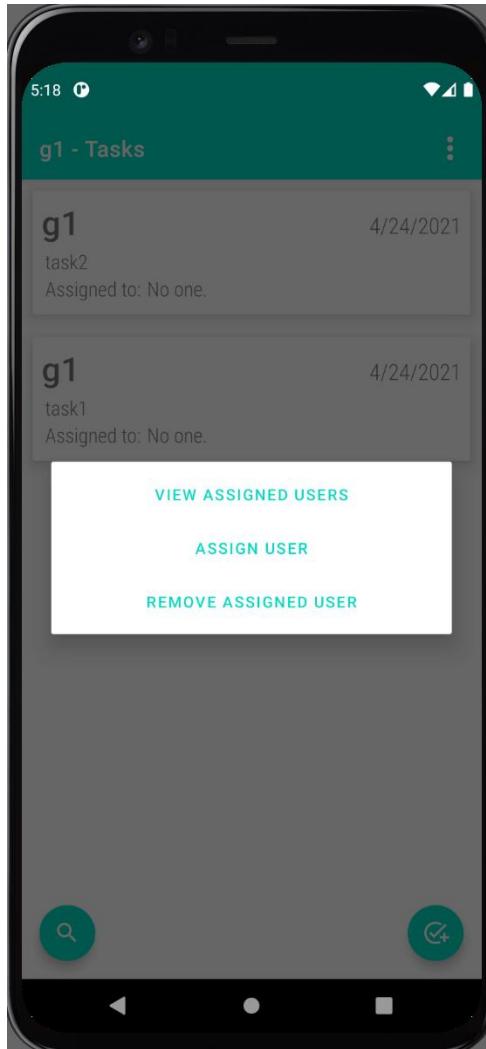
Postcondition: The task is edited successfully and updated in the database.



EUC2.4: Assign Tasks

Precondition: This use case requires that the user is an admin of the group and the task is created.

Actor: Group Admin	System: ShareList/ManageTask
1. TUCBW the group admin clicks on the Assign User button.	0. The system displays options: View Assigned Users, Assign User and Remove Assigned Users
3. The group admin selects the user to assign.	2. System allows the group admin to choose the user to assign the task. *4. System assigns the selected user.
5. TUCEW group admin sees the user is assigned to the task.	
Postcondition: The task is assigned to the respective user successfully.	

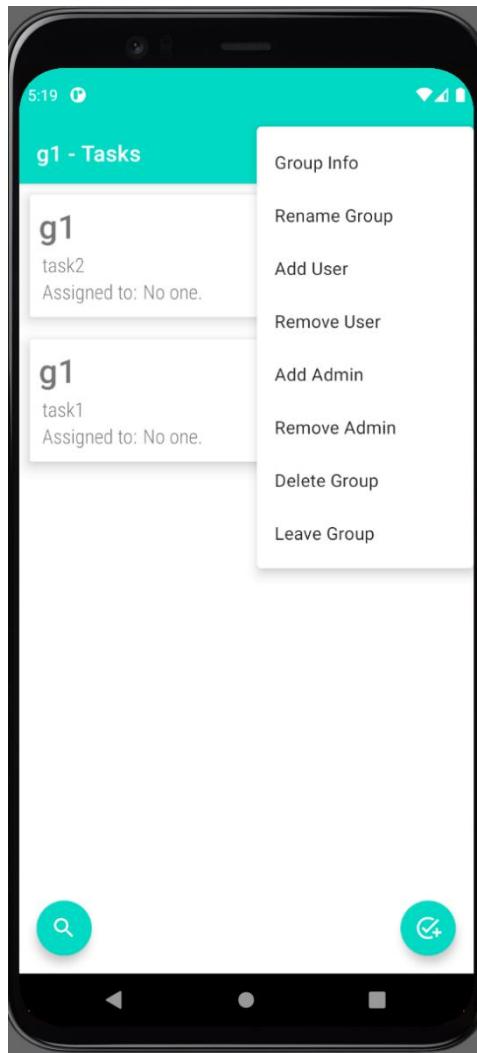


EUC3.1: Add Users

Precondition: This use case requires that the user is an admin of the group.

Actor: Group Admin	System: ShareList/ManageGroup
1. TUCBW admin clicks on the 'Add User' button.	0. System displays the Add User button.
3. Group admin selects the user to be added.	2. System prompts admin to enter the user's email to be added.
5. TUCEW group admin sees the user added to the group.	*4. System adds the user and displays a message "User has been added."

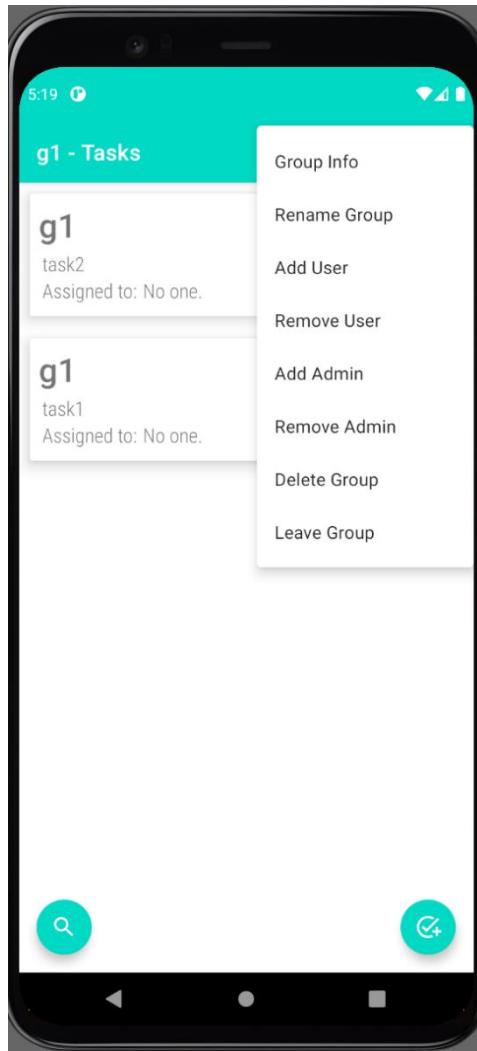
Postcondition: The user is successfully added to the respective group.



EUC3.2: Remove users

Precondition: This use case requires that the user is an admin of the group.

Actor: Group Admin	System: ShareList/ManageGroup
1. TUCBW admin clicks on the Remove User button.	0. System displays the Remove User button. *2. System prompts admin to enter the user's email to be removed.
3. Group admin selects the user to be removed.	4. System displays message to confirm the removal of the user.
5. TUCEW group admin sees the user is no longer a member of the group.	
Postcondition: The selected user is successfully removed from the respective group.	

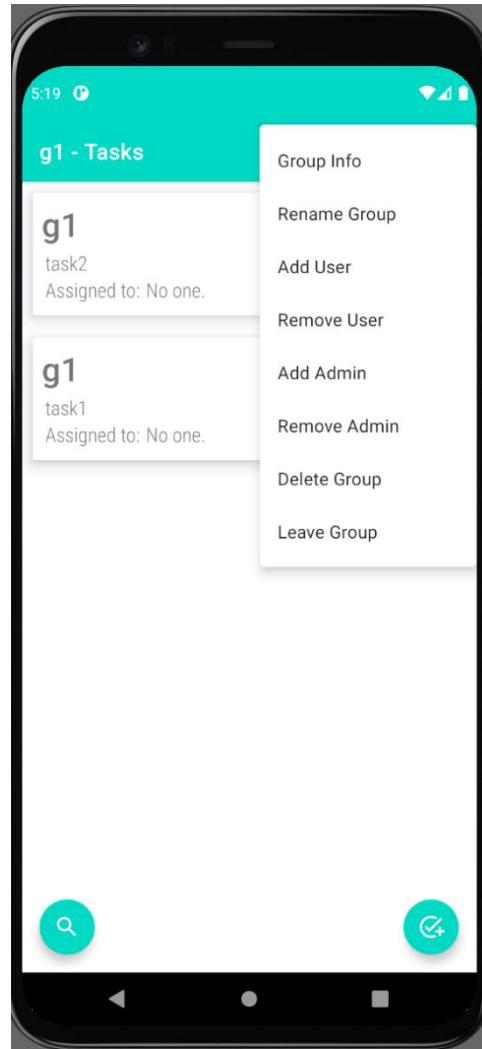


EUC3.3: Add Admins

Precondition: This use case requires that the user is an admin of the group.

Actor: Group Admin	System: ShareList/ManageGroup
1. TUCBW group admin clicks on the 'Add Admin' button.	0. System displays the Add Admin button. *2. System prompts admin to select another user to be added as an admin.
3. TUCEW group admin sees the user is now an admin.	

Postcondition: The user is successfully added as the admin of the respective group, and the database is updated to reflect this change.

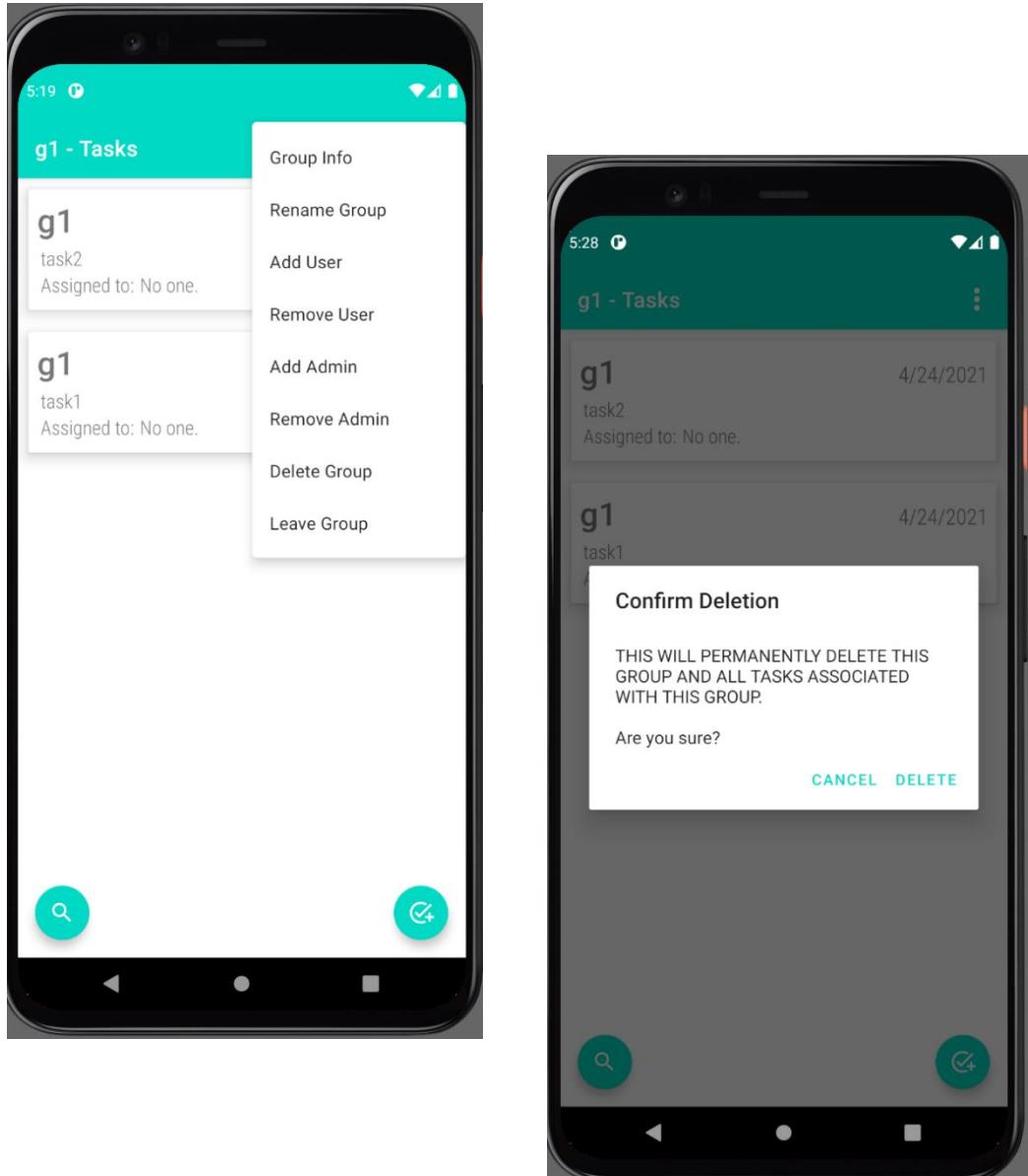


EUC3.4: Delete Group

Precondition: This use case requires that the user is an admin of the group and the group is created.

Actor: Group Admin	System: ShareList/ManageGroup
	0. System displays the Delete group button.
1. TUCBW group admin clicks on the Delete group button.	2. System asks the group admin for confirmation.
3. Group admin selects Delete.	
5. TUCEW the group, related tasks and users, being removed from the records.	*4. System deletes the group.

Postcondition: The group is deleted successfully, and the entry is removed from the database.

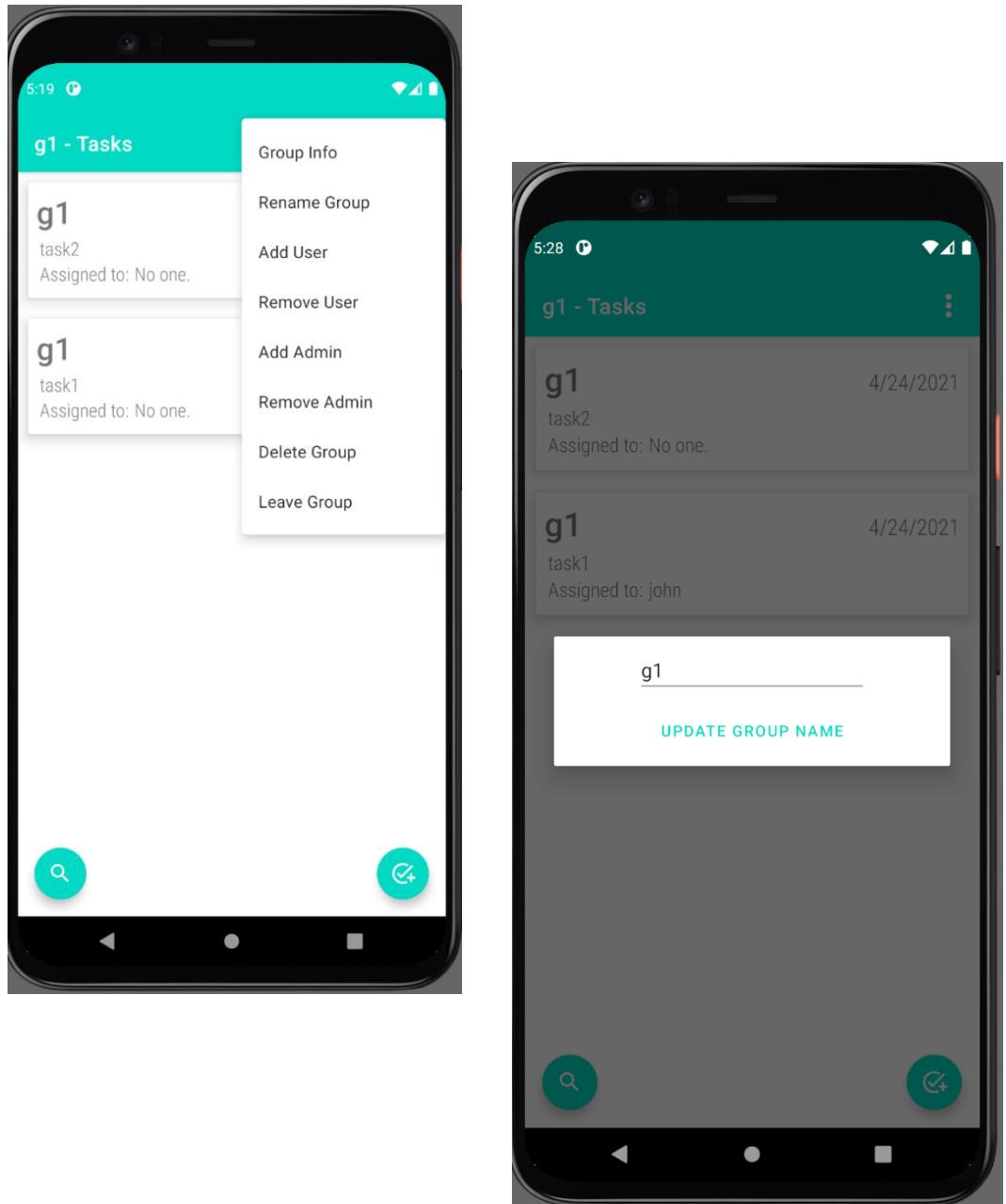


EUC3.5: Edit Group

Precondition: This use case requires that the user is an admin of the group and the group is created.

Actor: Group Admin	System: ShareList/ManageGroup
1. TUCBW admin clicks on the 'Rename Group' button.	0. System displays the Rename Group button. *2. System provides an option to edit the group name.
3. TUCEW group admin sees that the group name is updated.	

Postcondition: The group name is updated successfully, and database reflects these changes.

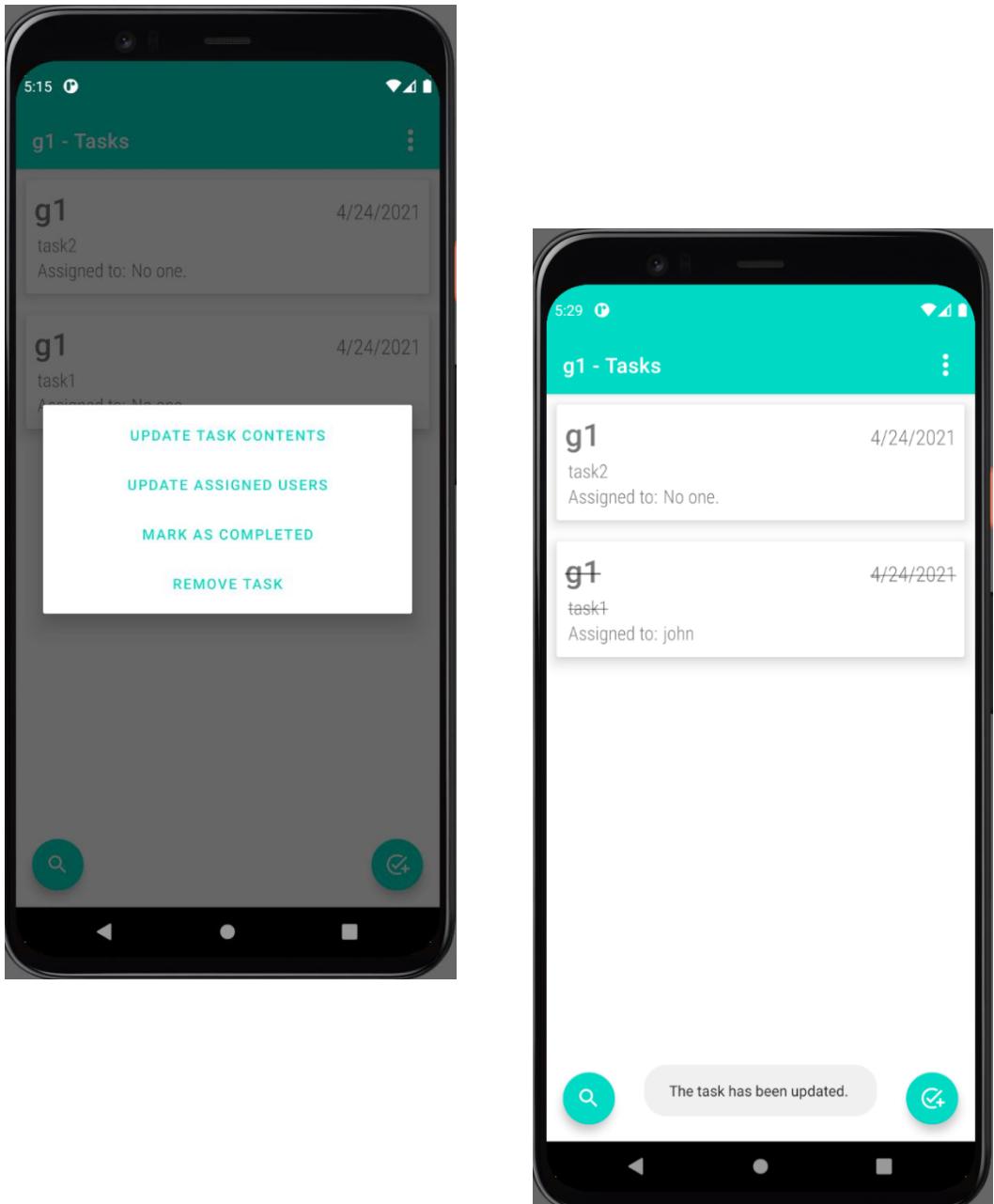


EUC4: Mark Task as Completed

Precondition: This use case requires that the user is a member of the group and the task is not marked as complete.

Actor: User	System: ShareList/ManageTask
1. TUCBW user clicks on the 'Mark Task as Completed' button.	0. System displays the 'Mark Task as Completed' button. *2. System displays a message "The task has been updated".
3. TUCEW the user sees the task as crossed off.	

Postcondition: The task is marked as completed, and its status updated in the database.

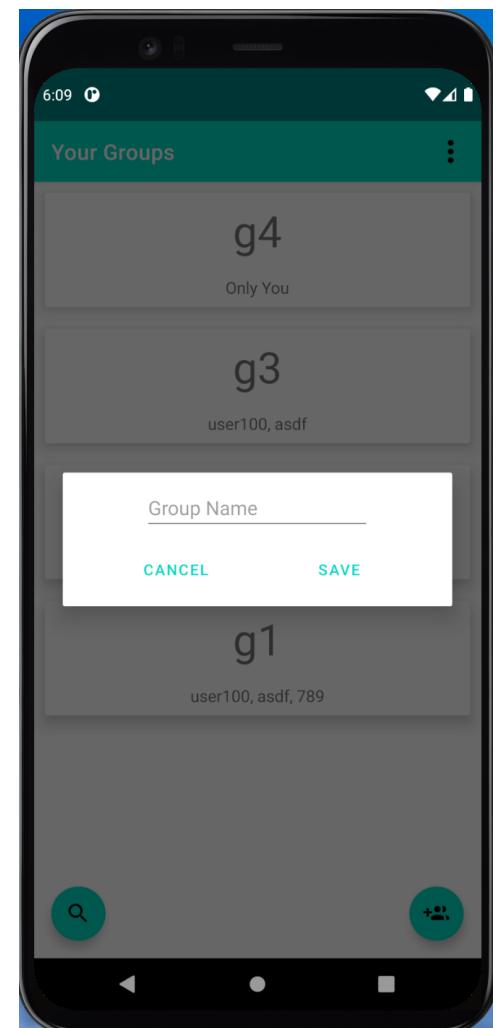
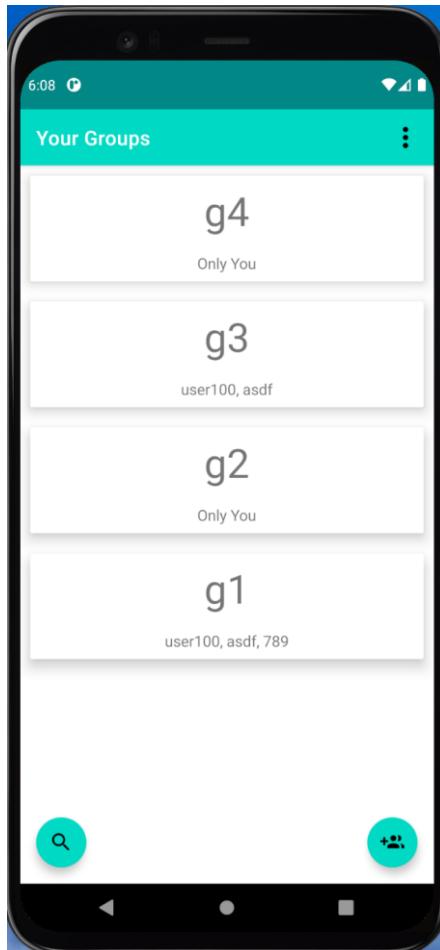


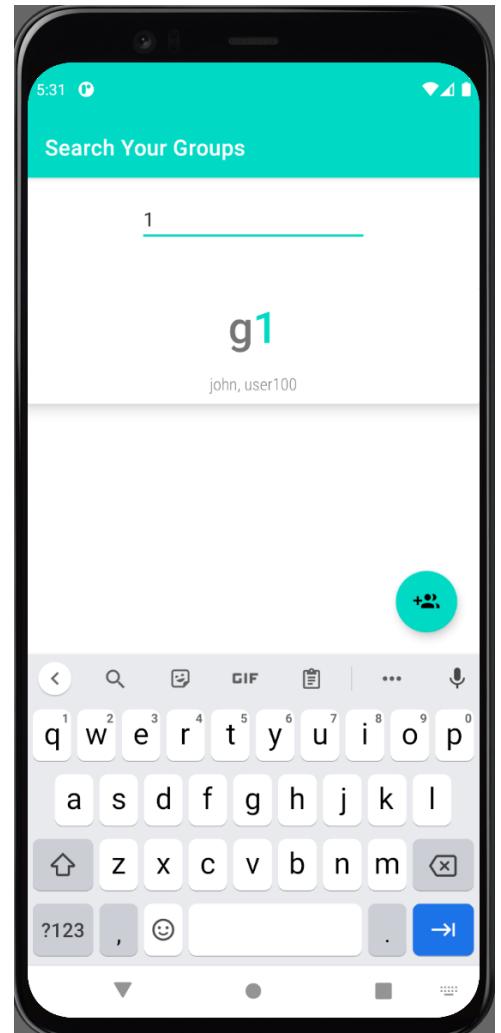
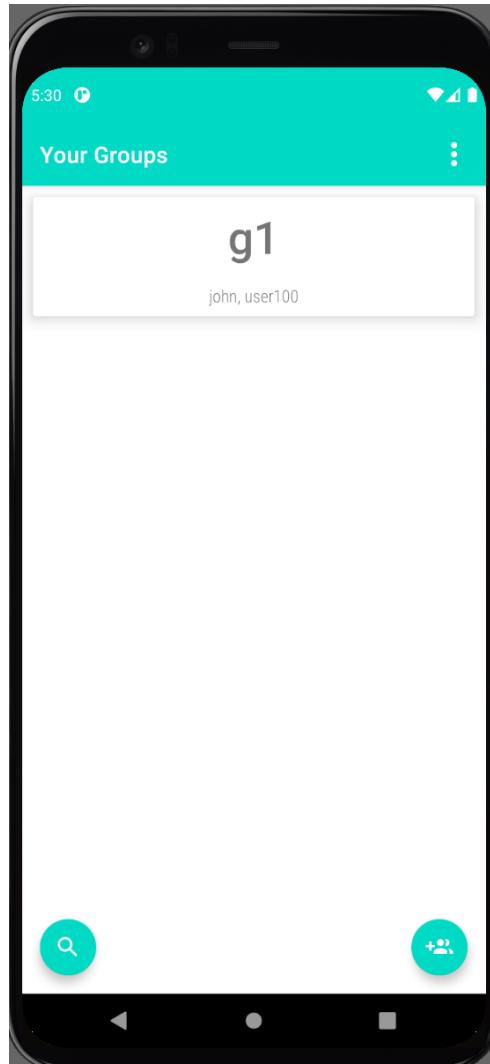
EUC5: Create Group

Precondition: This use case requires that the user is logged in to the ShareList App and is seeing the 'Create Group' activity.

Actor: User	System: ShareList App
1. TUCBW user clicks on the 'Create Group' button.	0. System displays the Create Group activity.
3. The user fills in the text field and clicks the save button.	2. System displays text field to enter group name.
5. TUCEW the user sees a confirmation "The group has been added".	*4. System checks for errors and processes the save request.

Postcondition: The group is created, stored in the database, and available to be acted upon immediately.





EUC6: Search by Keywords

Precondition: This use case requires that the user is logged into the app.

Actor: User	System: ShareList App
1. TUCBW user clicks on the Search button and enters the search keyword.	0. System displays the Search button. 2. System displays search the result to the user
3. TUCEW user sees the correct search results.	

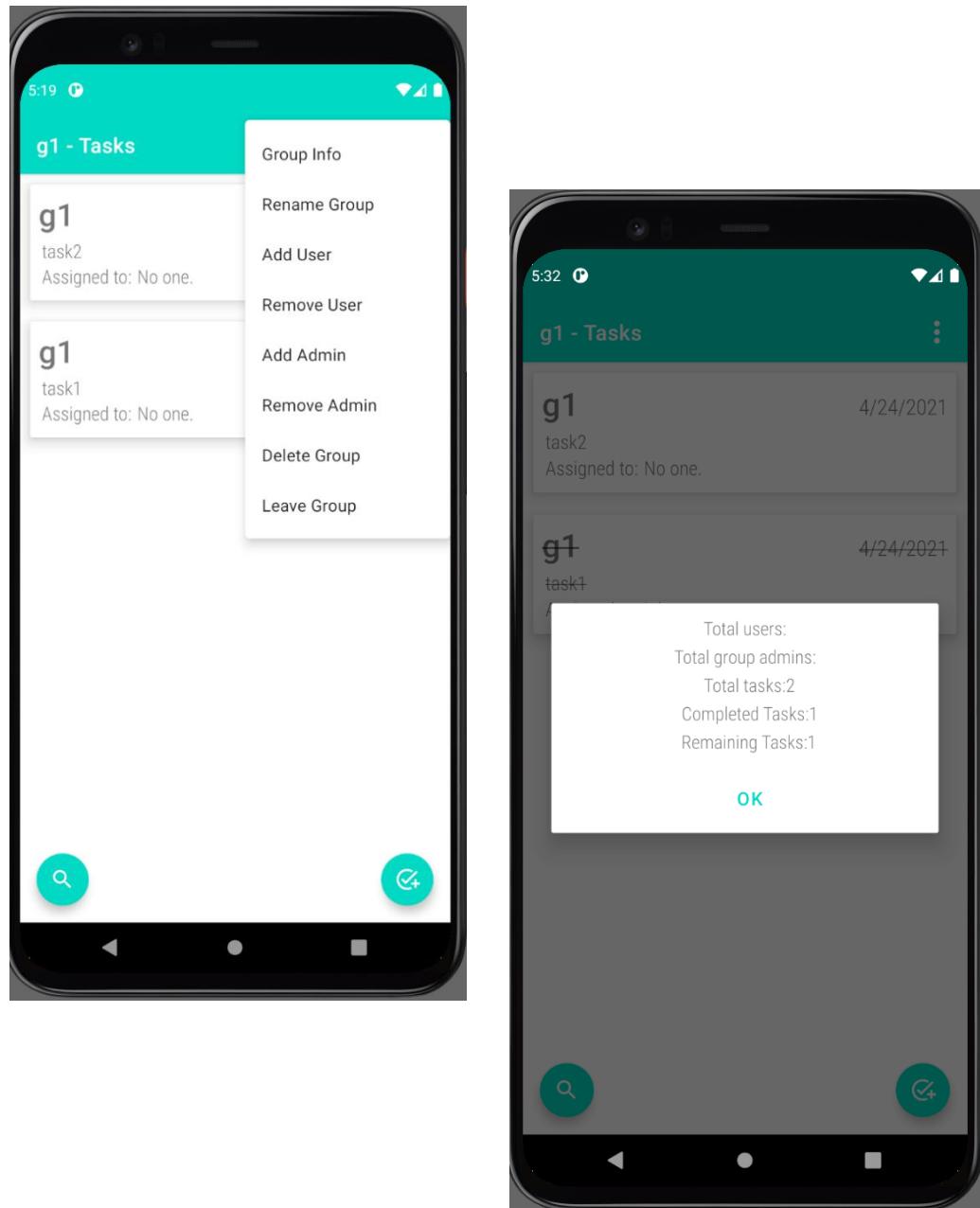
Postcondition: The search result is available to be acted upon.

EUC7: Generate Summary

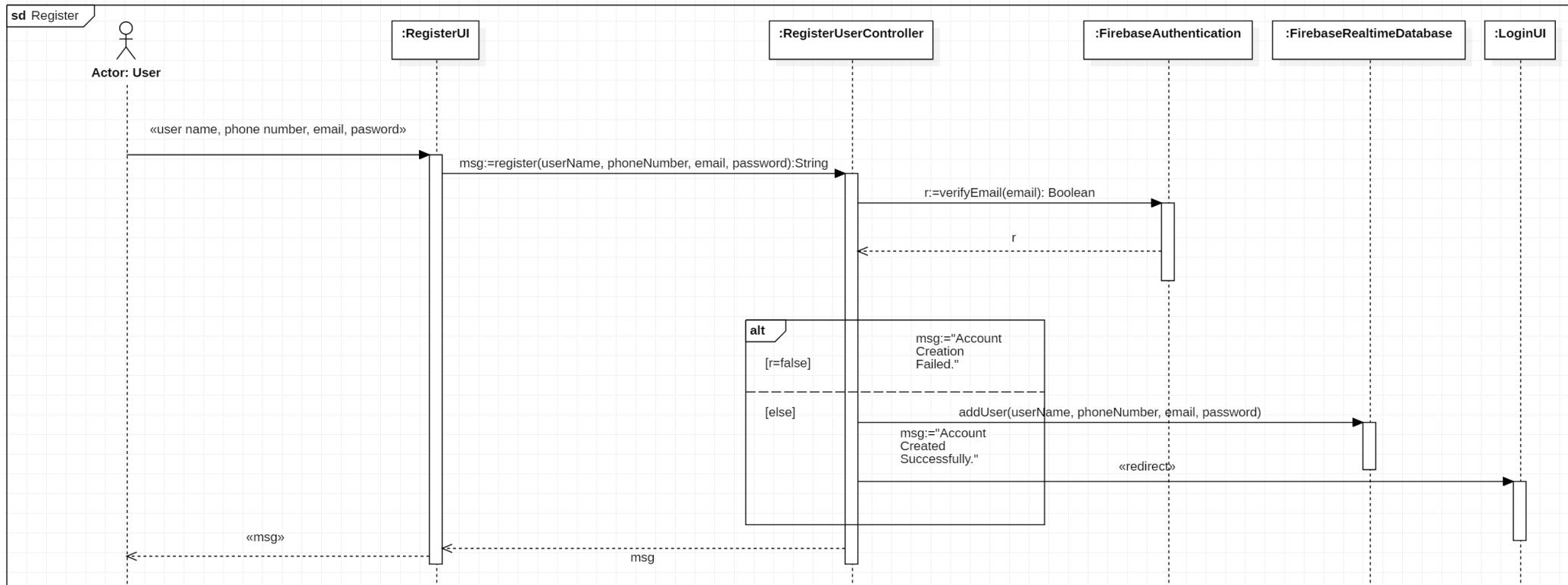
Precondition: This use case requires that the user is logged in and is a member of the particular group.

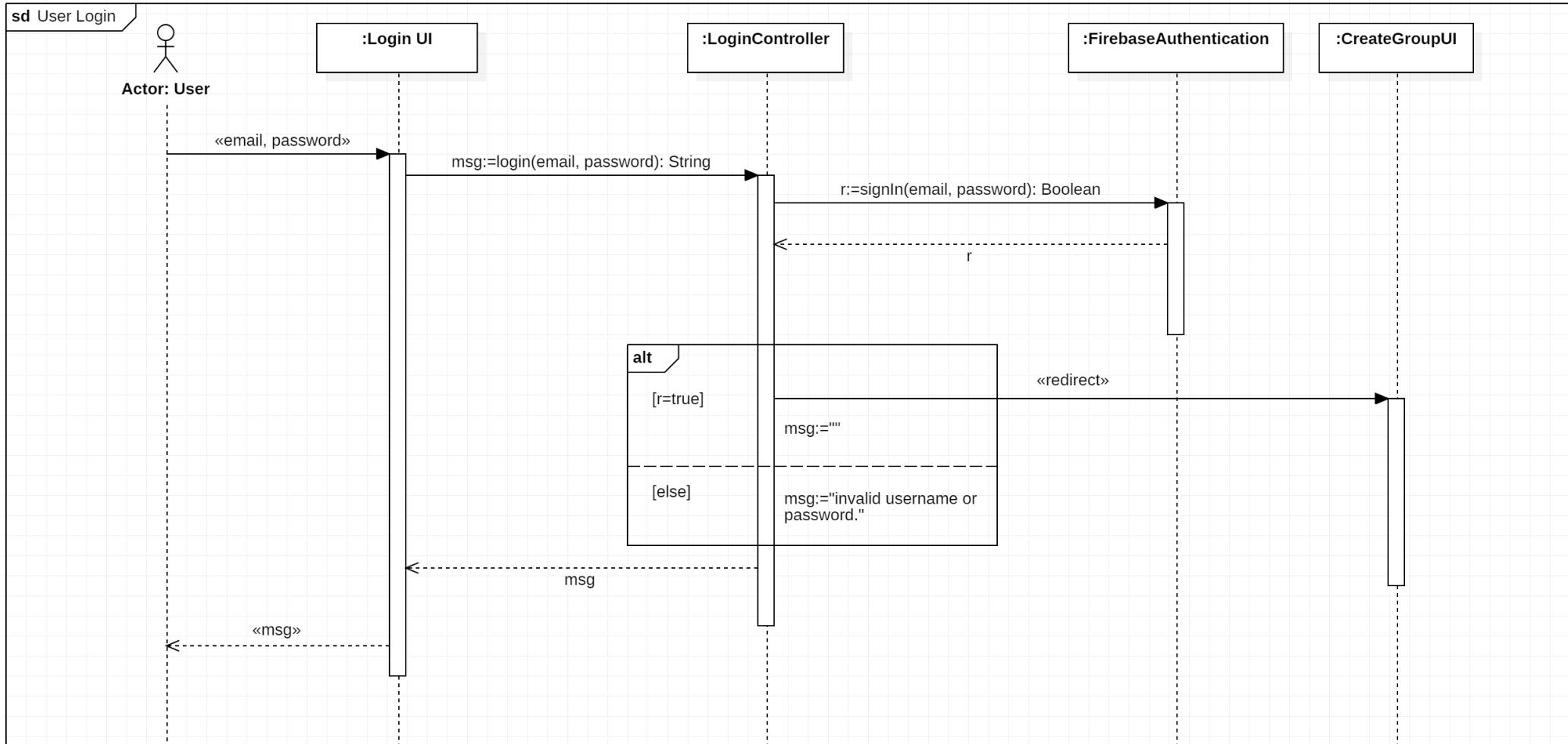
Actor: User	System: ShareList App
1. TUCBW user clicks on the 'Group Info' button.	0. System displays the 'Group Info' button. 2. System displays the summary of the tasks.
3. TUCEW the summary of group is displayed to the user.	

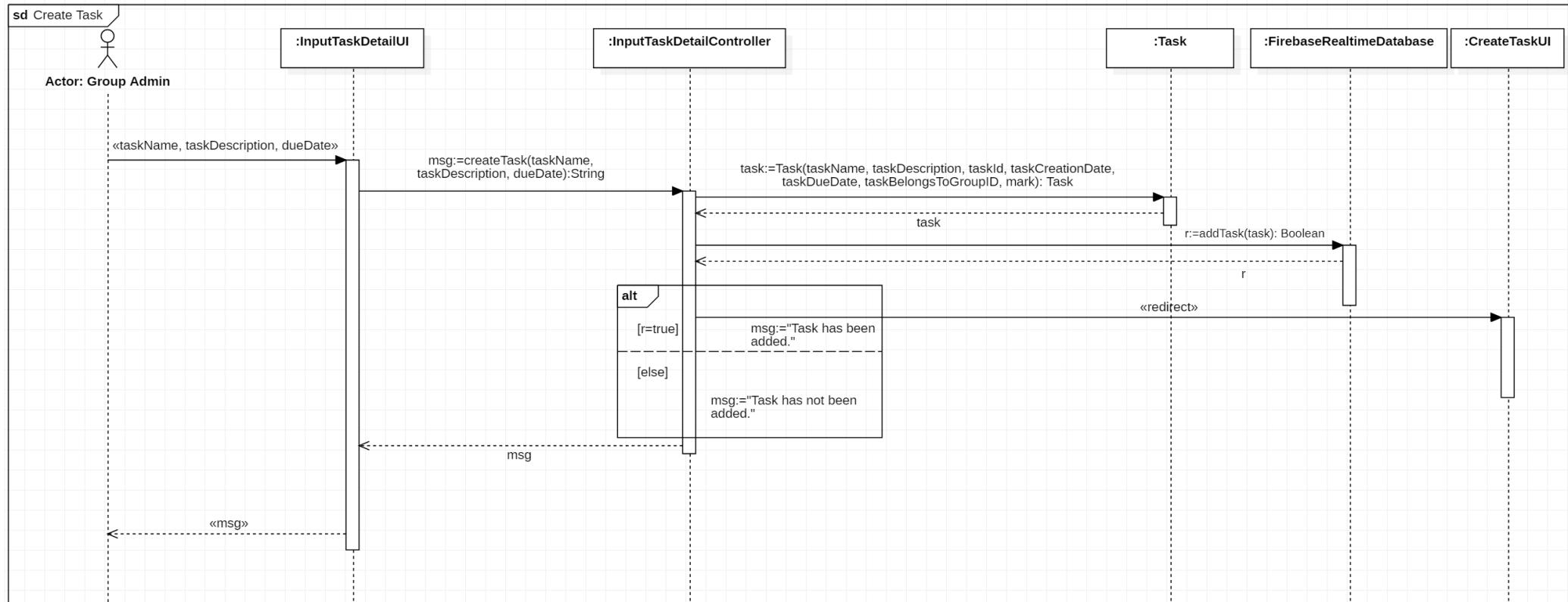
Postcondition: The Summary of the group is displayed to the user.

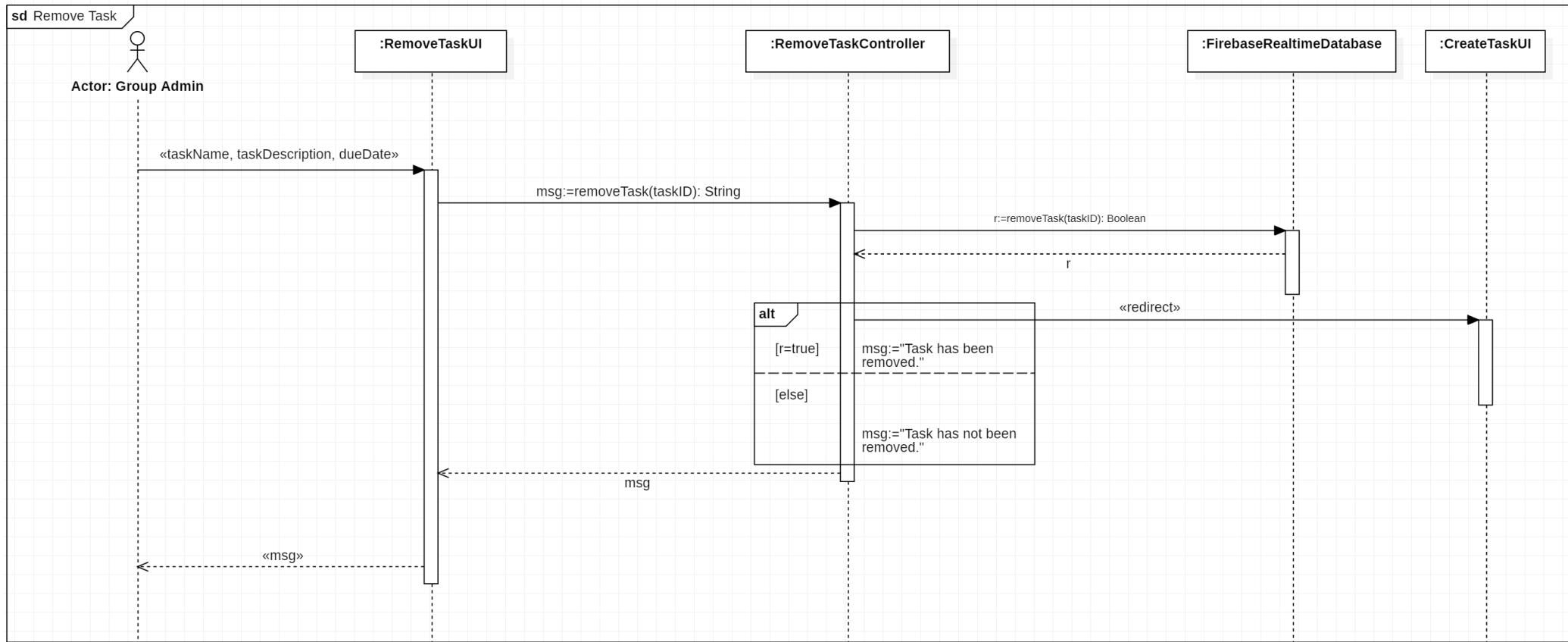


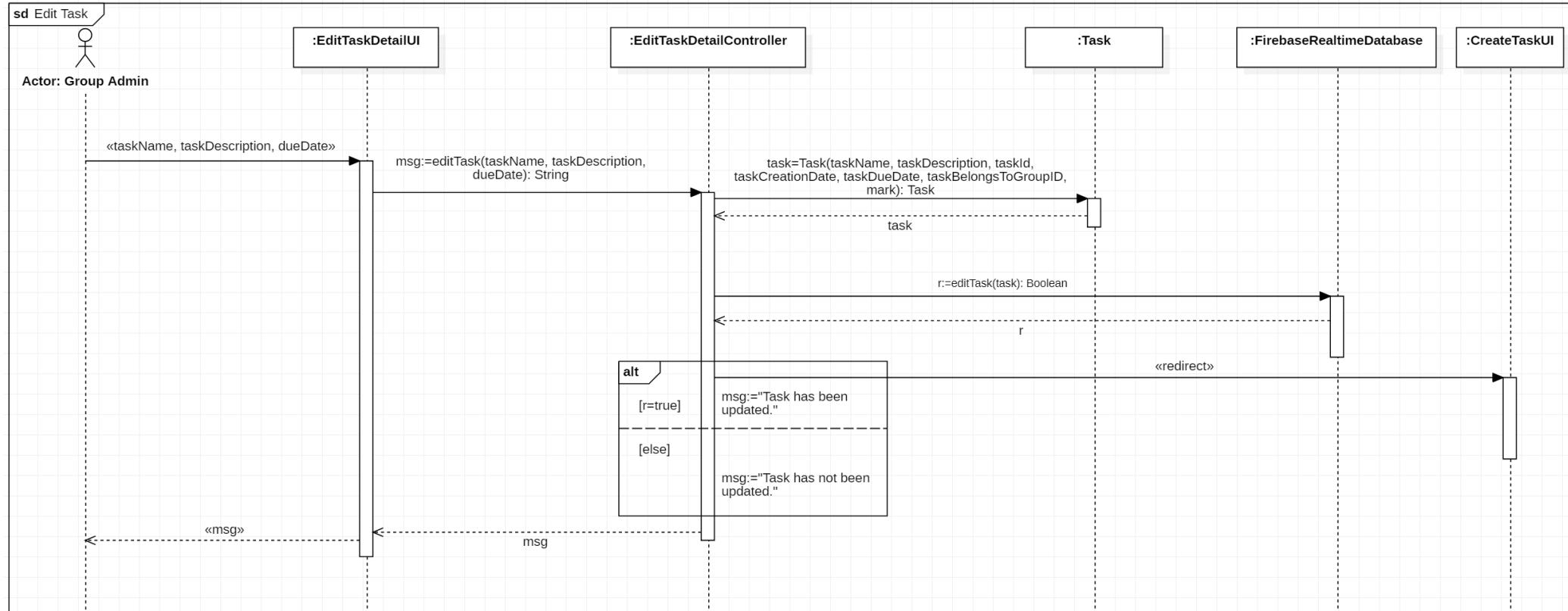
Design Sequence Diagrams

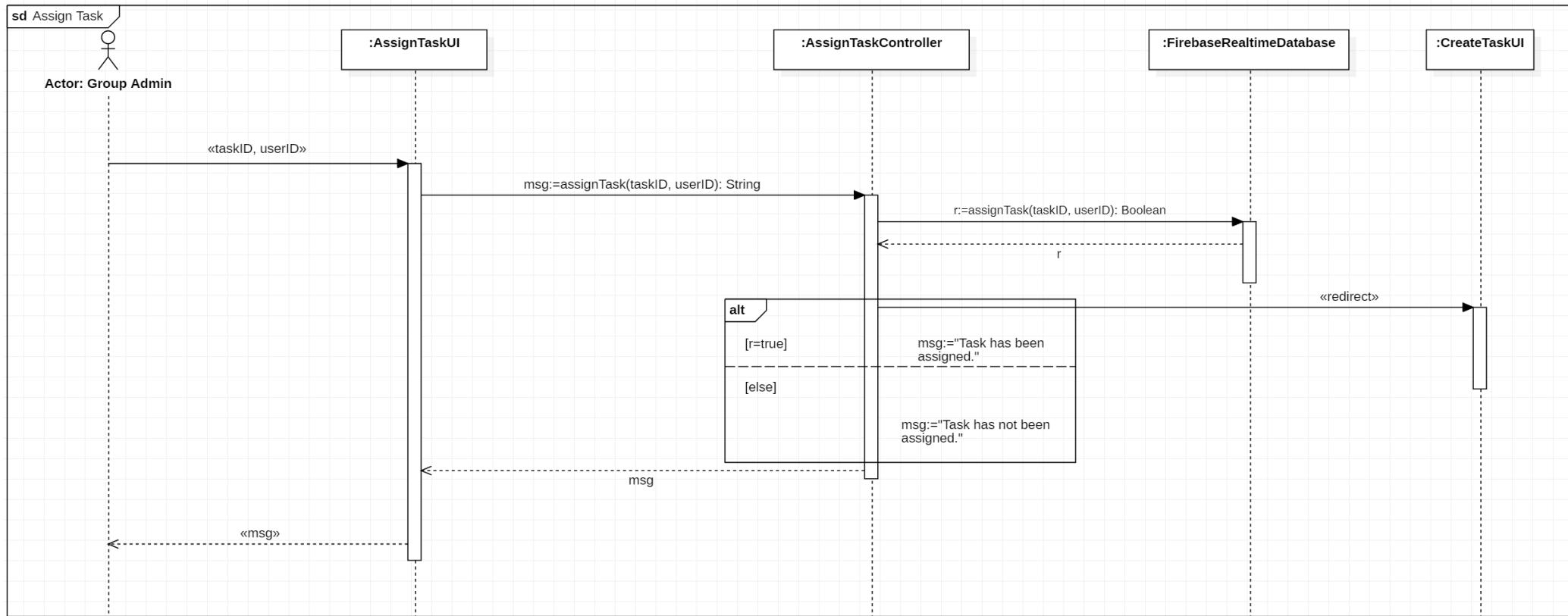


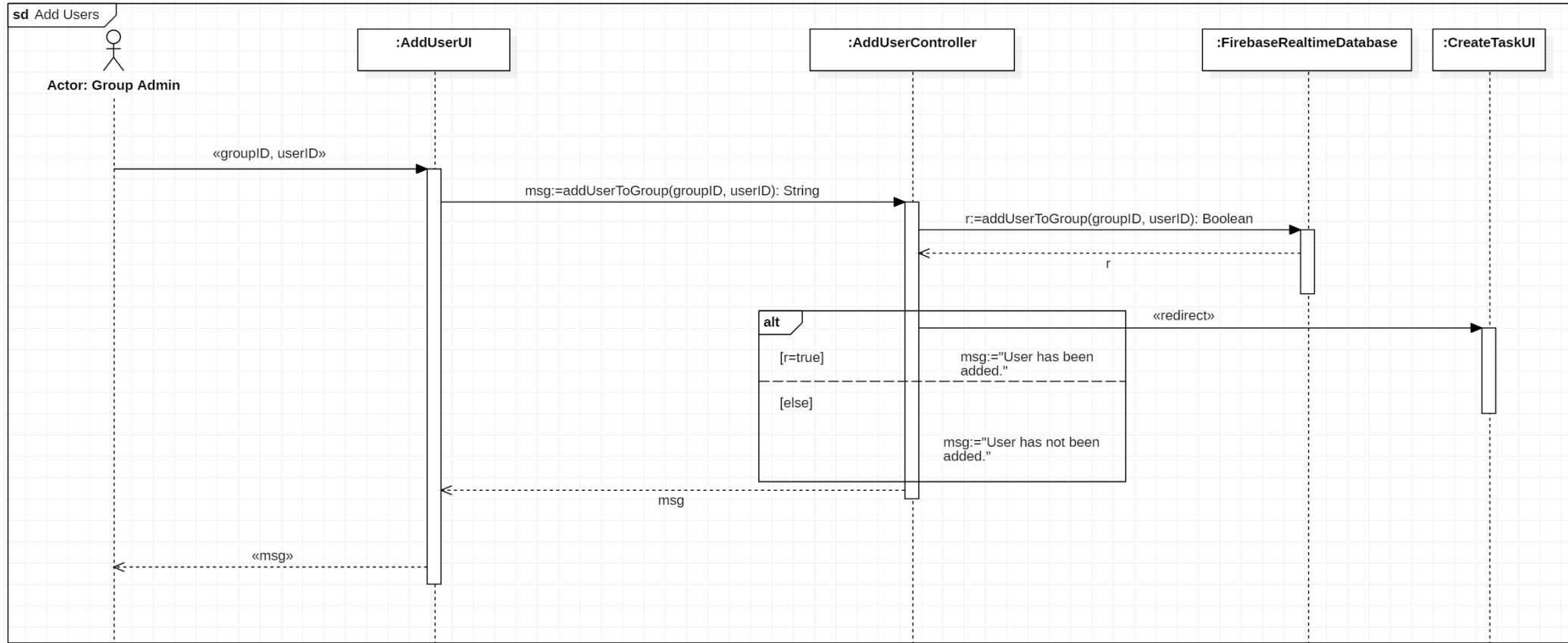


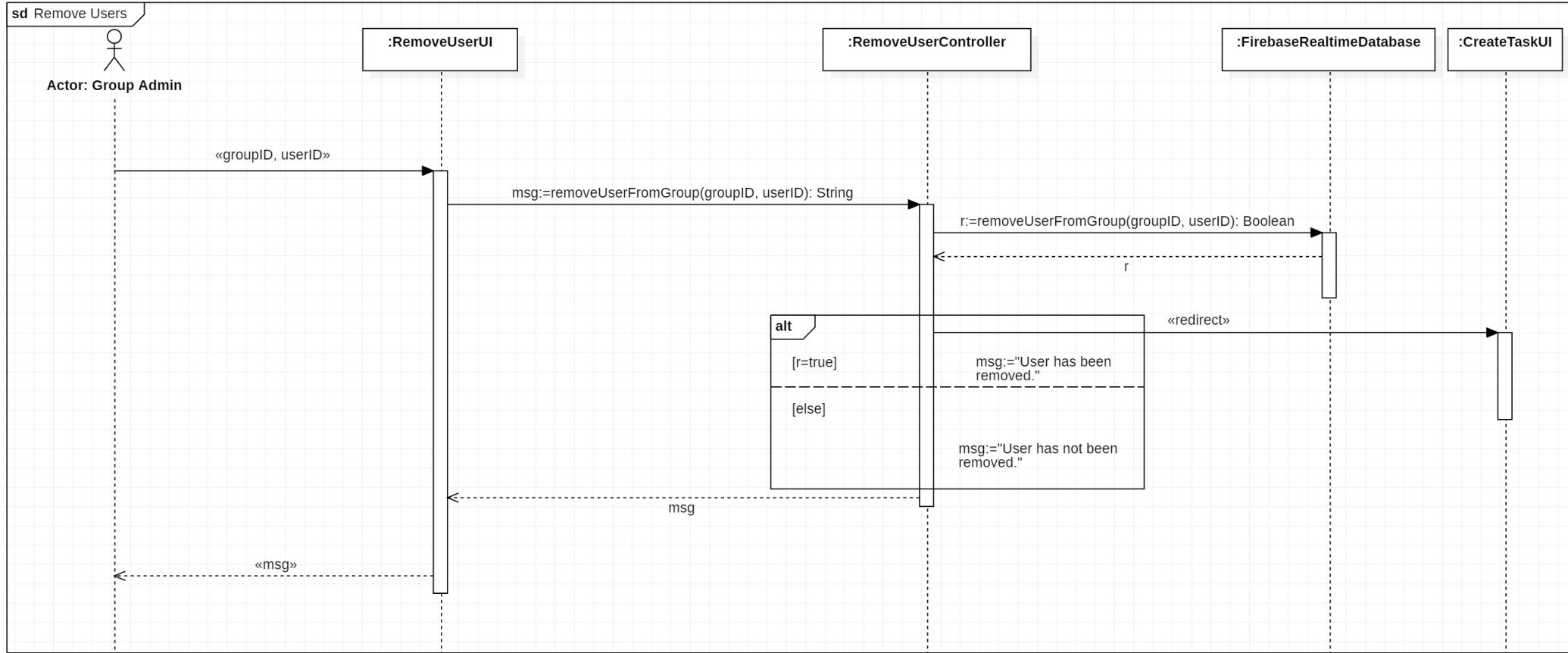


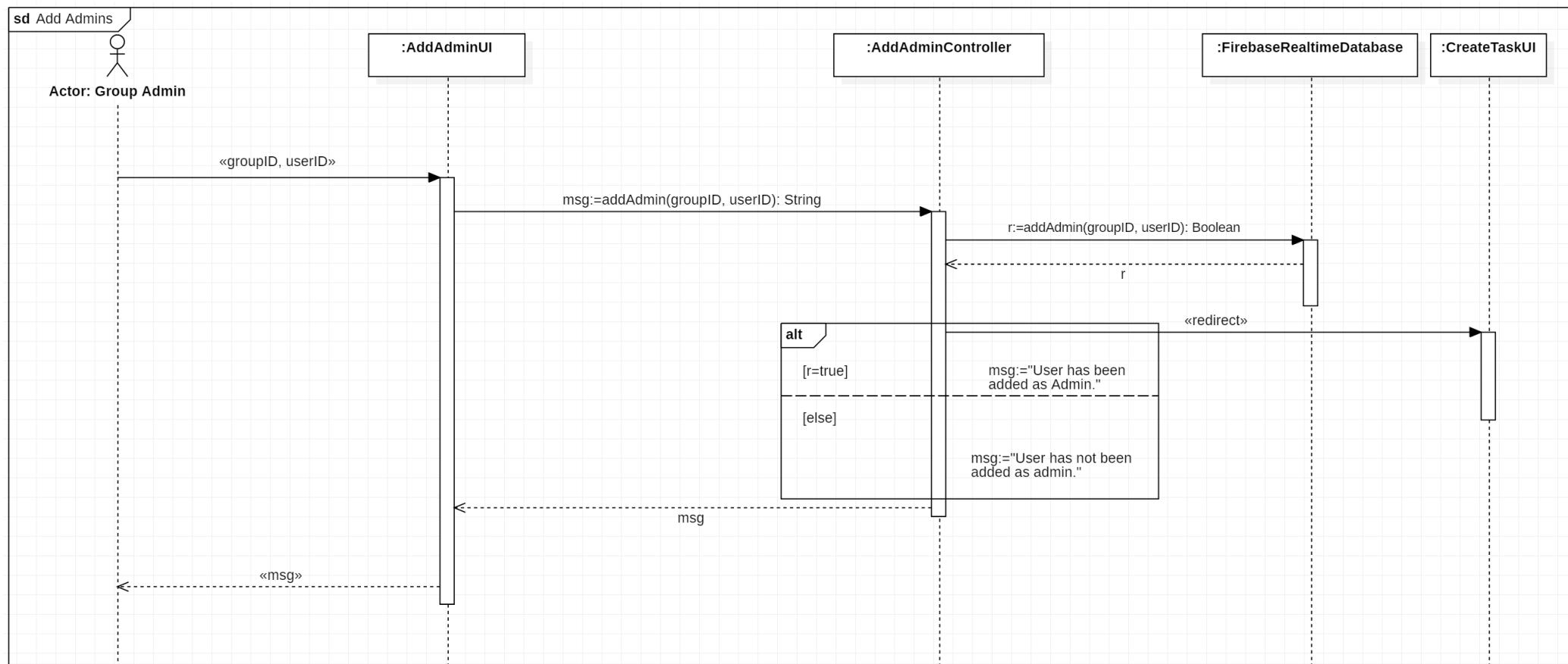


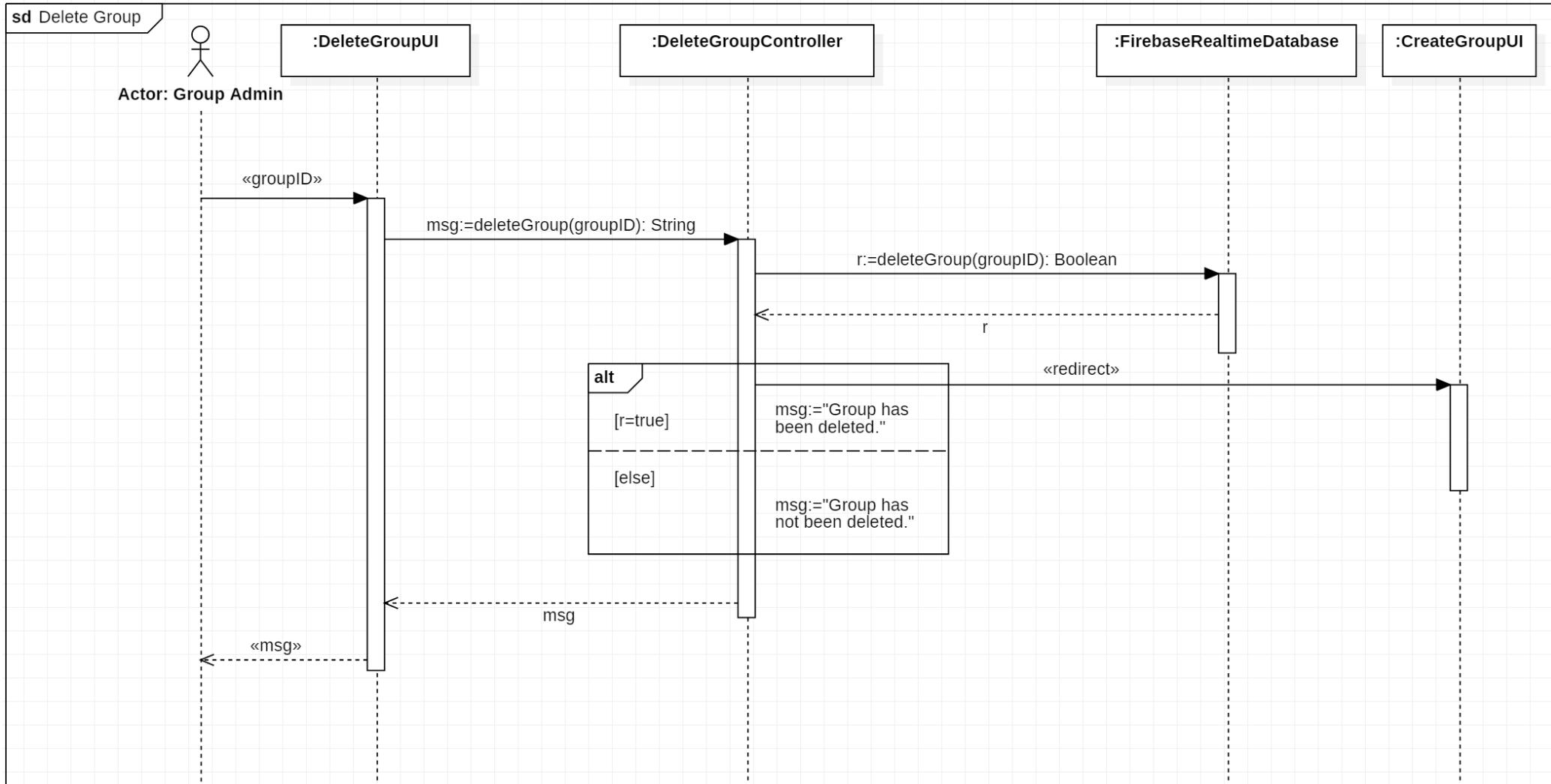


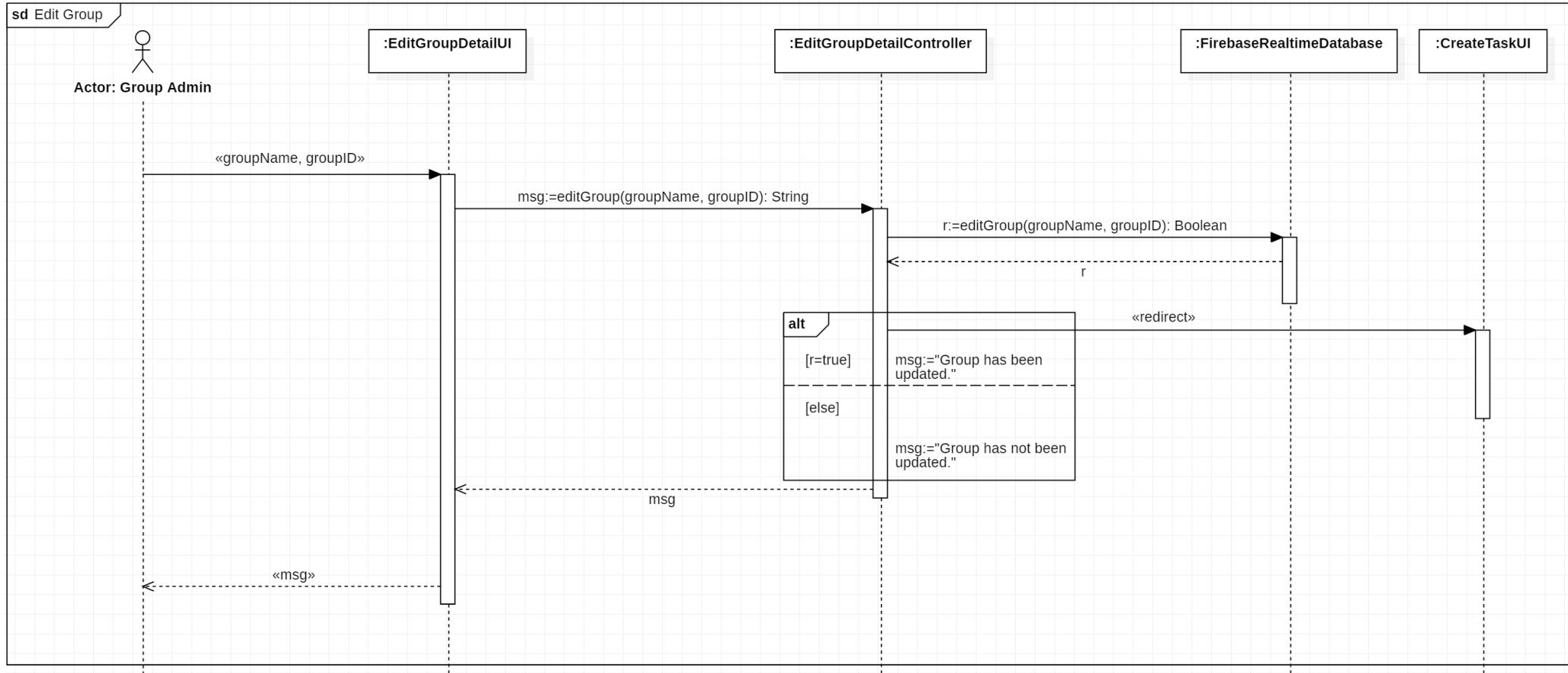


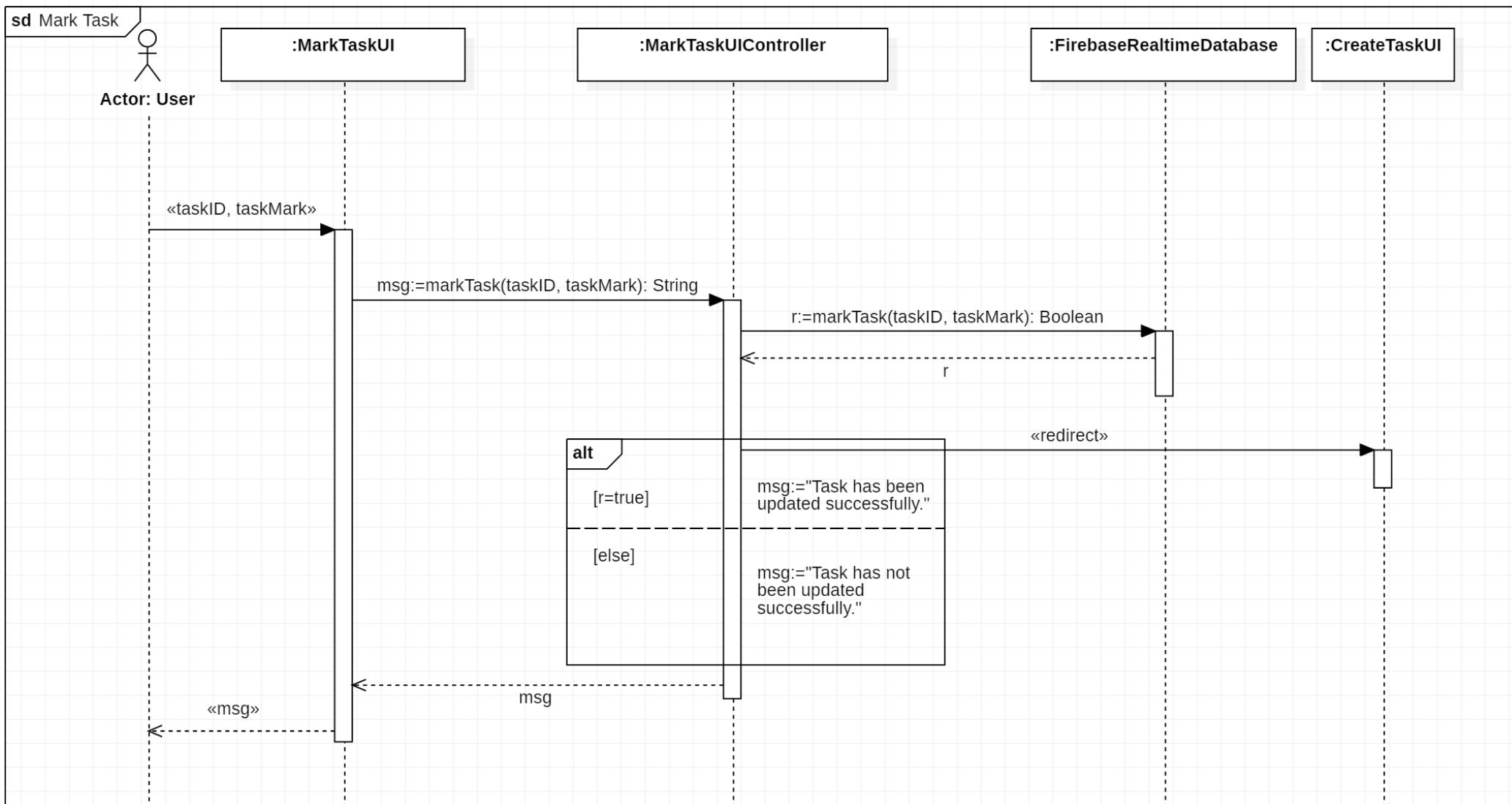


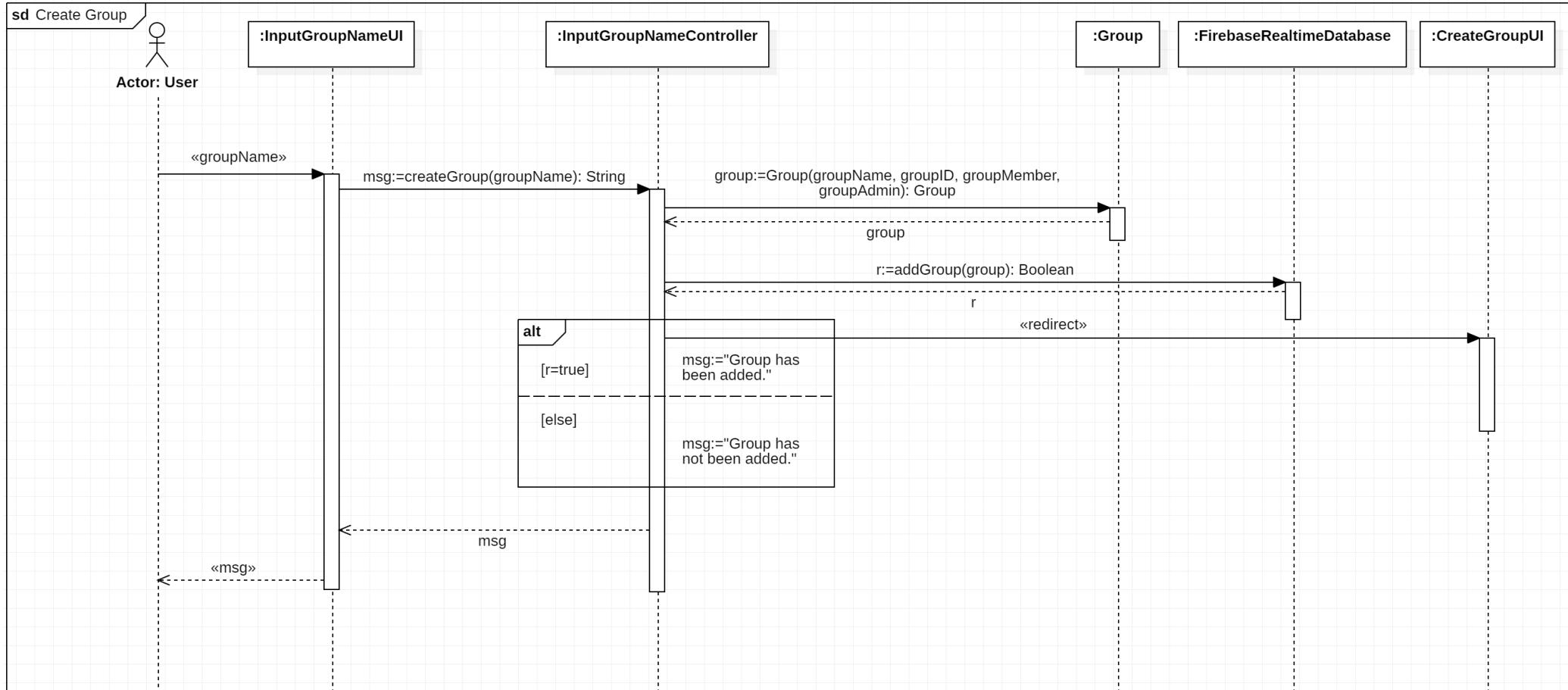




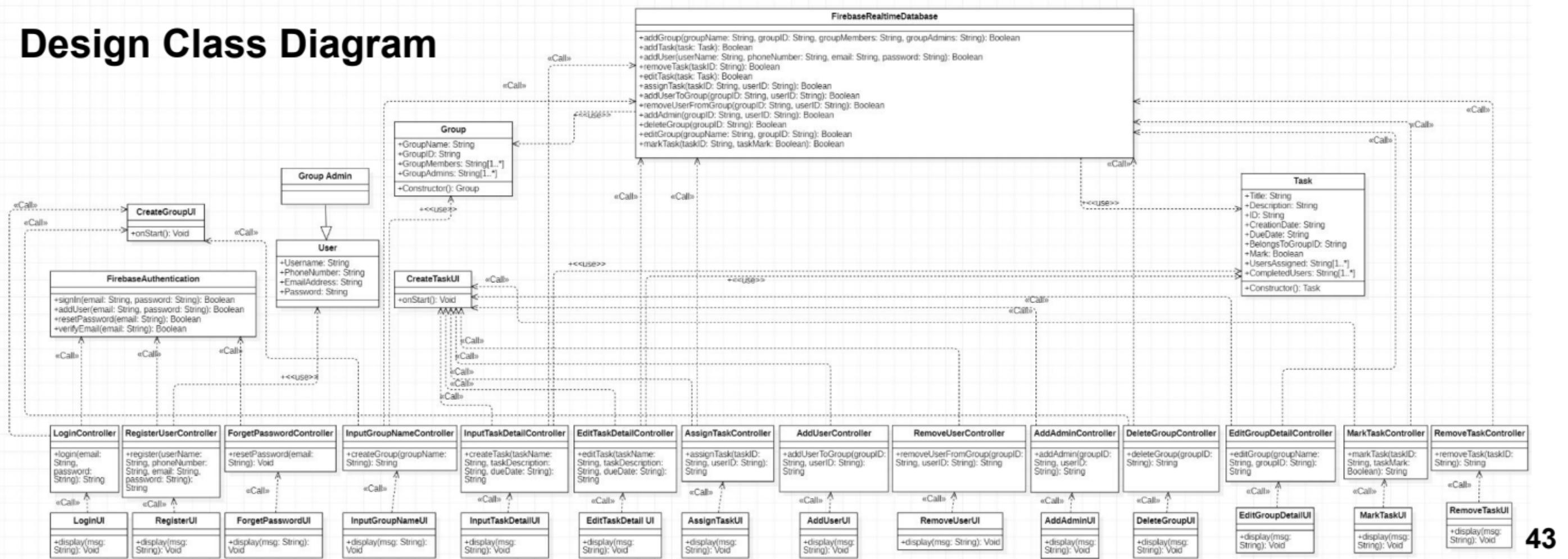








Design Class Diagram



Task List

Assigned to...	Iteration 2	Iteration 3	Use Case
Mohammed & Anum	Add tasks.		2.1
		Remove tasks.	2.2
Yanzhi & Yu	Register.		1.1
	Login.		1.2
Mohammed & Anum	Manage task.		2
	Edit tasks.		2.3
Hannan	Mark task as completed.		4
	Create groups.		5
	Search by keywords.		6
		Generate summary.	7
Yanzhi & Yu	Log Off.		1.3
		Reset password.	1.4
Mohammed & Anum	Assign tasks.		2.4
	Manage group.		3
	Add users.		3.1
	Remove users.		3.2
		Add admins.	3.3
		Delete groups.	3.4
		Edit groups.	3.5