

Full Project Documentation

(For Project#17)

Made by:

Name	ID
احمد علاء الدين السيد مرتضى	201900065
احمد محمد ابراهيم محمد الدقنس	201900083
احمد عيسى محمود احمد	201900074
احمد عبد المنجي عبد الموجود ابراهيم تليمه	201900059
احمد محمد محمد علي ابراهيم	201900095

TABLE OF CONTENTS

(click on to go or use Bookmarks)

SRS.....	3
USER REQ.....	3
<i>Functional Requirements</i>	3
<i>Non-Functional Requirements.....</i>	4
SYSTEM REQ.....	5
<i>Functional Requirements</i>	5
<i>Non-Functional Requirements.....</i>	6
USE CASE DIAGRAM.....	7
USE-CASES DESCRIPTION	8
ACTIVITY DIAGRAMS	13
DATABASE SPECIFICATION	16
ERD SCHEMA	16
TABLES	17
SYSTEM ARCHITECTURE.....	18
SEQUENCE DIAGRAMS.....	19
SEQUENCE DIAGRAMS	19
SYSTEM SEQUENCE DIAGRAMS (SSDs)	30
COLLABORATION/COMMUNICATION DIAGRAM(S).....	42
CLASS DIAGRAM.....	52
INITIAL VERSION	52
INTERMEDIATE VERSION.....	53
FINAL VERSION.....	54
PACKAGE DIAGRAM	55
APPLIED DESIGN PATTERN DESCRIPTION.....	56

User REQ :

Functional requirements:

1. The system shall allow the **admin** to add officers and assign them to a certain area.
2. The system shall allow the **admin** to manipulate the officer's data including the area.
3. The system shall allow the **admin** to get reports with parameters like state, age, gender, education, etc.
4. The system shall allow the **admin** to search in the report and sort the results in ascending or descending order for name , id ,age .
5. The system shall allow the **Users** to make request to add their family members with their details (name, dob, sex, email, phone, Education, Occupation, Address,Area, City, State) and the The system shall send that request to the officer which assigined to his/her area .
6. The system shall allow the **Users** to Upload their member's Identification Card / Birth Certificate / Document for each Member.
7. The system shall allow the **Users** to view their members and their details.

8. The system shall allow the **officer** to view the new request from the users in his area.
9. The system shall allow the **officer** to Validate or reject the request and ask the users in his/her area for corrections and view the corrected request.
10. The system shall allow the **User** view Request status and show the correction requests if exist .
11. The system shall allow the **officers** to login in their accounts with their Username and password.

Non-functional requirements:

1. The system has the ability to prevent the officers from access the user's data who lives in another area that is not assigned to him/her.
2. The system Guarantees the correctness of the date sent by the users through the officers .
3. The system prevents the users from accessing another user's data.
4. The system provides means for displaying reports to the admin in different ways.
5. The system provides means for communication between the officer and their users when errors occur.

system REQ :

Functional requirements:

1.1 / The system shall provide a form for admin for adding new officers and adding their data including the area their work on by providing appropriate input fields for every attribute.

2.1/ The system shall provide a form for viewing the officer's data .

2.2/ The system shall allow the admin to manipulate the officer data including the area by providing properate input fields for every attribute .

3.1/ The system shall provide a tabular form for showing the data of the reports taken form the database.

3.2/ The system shall allow the admin to choose which parameter from the report to view through an options list.

4.1/ The system shall provide a text field to type in it the parameter the admin want to find in the report.

4.2/ The system shall allow the admin to choose which way to view the sorted report (ascending or descending) and which parameter to search by through an options list.

5.1/ The system shall provide the form for the user to add the member by providing appropriate input fields for every attribute.

10. The system shall allow the **User** view Request status and show the correction requests if exist.

6.1/ The system shall provide a file input field for adding pdf, images in the form referred to in point (5.1).

7.1/ The system shall provide a form for viewing the user's data.

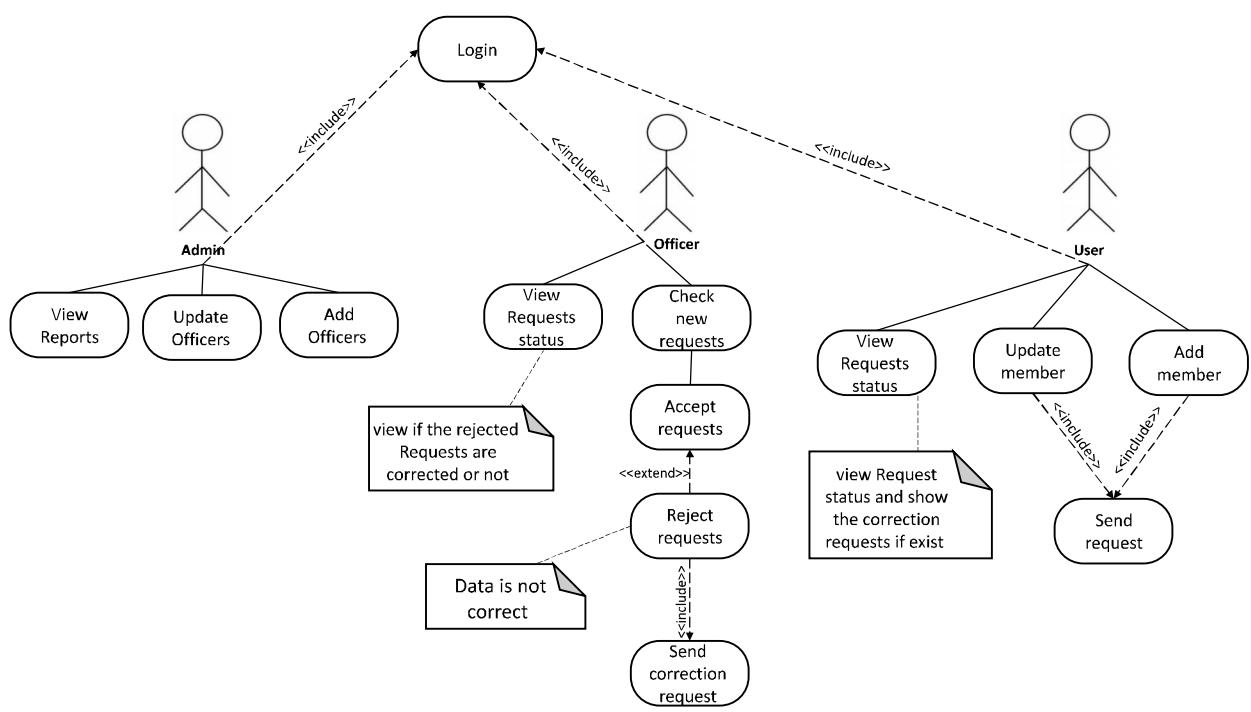
8.1/ The system shall provide to the officers a tabular form for showing add/update requests sent by their user.

9.1/ The system shall allow the officer to Validate or reject the request through 2 buttons.

9.2/ If the request is rejected The system shall provide a form for viewing the reason for rejecting the request to the user(correction request).

Non-functional requirements:

1.1/ The system shall guarantee the correct relations between officers and their users when updating the area of anyone by searching for the common areas (in the database) whenever the officer view the list of the user.



Attributes	Description
Identifier and name	UC-1 and add officer
Initiator	Admin
Goal	Add new officer's data
Pre-condition	Admin should be logged in at the beginning
Post-condition	The system will add a new officer's data
Assumptions	The officer's data not found
Main success scenario	<ul style="list-style-type: none"> 1- Admin should be logged in 2- Admin choose add officer 3- Admin add officer's data 4- Admin choose an area for the officer The system save the officer's data

Attributes	Description
Identifier and name	UC-2 and send correction request
Initiator	Officer
Goal	Send a request to user to re-correct the errors
Pre-condition	The officer rejects a request
Post-condition	User receive a request to re-correct the errors
Assumptions	The data has errors
Main success scenario	<ul style="list-style-type: none"> 1- Officer check request 2- Officer rejects a request 3- Officer Send a request to user to re-correct the errors.

Attributes	Description
Identifier and name	UC-3 and update member
Initiator	User
Goal	Update new member's data
Pre-condition	User should be logged in at the beginning
Post-condition	User send request to officer
Assumptions	The member's data is found
Main success scenario	<ul style="list-style-type: none"> 1- User should be logged in 2- User choose update member 3- User update member's data 4- User send request to officer

Attributes	Description
Identifier and name	UC-4 make login
Initiator	Everyone use system
Goal	Log in to the system and use it
Precondition	You shall have an account to login
Postcondition	After logging in, the system determines the identity of the user (Admin-officer-user) and gives him his own permissions
Assumptions	Everyone use system have an account
Main success scenario	<ol style="list-style-type: none"> 1. Enter user name. 2. Enter password. 3. Confirm password.

Attributes	Description
Identifier and name	UC-5 update officers
Initiator	Admins only
Goal	update officers data
Precondition	Admin must be logged in at the beginning
Postcondition	the admin specify the officer who wants to update his data and the system will record new officer's data
Assumptions	There is data for officers on the system
Main success scenario	<ol style="list-style-type: none"> 1. Admin login to the system. 2. Select officer. 3. Update officer's data and save it on system officer's data.

Attributes	Description
Identifier and name	UC-6 view requests status
Initiator	users
Goal	View Requests status and show the correction requests if exist
Precondition	User have an account and login.
Postcondition	User can View Requests status
Assumptions	The user make request
Main success scenario	<ol style="list-style-type: none"> 1. User login to the system. 2. The user choose operation (add \ update). 3. View Requests status and show the correction requests if exist.

Attributes	Description
Identifier and name	UC-7 Add Member
Initiator	User
Goal	Add new member
Precondition	User must Login into system
Postcondition	Send Request to officer to add new member
Main success scenario	<ol style="list-style-type: none"> 1. User Must login into system 2. he choose add member to add new member 3. Enter member 's data 4. User send Request to officer of his region to add this new member

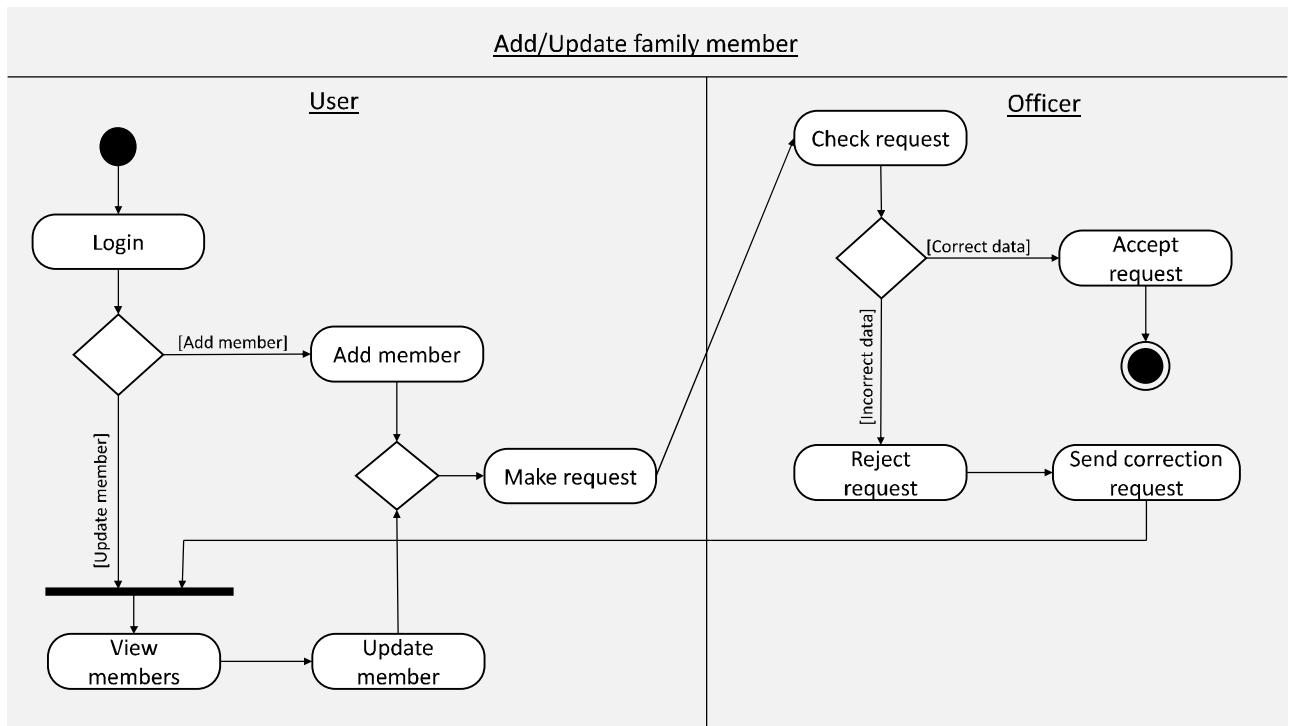
Attributes	Description
Identifier and name	UC-8 View Reports
Initiator	Admin
Goal	Check any reports send from officers or Users
Precondition	Login
Postcondition	system will show reports to admin
Main success scenario	<ol style="list-style-type: none"> 1. Admin login into system. 2. he choose to view reports . 3. Admin can show the reports in different ways(sorted/unsorted,asc,desc).

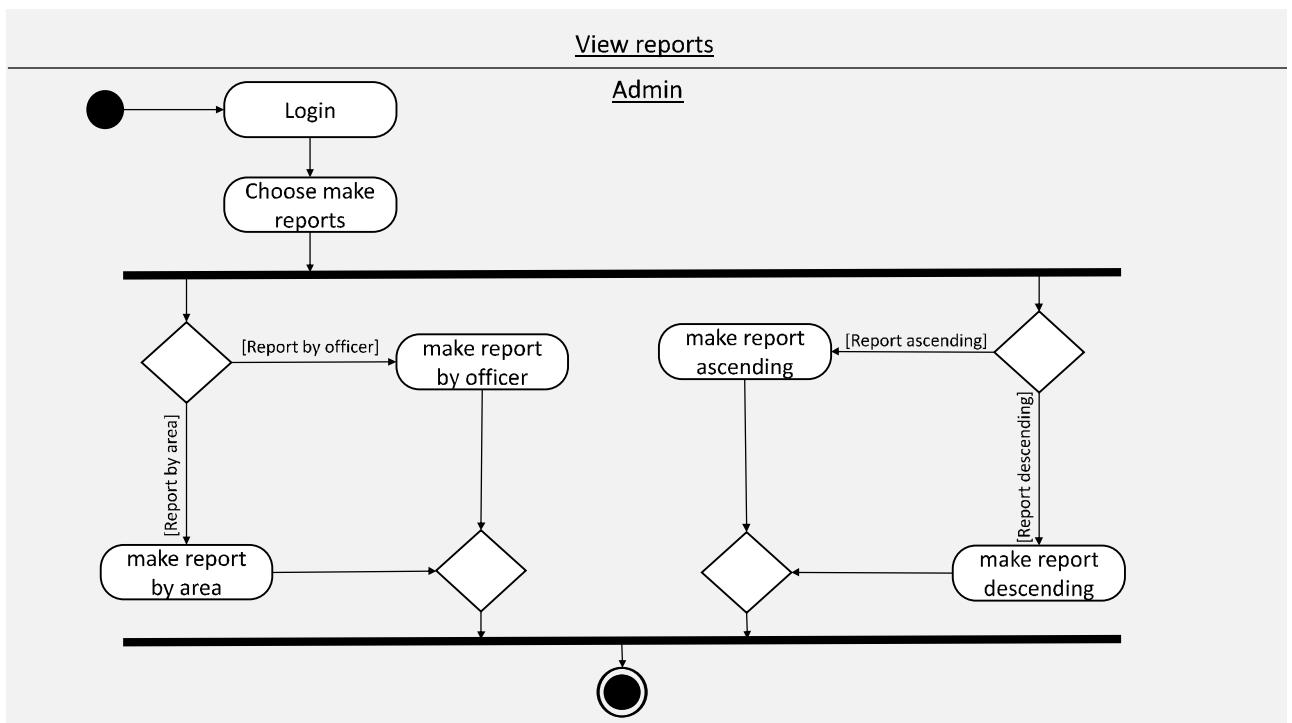
Attributes	Description
Identifier and name	UC-9 Accept Requests
Initiator	officers
Goal	Accept Requests from Users
Precondition	Login then check new requests
Postcondition	Accept or reject requests
Main success scenario	1.officers login into system. 2.officers choose to check requests . 3.officers check requests send by Users . 4.officers could accept or reject requests .
Extension	if officers find out that data entered by the users not completer or not correct it reject the request then send correction request to users

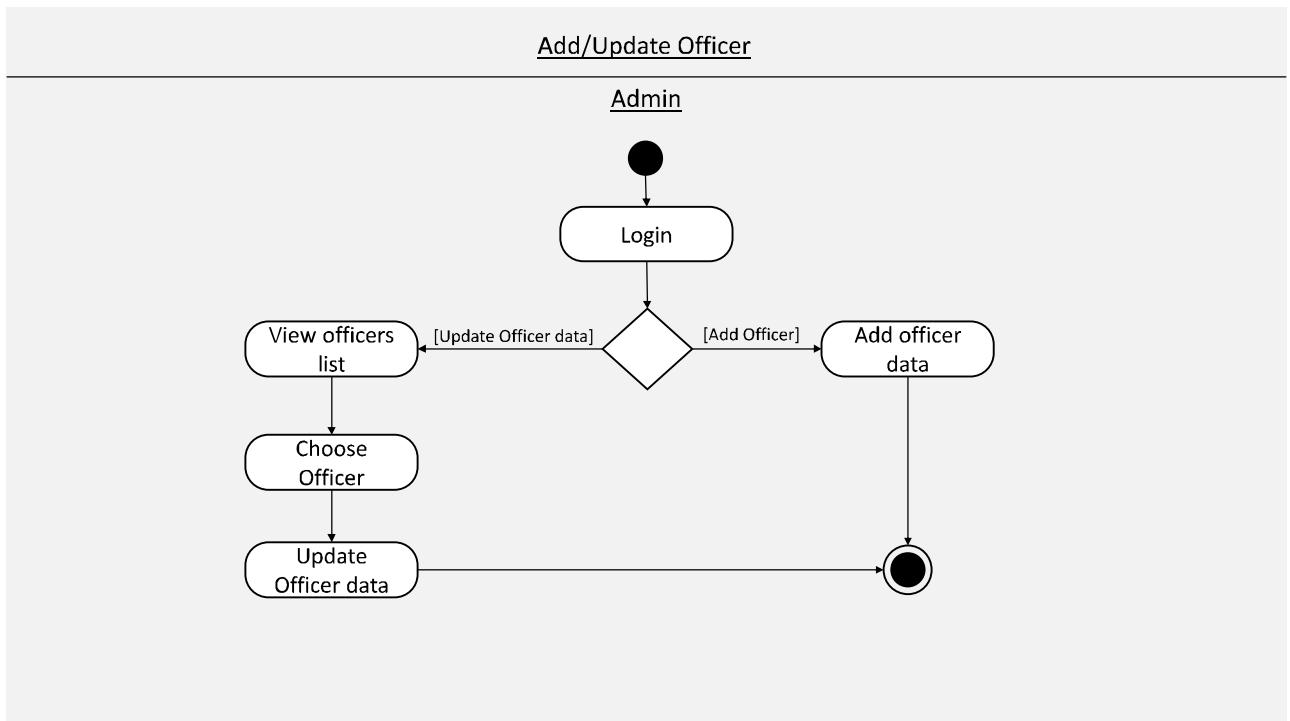
Attributes	Description
Identifier and name	UC-10 View Requests Status
Initiator	Officer
Goal	View if the rejected requests are corrected or not
Precondition	The officer must login
Postcondition	A report shows the status of the rejected requests
Assumptions	The expected Initiator is the officer to revise the rejected requests
Main success scenario	1.The officer wants to view the status of the rejected requests

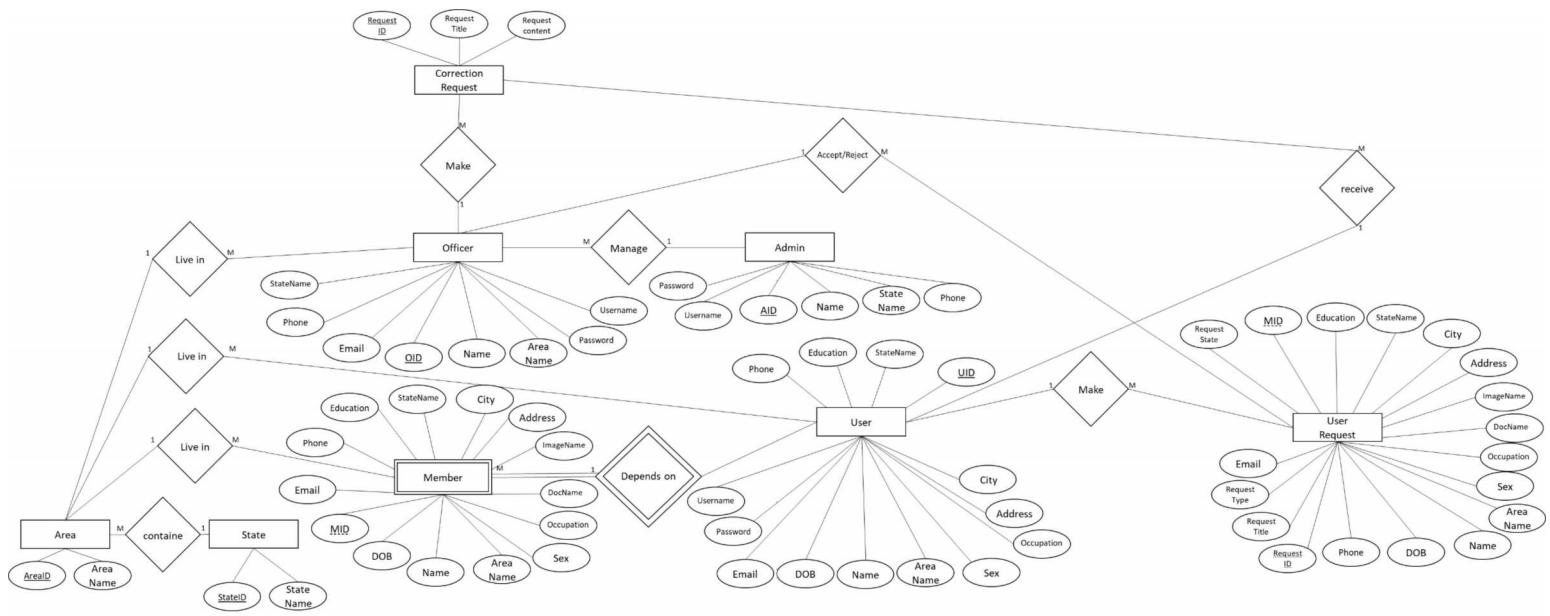
Attributes	Description
Identifier and name	UC-11 Check New Requests
Initiator	Officer
Goal	Check if there are any new requests from the user
Precondition	The officer must login
Postcondition	A report shows the requests that the user sent
Assumptions	The expected Initiator is the officer to view user requests
Main success scenario	<ol style="list-style-type: none"> 1.The officer wants to view user requests 2.The officer will accept or reject the requests

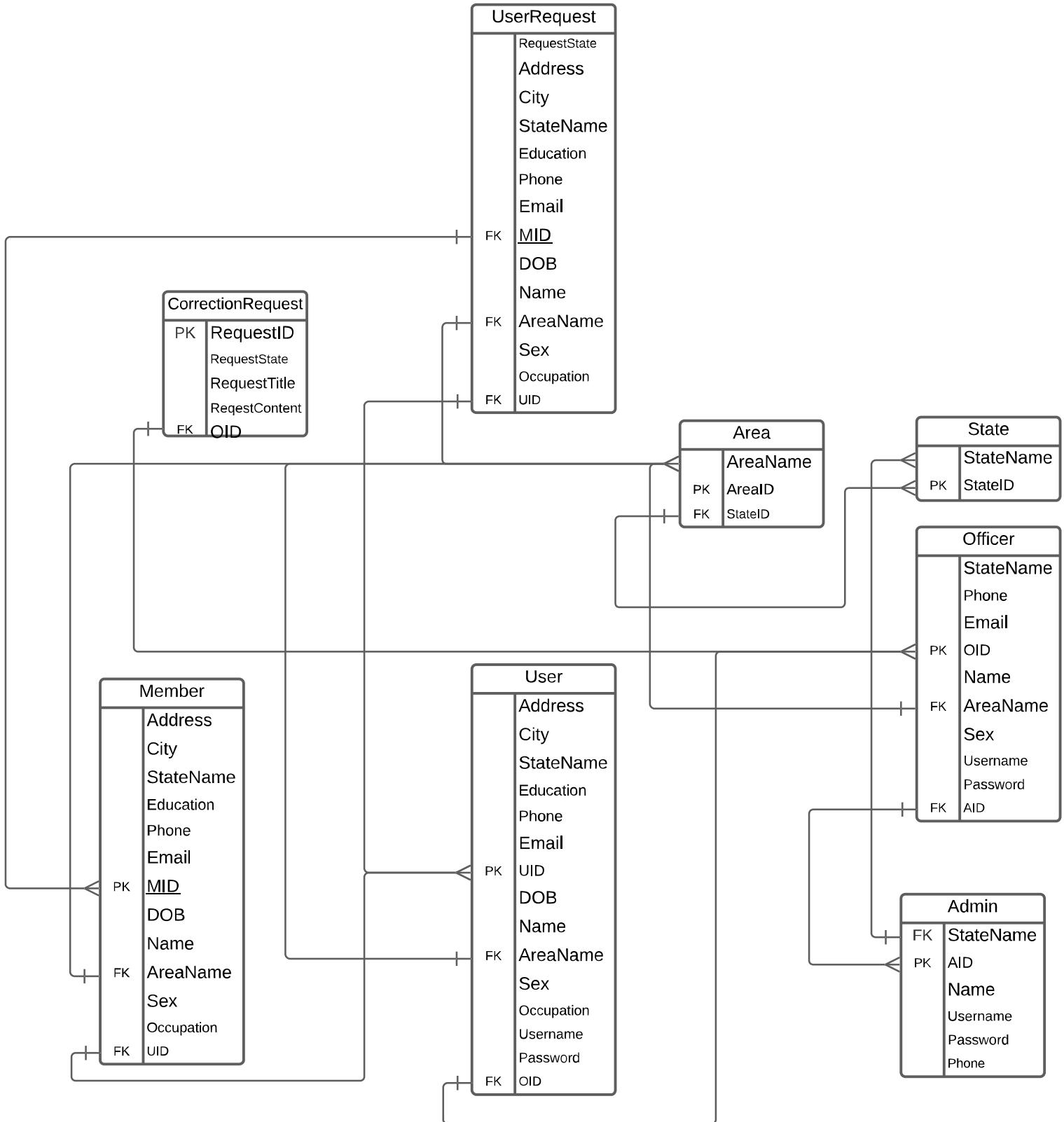
Attributes	Description
Identifier and name	UC-12 Send Request
Initiator	User
Goal	Send the change which the user wants to do in a request
Precondition	The User must login
Postcondition	The users request will be sent to the Officer
Assumptions	The expected Initiator is the User to send his requests
Main success scenario	<ol style="list-style-type: none"> 1.the user will make a request 2.the system will send it to the officer

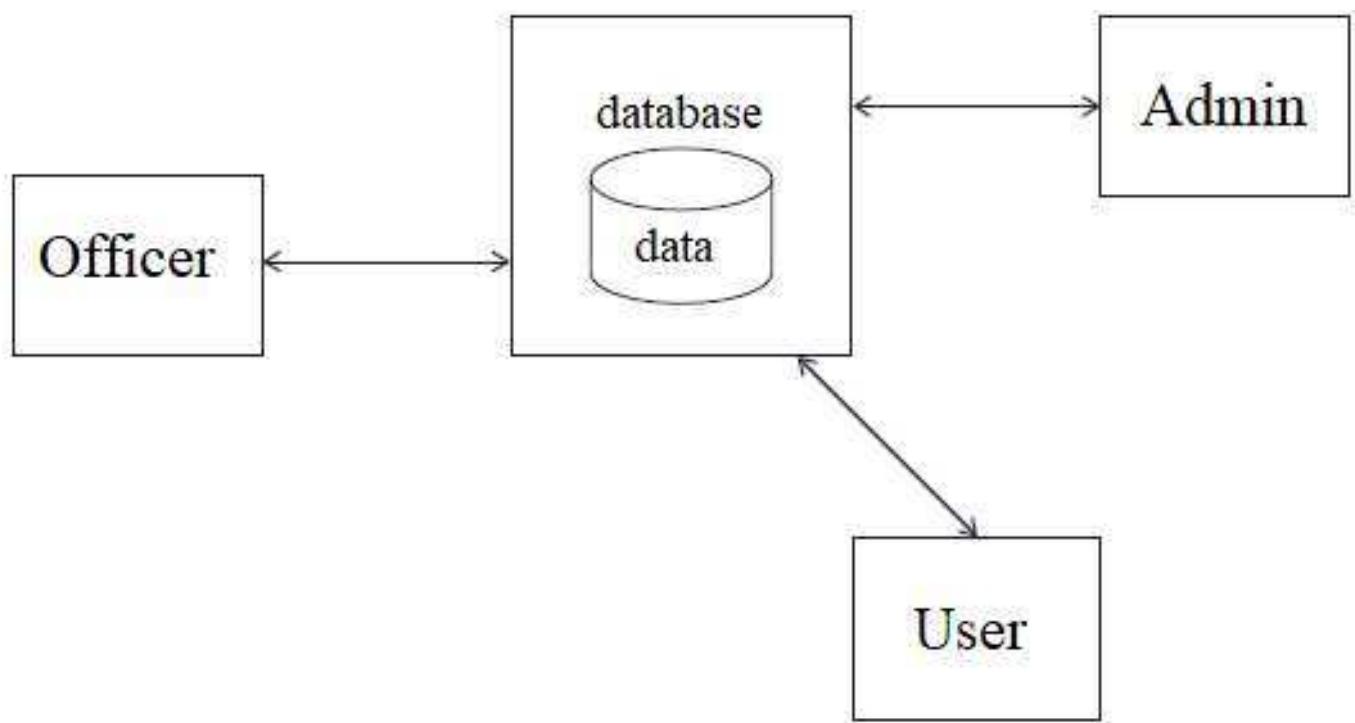




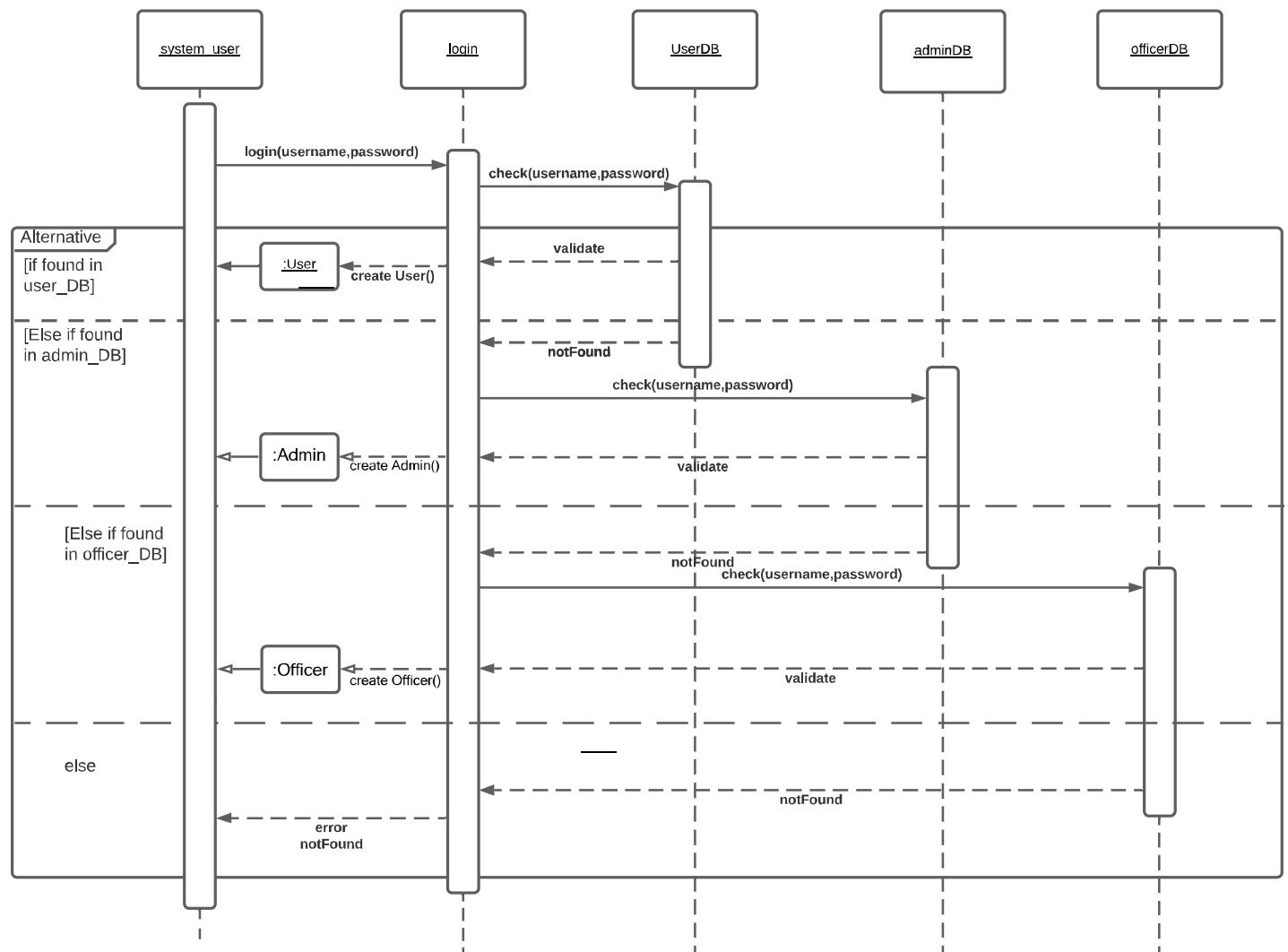




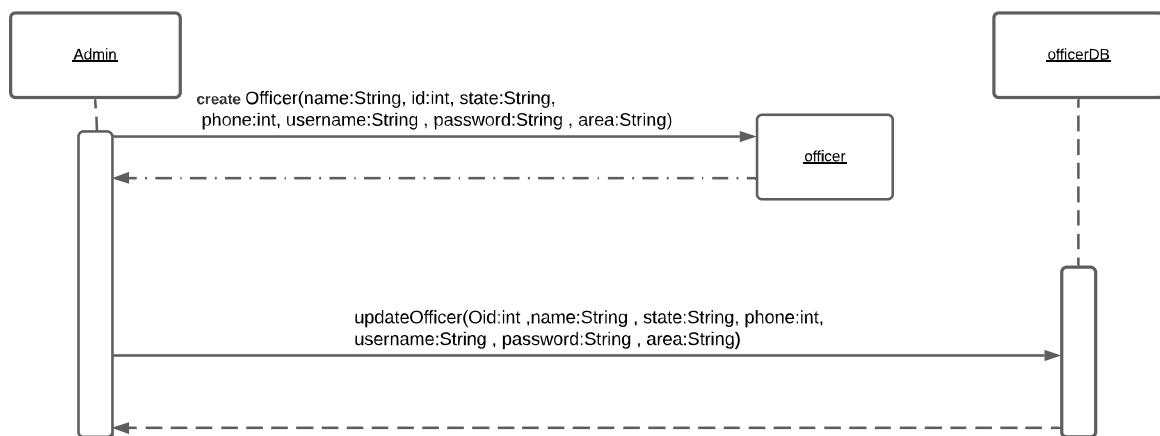




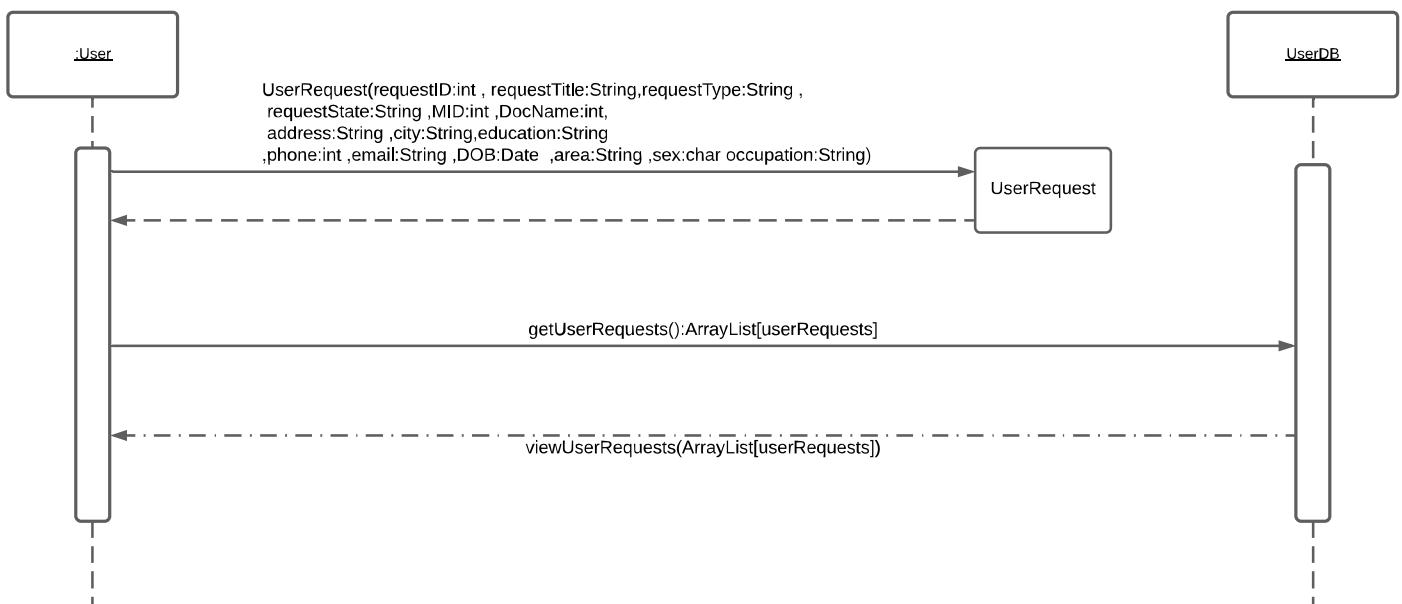
login



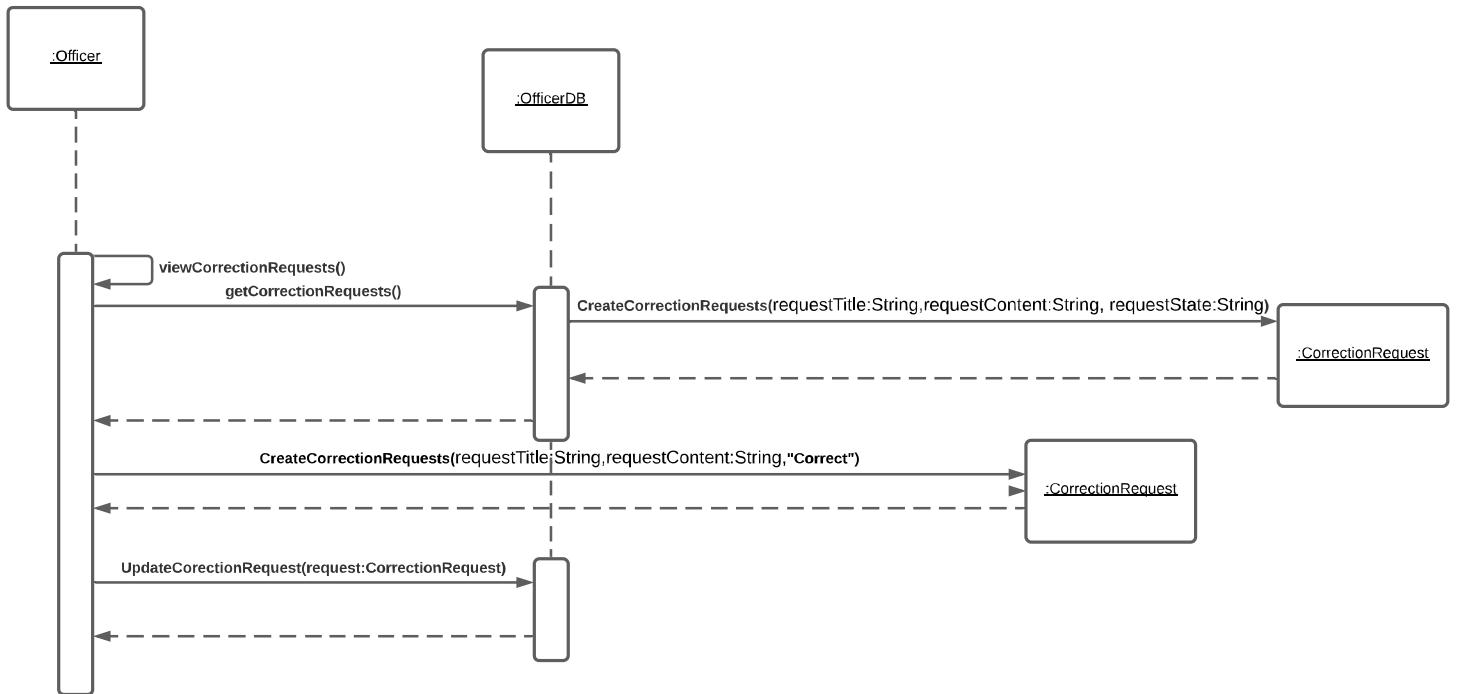
Update_Officer



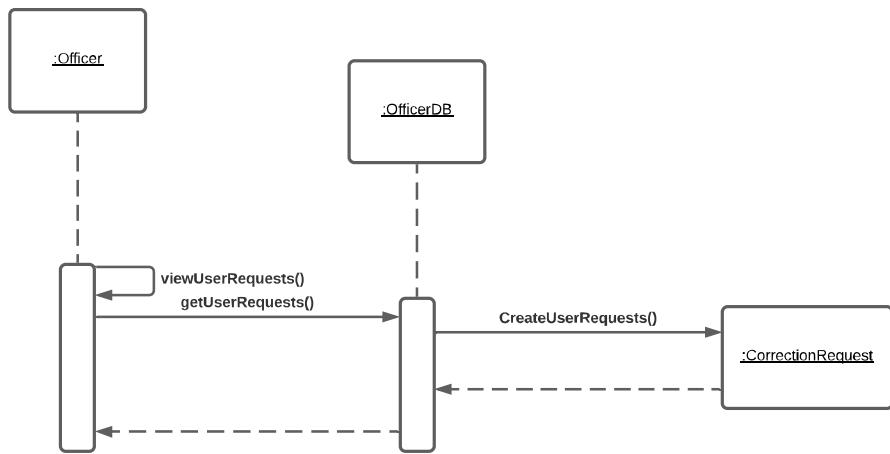
View_User_Request



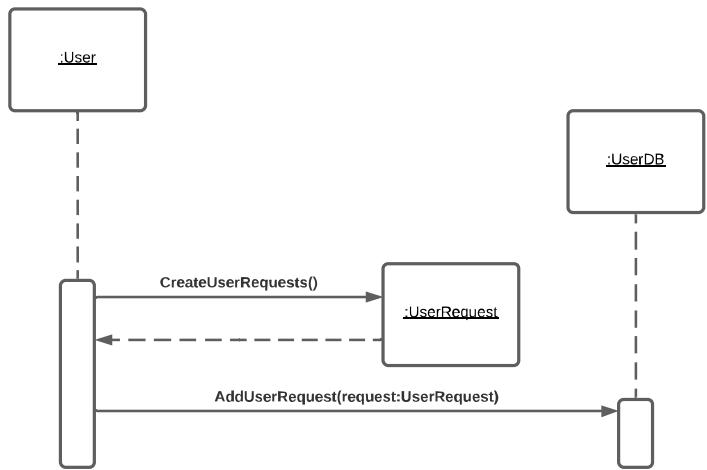
View Requests Status



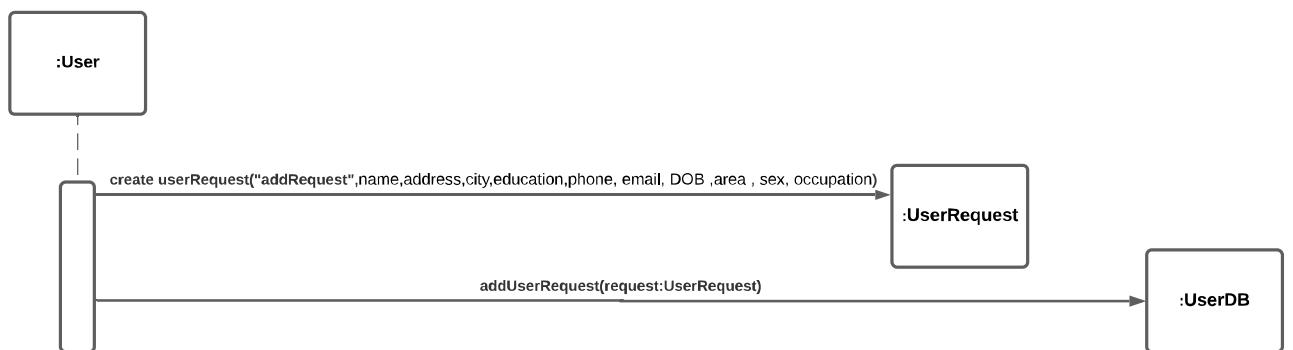
Check New Requests



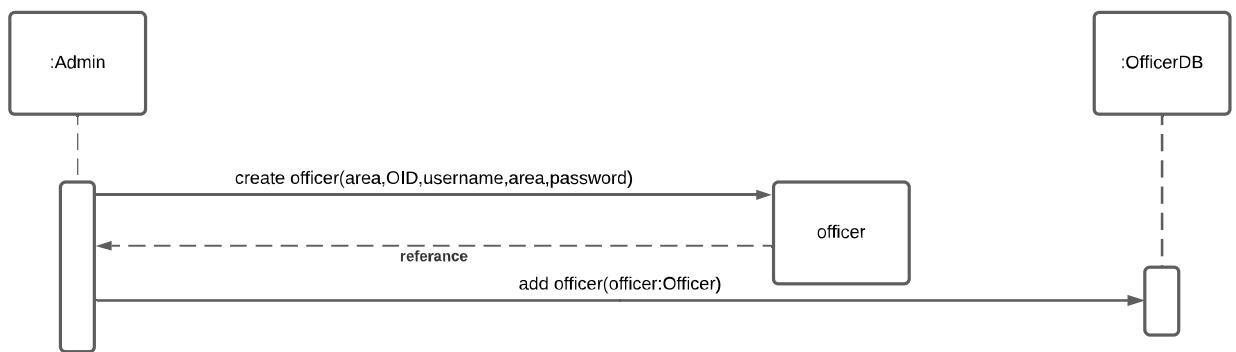
Send Requests



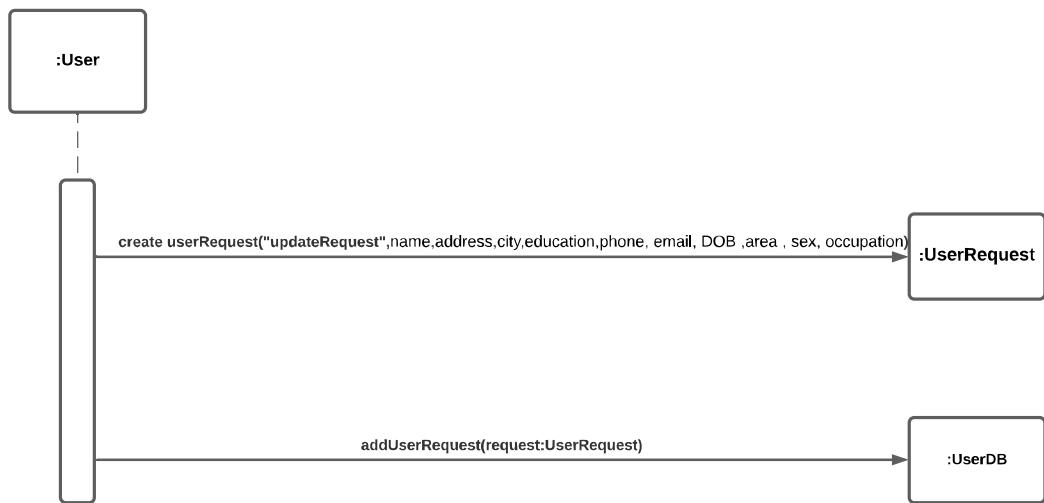
Add_member



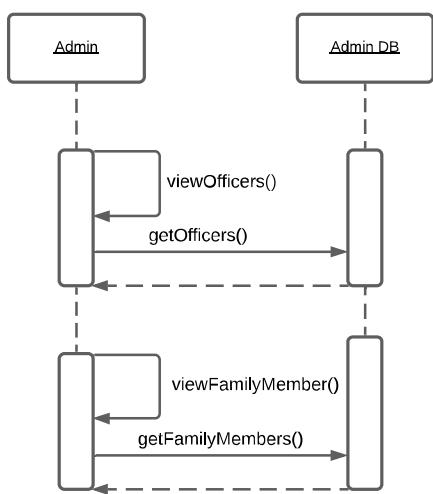
Add_Officer



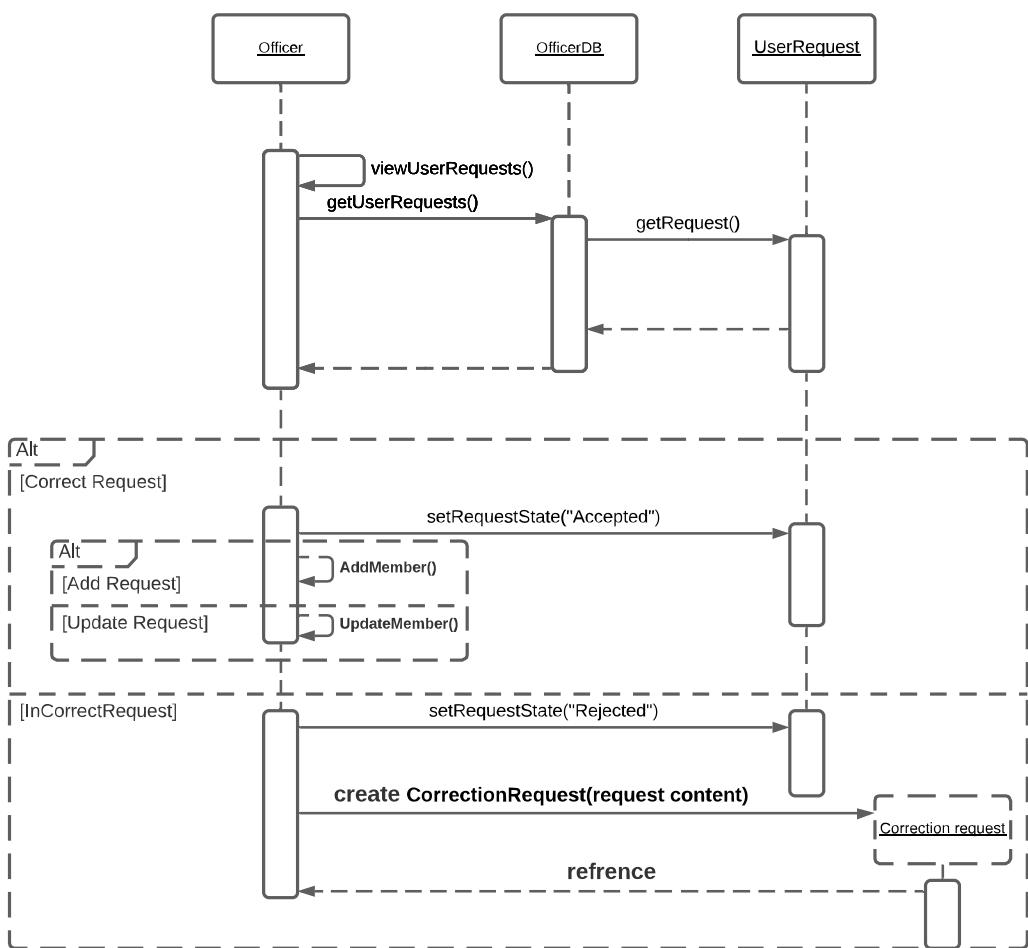
Update_member



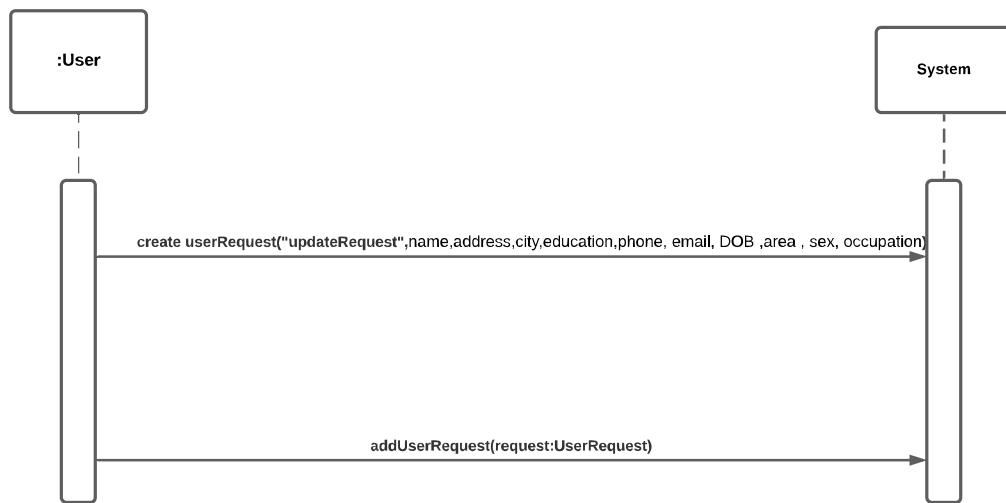
View_Report



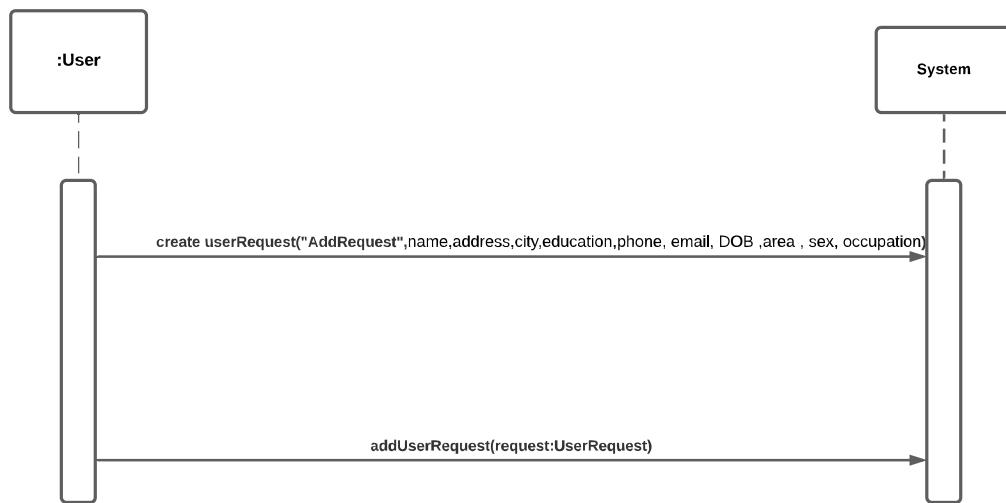
Accept Or Reject Request



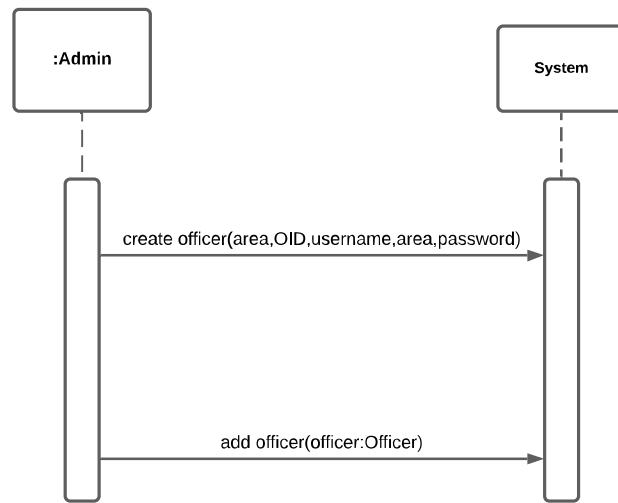
Update_member



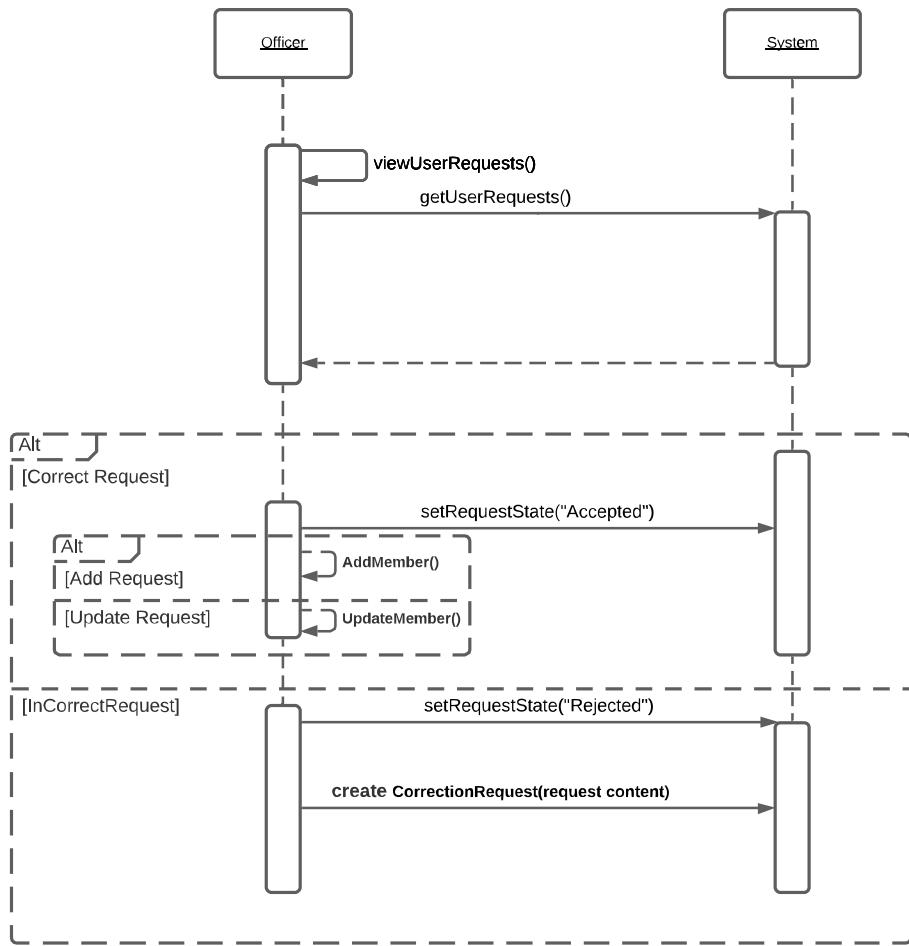
Add_member



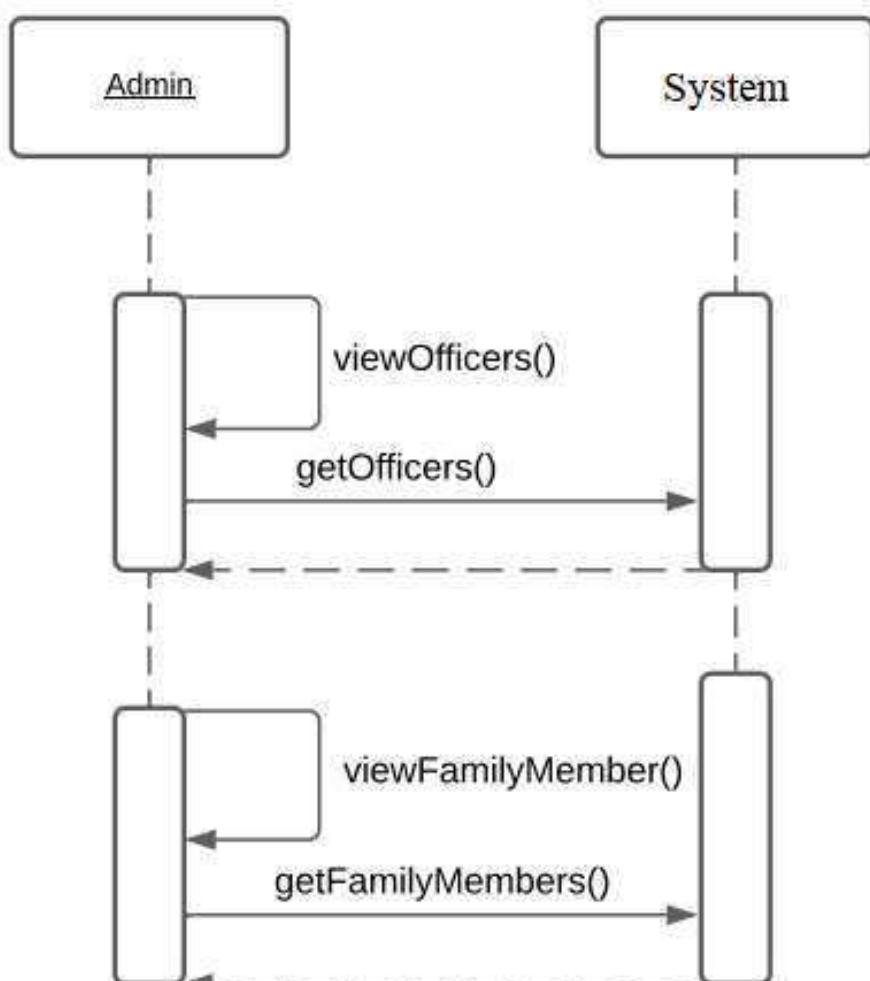
Add_Officer



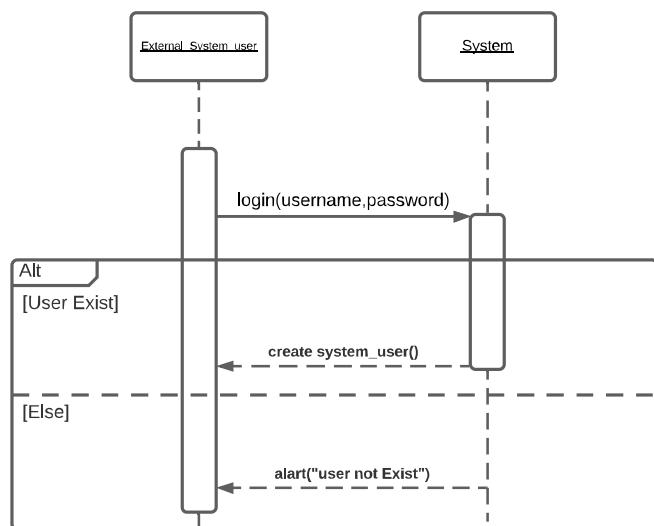
Accept Or Reject Request



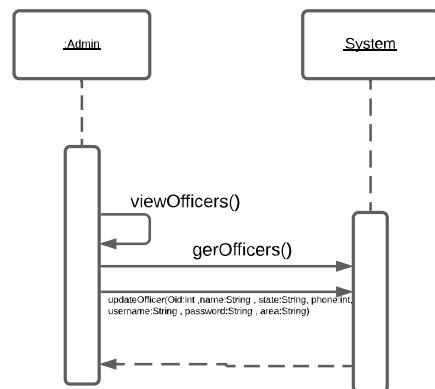
View_Report



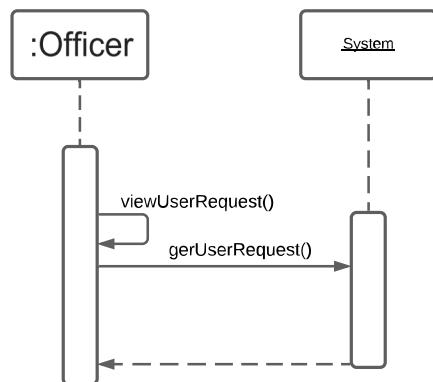
login



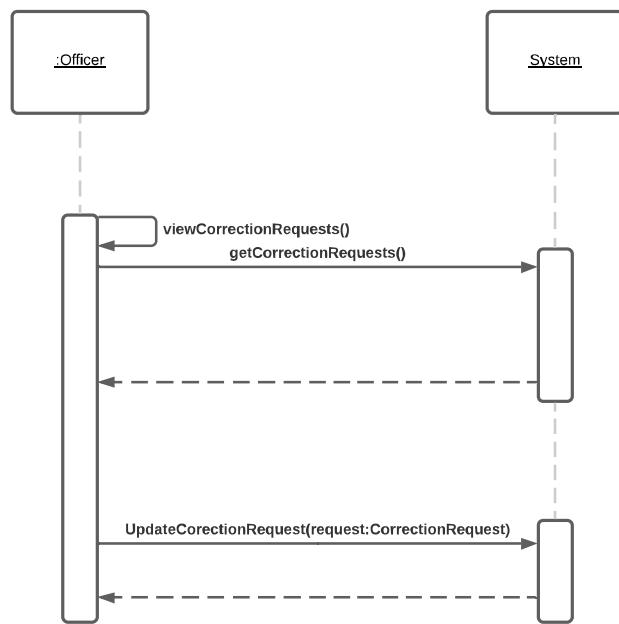
Update_Officer



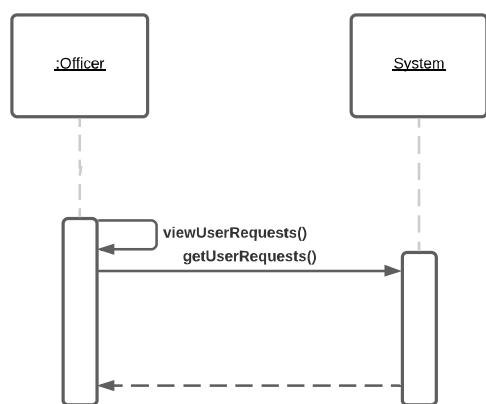
View_Request



View Requests Status



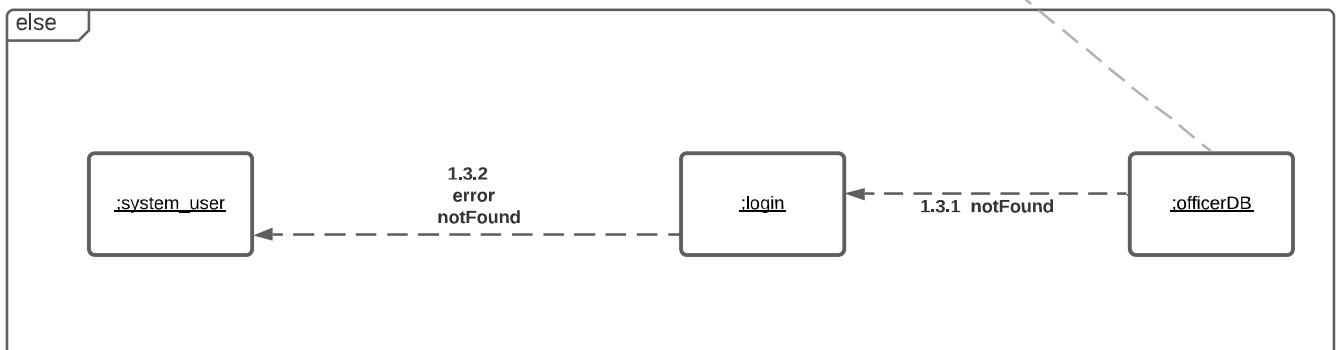
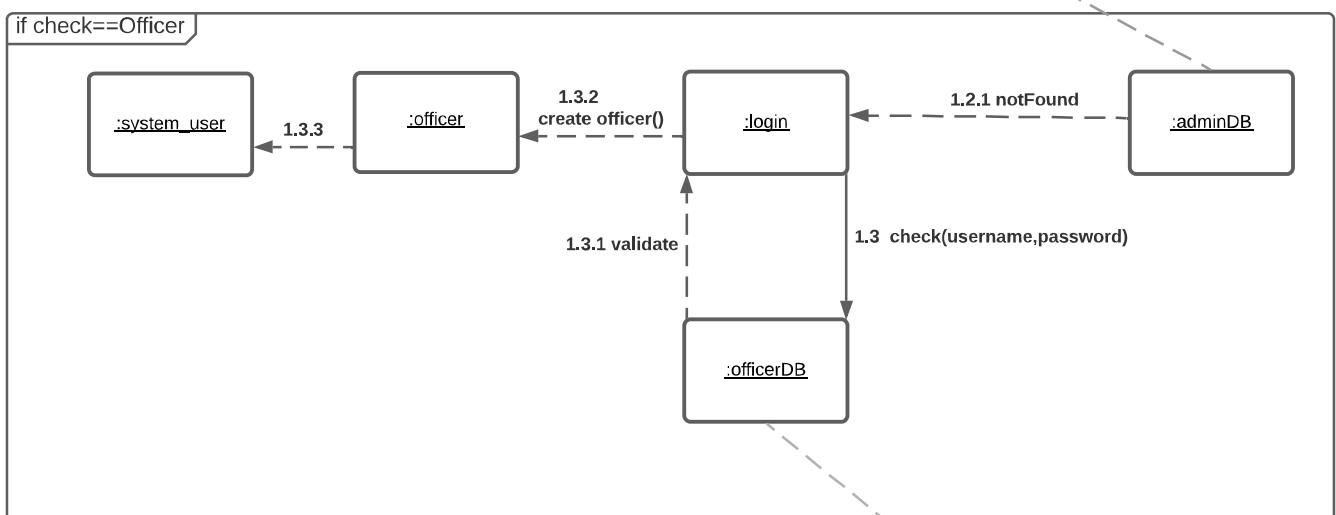
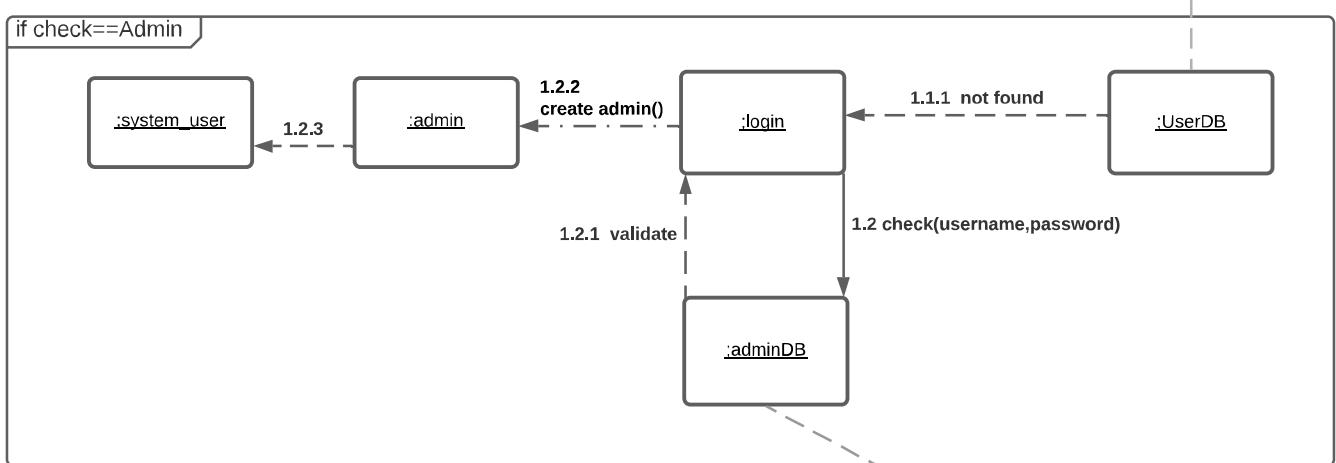
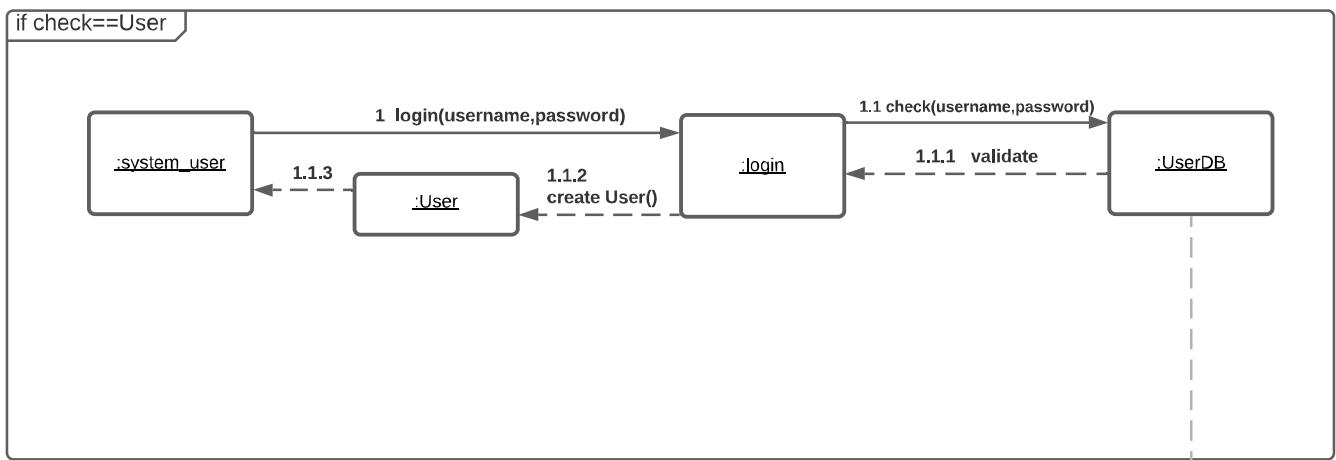
Check New Requests



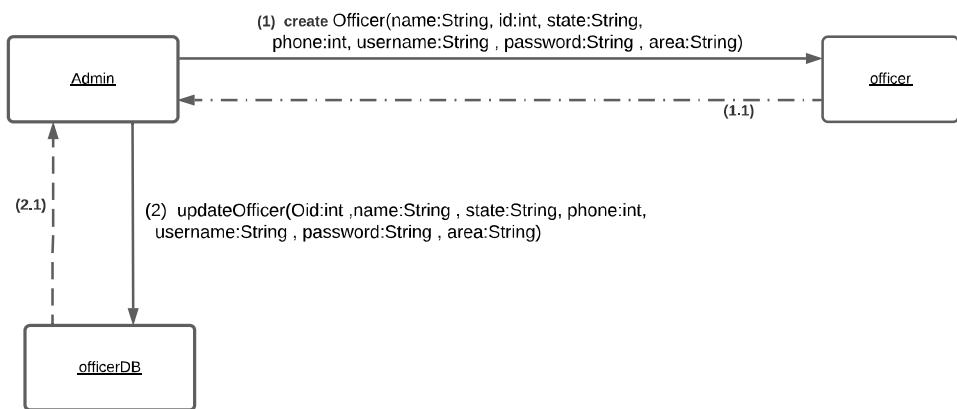
Send Requests



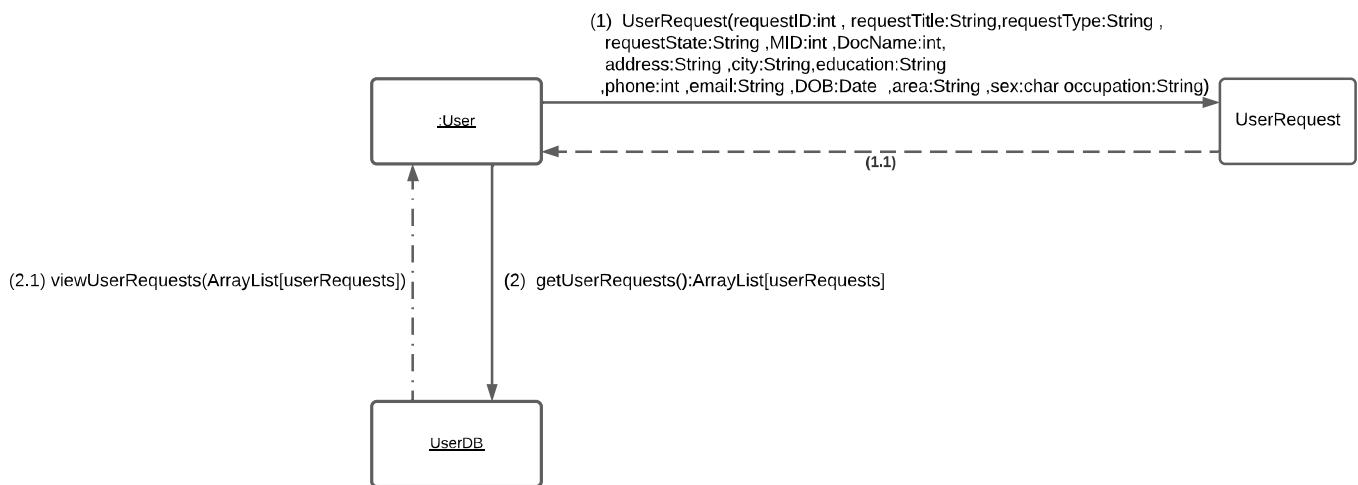
login



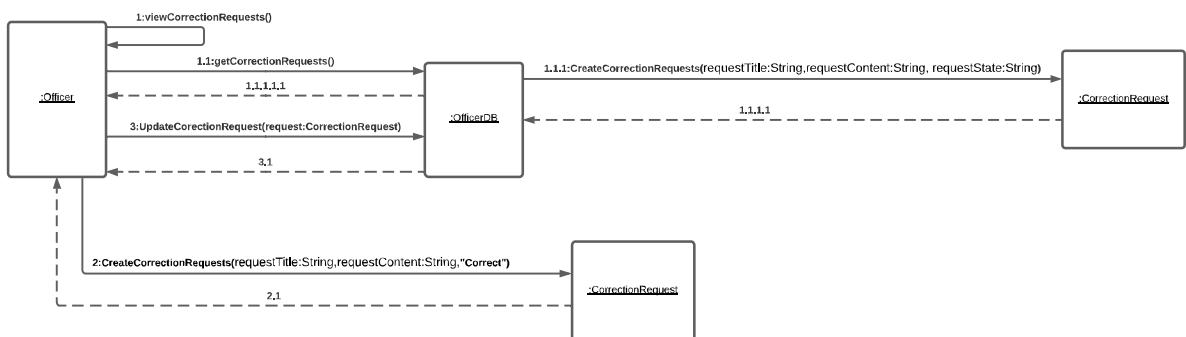
Update_Officer



view User Requests



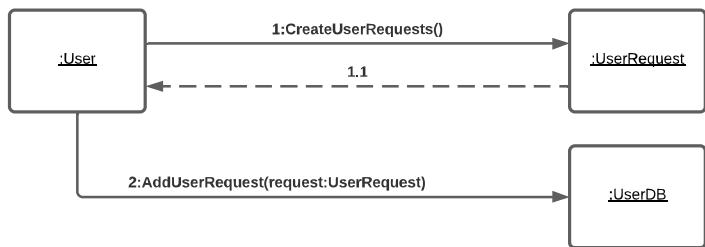
View Requests Status



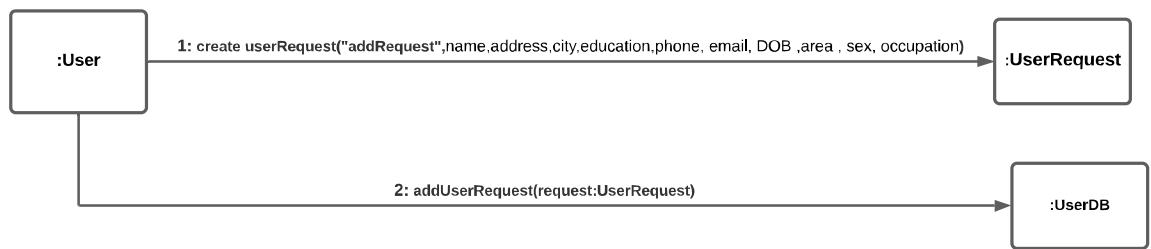
Check New Requests



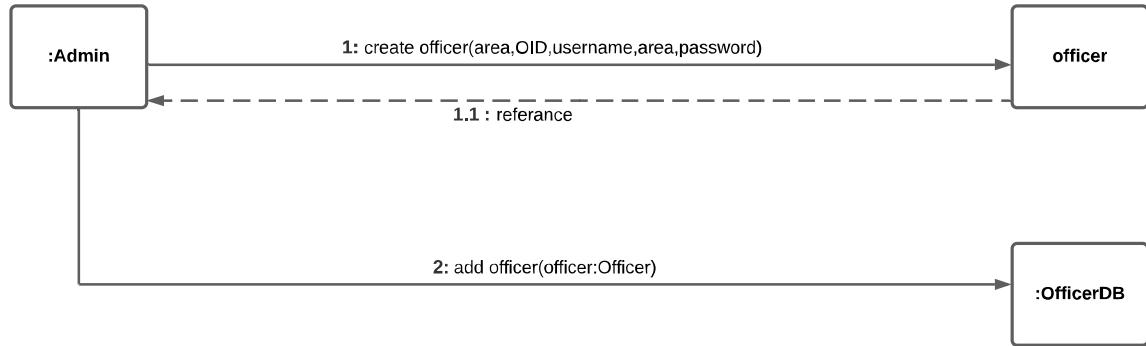
Send Requests



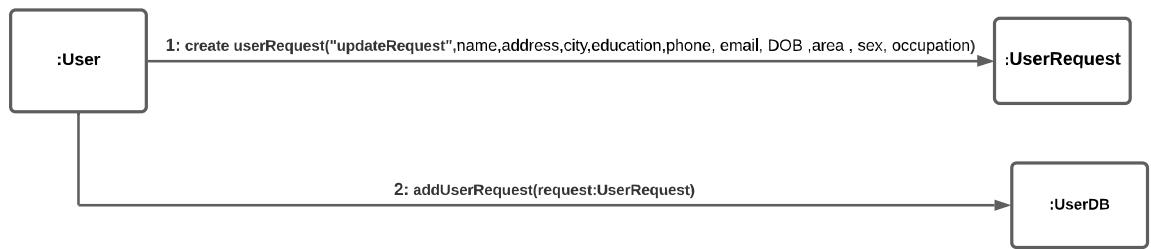
Add member



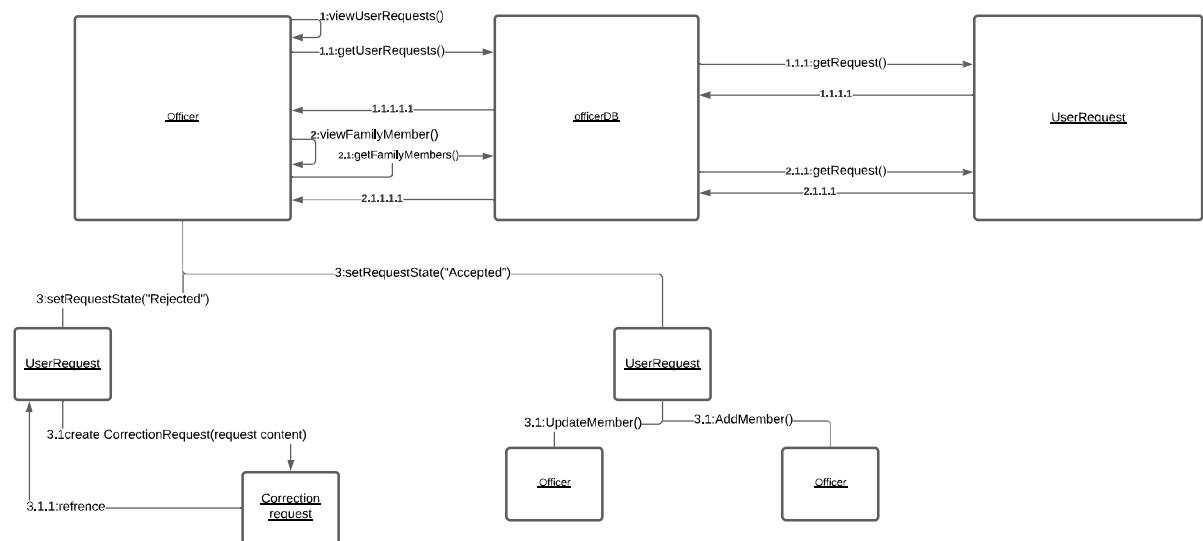
Add Officer



Update member



Accept and Reject Requests



View Report

