UNIVERSITY OF TURKISH AERONAUTICAL ASSOCIATION

AERONAUTICS and ASTRONAUTICS

**2017-2018 Fall Semester**

**CENG-203**

**Object-Oriented Design and Programming**

**Project Phase 1 RAD**

# Missing Person System

| **Group Members** | **Student Id** | **Name** | **Surname** |
|---|---|---|---|
| **1.** | 170444014 | Batuhan | Günday |
| **2.** | 170444022 | Ömer | Çelik |

# Case Study For Missing Person System

## INTRODUCTION

This system was designed for keeping the missing person database and simplify the find that people. To provide that system safety, every user must have to a verified account. These verifications are made by admins.

This system has 3 different roles. They are "User, Informer, and Admin". Users can add Missing Person to the system, update and remove that which is added by the same user. Informers can see the list of Missing Person and write feedback about these people. If this feedback is confirmed by an admin it will send to the user.

## REQUIREMENT ELICITATION

### IDENTIFYING ACTORS AND SCENARIOS

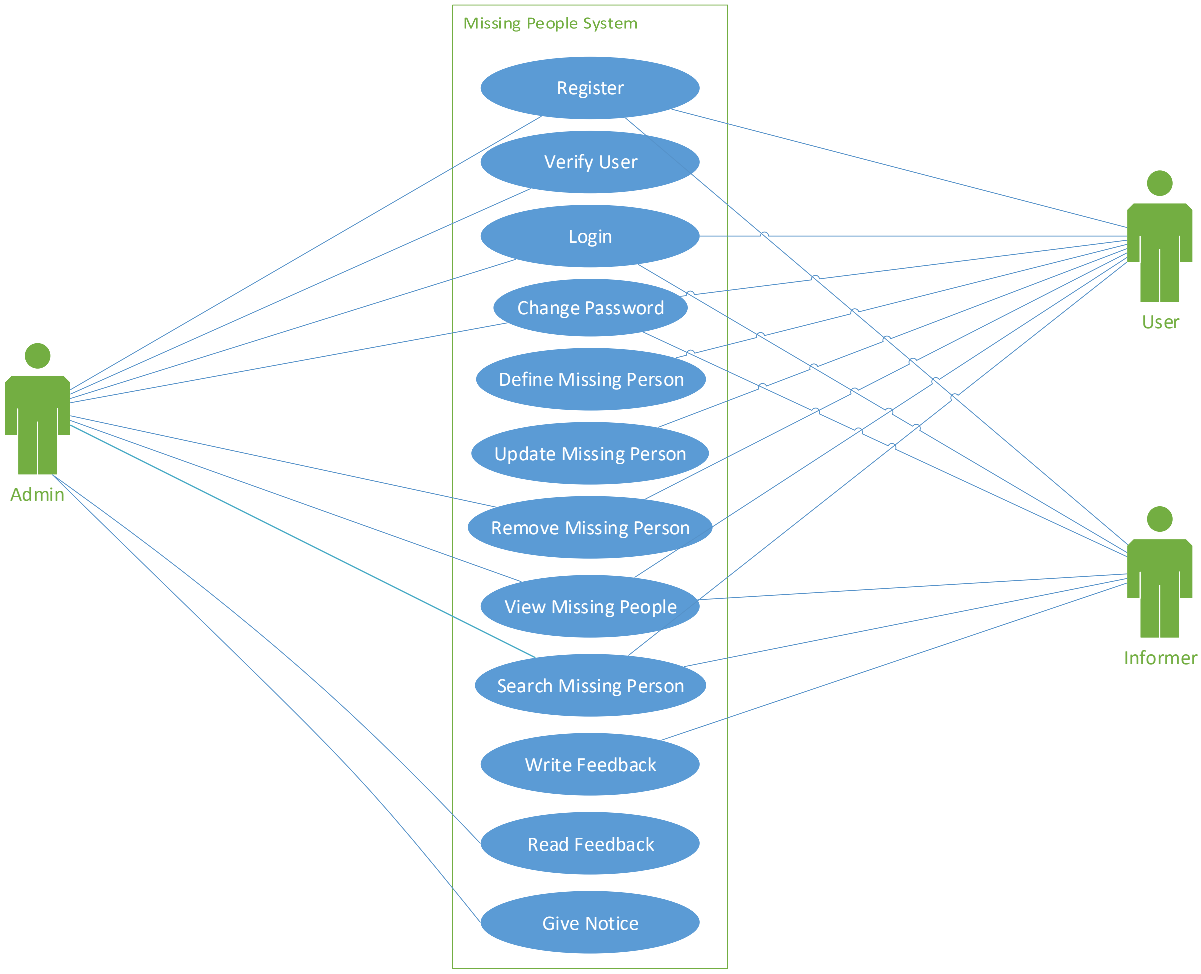
Actors

1. User
2. Informer
3. Admin

Scenarios

- Starting Scenario: Register
- Other Scenarios: Verify User, Login, Change Password, Define Missing Person, Update Missing Person, Remove Missing Person, Search Missing Person, Write Feedback, Read Feedback, Give Notice, View Missing People, Update User

## IDENTIFYING USE CASE



Register: Everyone must be register to the system.
Verify User: Only Admin can verify user.
Login: All kind of actors must be login the system.
Change Password: Everyone can change their password.
Define Missing Person: Only User can define missing person.
Update Missing Person: Only User can update missing person.
Remove Missing Person: All Admins can remove any missing person, but User can only remove especially that they defined.
View Missing Person: Both User and Informer can view missing person.
Search Missing Person: Both User and Informer can search missing person.
Write Feedback: Only Informer can write feedback.
Read Feedback: Only Admin can read feedback.
Give Notice: Only Admin can give notice.
Update User: All the admins can update/remove user.

## IDENTIFYING NON-FUNCTIONAL REQUIREMENTS

Below are the consolidated nonfunctional requirements of MPS, Following the first version of the detailed

1. Usability: System should have user-friendly GUI and will designed according to user experience.
2. Reliability: If one of the people in the system has crash, this crash should not affect others.
3. Performance: This system must support multiple active user.
4. Supportability: Admin must be able to add new properties for the system.
5. Legal: Every actor who is registered to the system, automatically accepts MPS's legal licenses and terms of usage.

## SCENARIOS

1. Register

| **Name:** | **Register** |
|---|---|
| **Actors:** | Admin, User, Informer |
| **Flow of Events:** | • Admin, User and Informer must register to the Missing Person System by using createAccount() function.<br>• Register process must include some attributes such as username, password.<br>• Actors write their username and password to the textField.<br>• If username and password valid, system accepts them.<br>• If at least password or username are invalid register will be denied and system gives a message box  that returns "This username/password invalid. Try another username/password"<br>• Register request will be send to Admin for verification.<br>• If admin confirm your request your account has been created. |
| **Entry Condition:** | Actors click register button. |
| **Exit Condition:** | Actors successfully register or log out. |
| **Quality Requirements:** | • Password must contain 8-30 characters with at least one number and one uppercase character.<br>• Every username must be unique. |

2. Verify User

| **Name:** | **Verify User** |
|---|---|
| **Actors:** | Admin |
| **Flow of Events:** | - Admin must login and open his/her dashboard.<br>- Dashboard have Account Request panel that panel should include Account Register Request.<br>- Admin will be confirming that received register request. |
| **Entry Condition:** | Admin must login and click verify user button. |
| **Exit Condition:** | Admin logout the system. |
| **Quality Requirements:** | - The information of the person who wants to create an account must be valid. |

3. Login

| **Name:** | **Login** |
|---|---|
| **Actors:** | Admin, User, Informer |
| **Flow of Events:** | - Admin, User and Informer must login to system.<br>- They must be entering their username and password to the textField.<br>- Click the login button.<br>- If username and password were entered to the system correctly, they will be entered to dashboard. Otherwise system gives a message box that returns “Check your username/password” |
| **Entry Condition:** | Login panel is opening frame for system. |
| **Exit Condition:** | Exit button of frame. |
| **Quality Requirements:** | - Username and password must be authenticated. |

4. Change Password

| **Name:** | **Change Password** |
|---|---|
| **Actors:** | Admin, Informer, User |
| **Flow of Events** | • Client click on forgot password button on login panel.<br>• They have to enter their email and username to verify their account.<br>• If email and username matched system accepts your request and client can change password. |
| **Entry Condition:** | Click on forgot password button. |
| **Exit Condition:** | Click on submit button. |
| **Quality Requirements:** | • Firstly, email and username have to be match.<br>• Password must not be the same as the old one.<br>• Password must follow the rules. |

5. Define Missing Person

| **Name:** | **Define Missing Person** |
|---|---|
| **Actors:** | User |
| **Flow of Events:** | • User must login to system.<br>• User must open the corresponding frame.<br>• User must define person with its all details.<br>• Click submit button. |
| **Entry Condition:** | Click define missing person button. |
| **Exit Condition:** | Click submit or exit button. |
| **Quality Requirements:** | • Missing person must be well defined. |

6. Update Missing Person

| **Name:** | **Update Missing Person** |
|---|---|
| **Actors:** | User |
| **Flow of Events:** | • User must login to system.<br>• User must open the corresponding frame that includes his/her own defined person.<br>• User changes that information he/she already submitted.<br>• User clicks the update button. |
| **Entry Condition:** | Click update person button. |
| **Exit Condition:** | Click update or exit button. |
| **Quality Requirements:** | • There must be a difference according to old one.<br>• Changes must be reliable. |

7. Remove Missing Person

| **Name:** | **Remove Missing Person** |
|---|---|
| **Actors:** | Admin, User |
| **Flow of Events:** | • Admin/User must login to system.<br>• If the one who wants to remove Missing Person is admin, he/she must open dashboard.<br>• If the one who wants to Remove Missing Person is User, then he/she opens the corresponding frame.<br>• If the admin removed a person, user will be notified, and person will be removed.<br>• If the user wants to remove person, a remove request will send to admin, if an admin confirm that request, then person will be removed. |
| **Entry Condition:** | Search and select the Missing Person then click the appearing remove button |
| **Exit Condition:** | Click remove button. |
| **Quality Requirements:** | • Must remove the right missing person. |

8. View Missing People

| **Name:** | **View Missing People** |
|---|---|
| **Actors:** | Admin, Informer, User |
| **Flow of Events:** | • Admin, Informer, User must be login.<br>• Missing People List shown on the main frame. |
| **Entry Condition:** | Login of all actors. |
| **Exit Condition:** | Logout. |
| **Quality Requirements:** | • Pull the list from the database properly. |

9. Search Missing Person

| **Name:** | **Search Missing Person** |
|---|---|
| **Actors:** | Admin, Informer, User |
| **Flow of Events:** | • Admin, Informer, User must be login.<br>• On the main frame there is a search box which is include  some properties about the missing person.<br>• They must fill at least one information of missing person.<br>• They must click the search button.<br>• Results will be filtered according to information you entered.<br>• If the actor wants to reset search and view all the missing people, they must click the reset search button. |
| **Entry Condition:** | Click the search button. |
| **Exit Condition:** | Click the reset search button or logout. |
| **Quality Requirements:** | • Results must be consistent.<br>• Results must be contained by the database. |

10. Write Feedback

| **Name:** | **Write Feedback** |
|---|---|
| **Actors:** | Informer |
| **Flow of Events:** | • Informer must be login.<br>• Informer must be select the person he/she wants to write feedback.<br>• Click submit button. |
| **Entry Condition:** | Select the person from the list and click the submit button. |
| **Exit Condition:** | Click the submit button. |
| **Quality Requirements:** | • Feedbacks must be rationale. |

11. Read Feedback

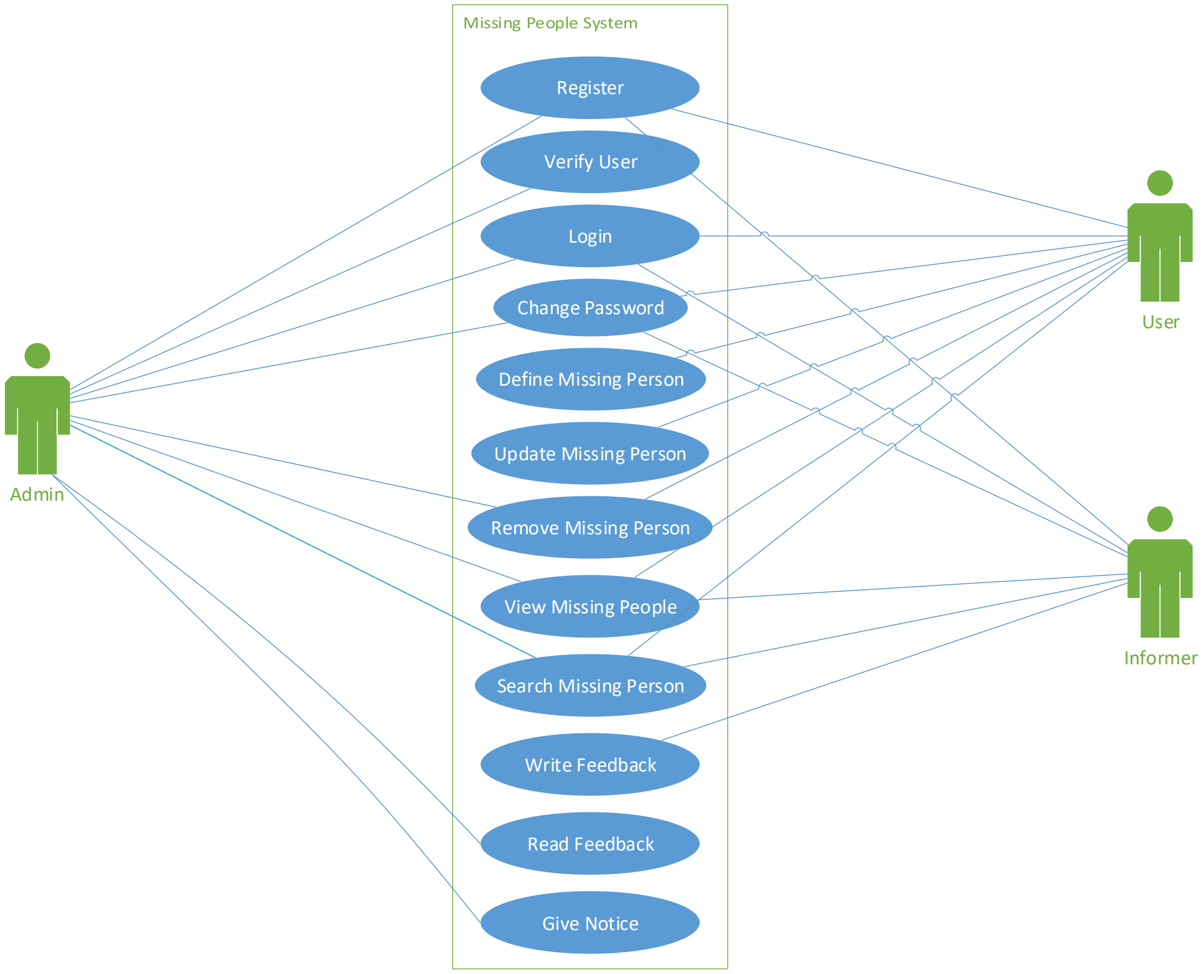
| **Name:** | **Read Feedback** |
|---|---|
| **Actors:** | Admin |
| **Flow of Events:** | • Admin must login.<br>• Click the View Feedbacks button on the dashboard.<br>• Feedback will be shown on the new frame. |
| **Entry Condition:** | Click the view feedbacks button. |
| **Exit Condition:** | Logout or exit from the feedbacks frame. |
| **Quality Requirements:** | • Feedbacks must be listed properly. |

12. Give Notice

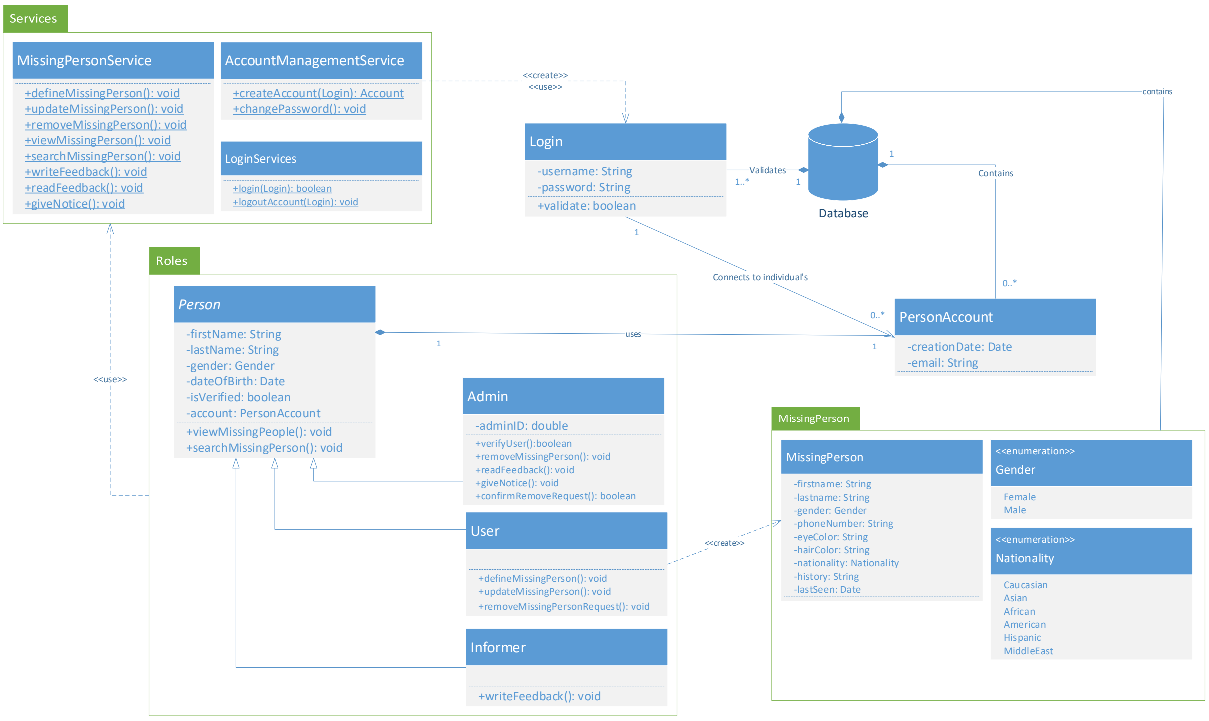
| | |
|---|---|
| **Name:** | **Give Notice** |
| **Actors:** | Admin |
| **Flow of Events:** | - Admin must login<br>- If the missing person is found, admin sends notification to the user.<br>- Then, missing person has been removed by system. |
| **Entry Condition:** | Admin choose a person from the list. |
| **Exit Condition:** | Admin clicks founded button. |
| **Quality Requirements:** | - Person must be founded. |

## SYSTEM MODELS

## USE CASE DIAGRAM

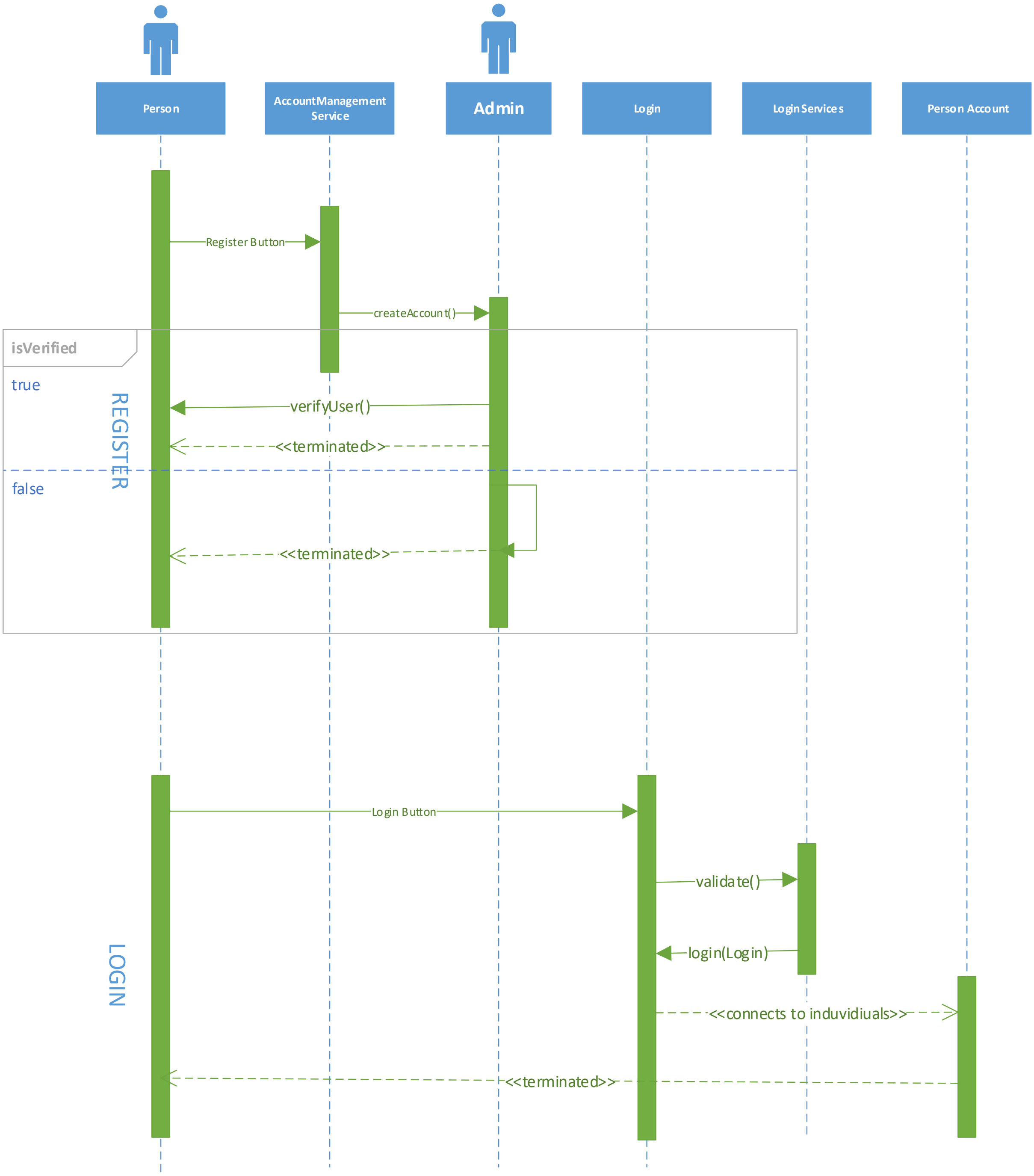


# OBJECT MODEL

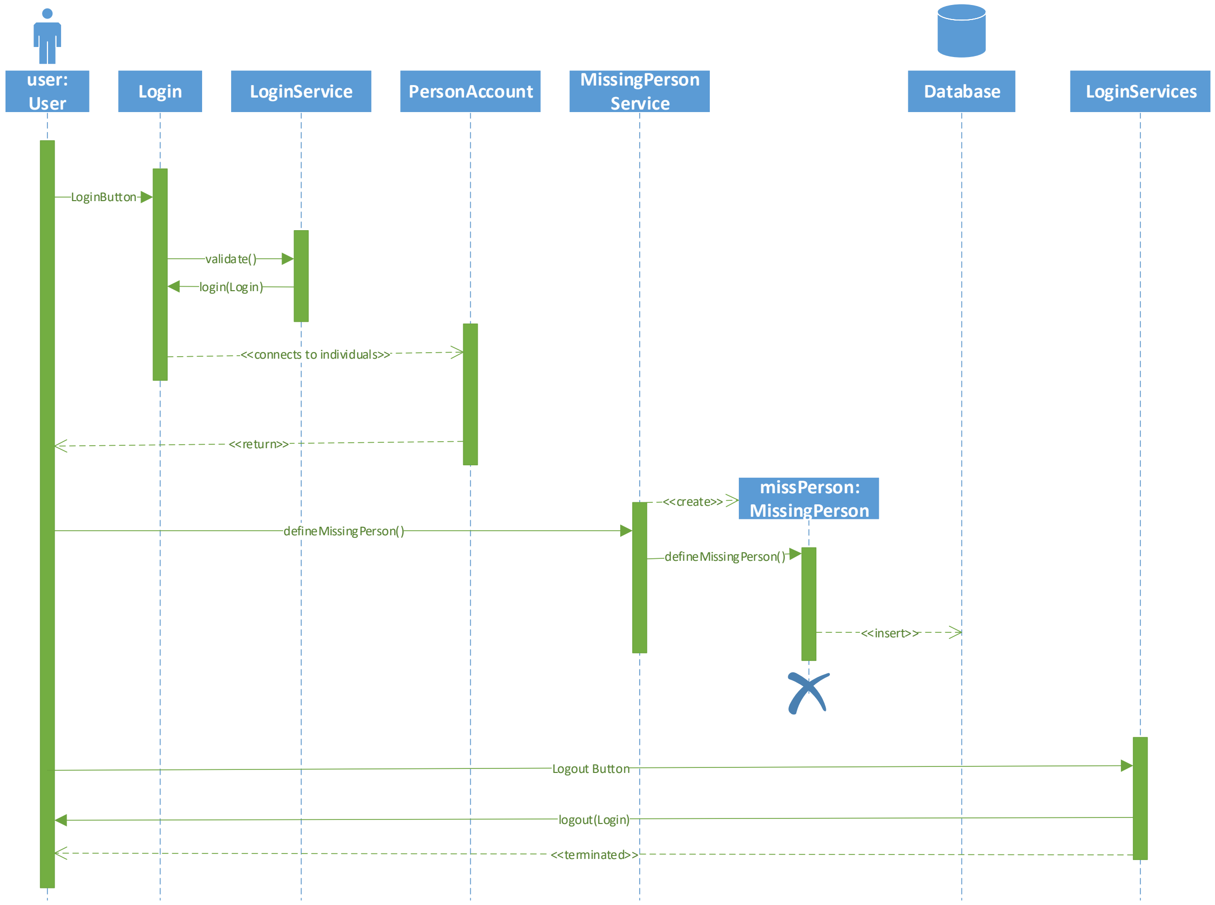


## SEQUENCE DIAGRAMS

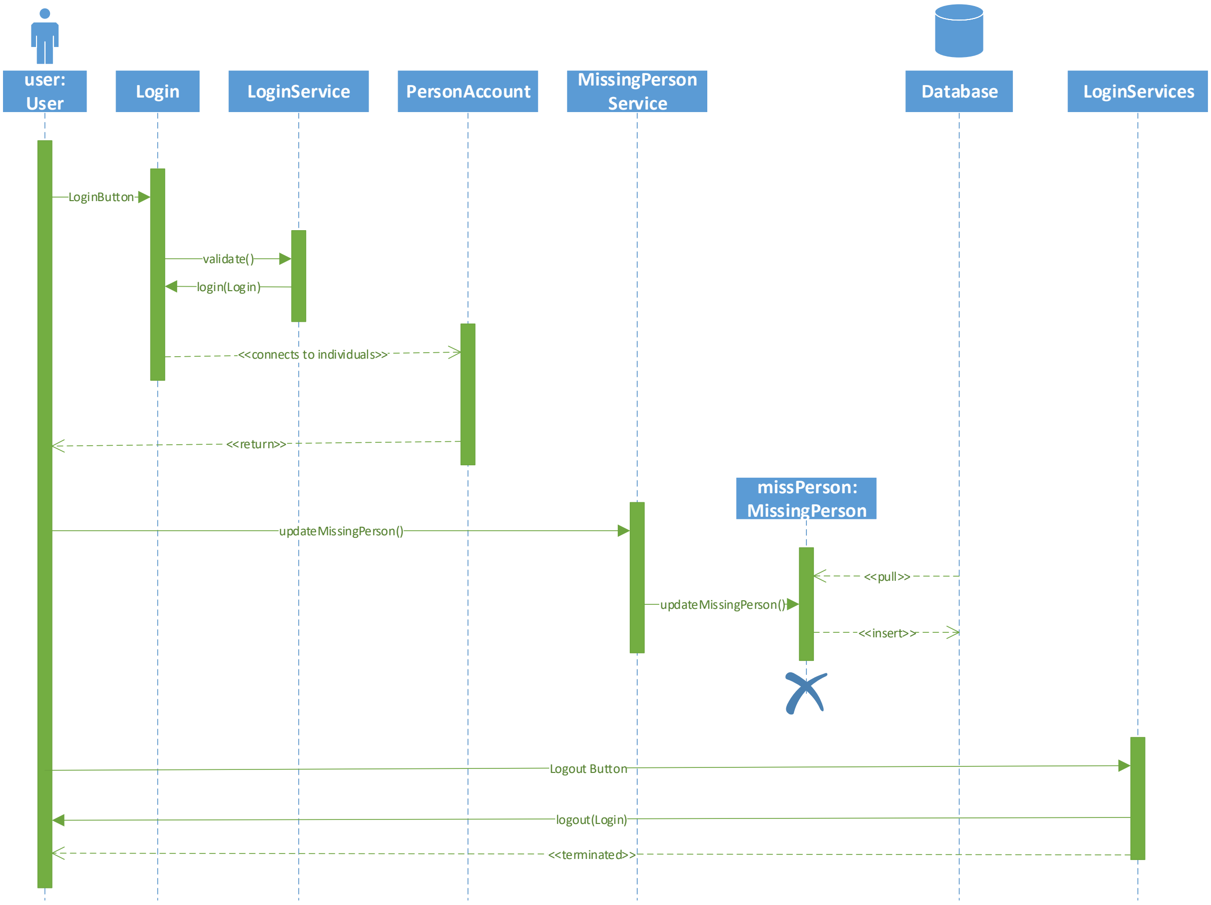
### 1. Account Transactions



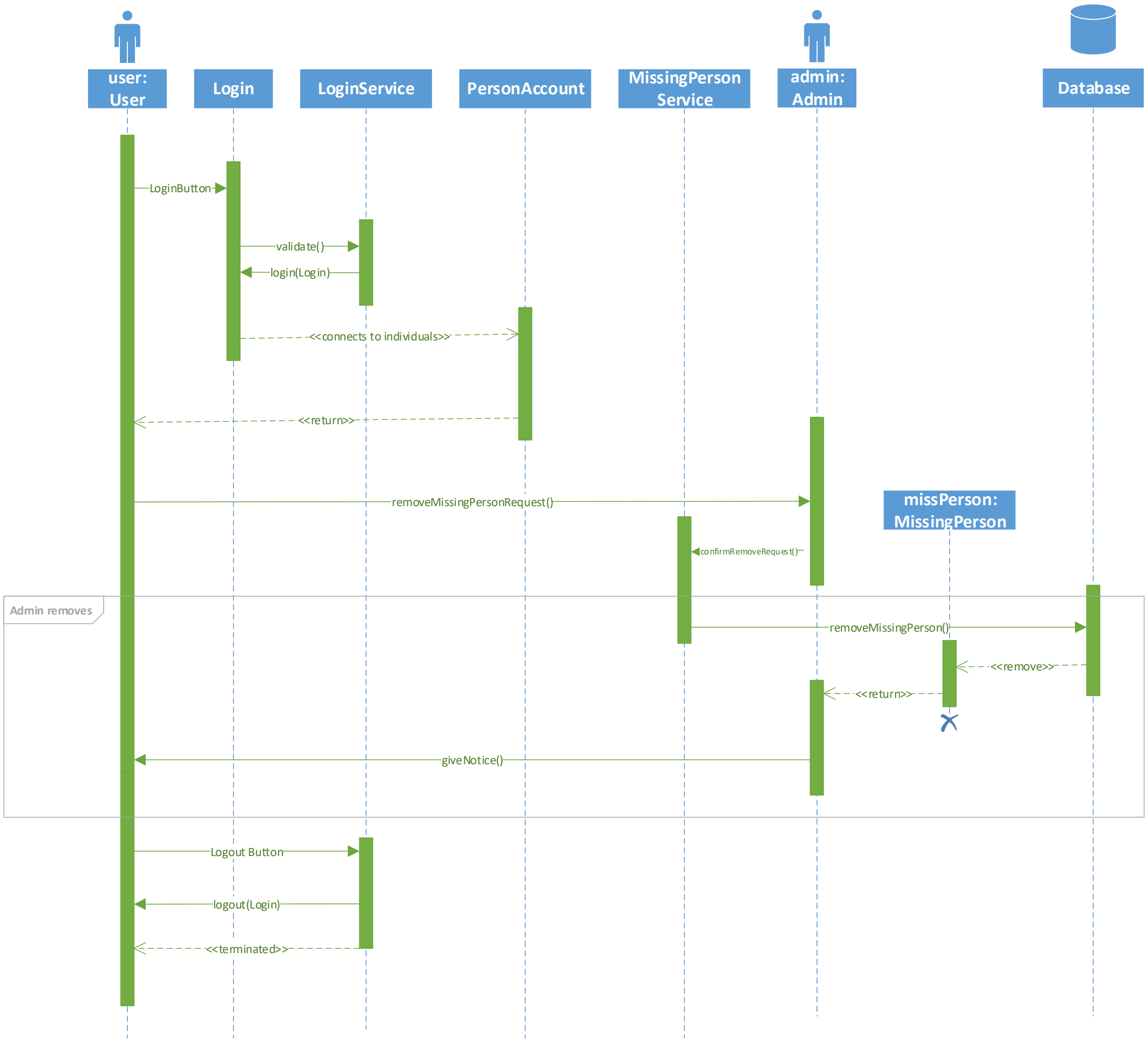
## 2. Define Missing Person



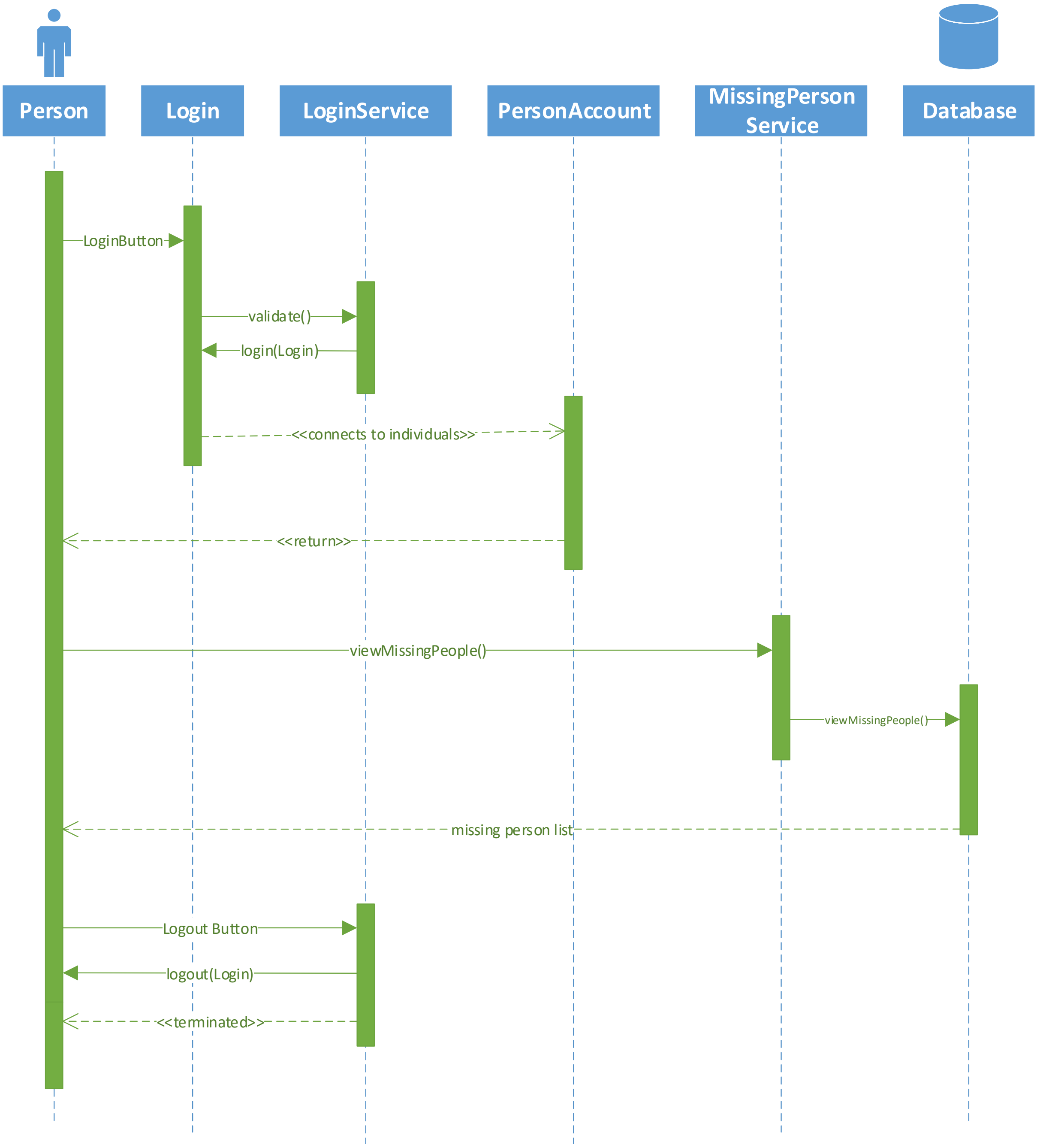
### 3. Update Missing Person



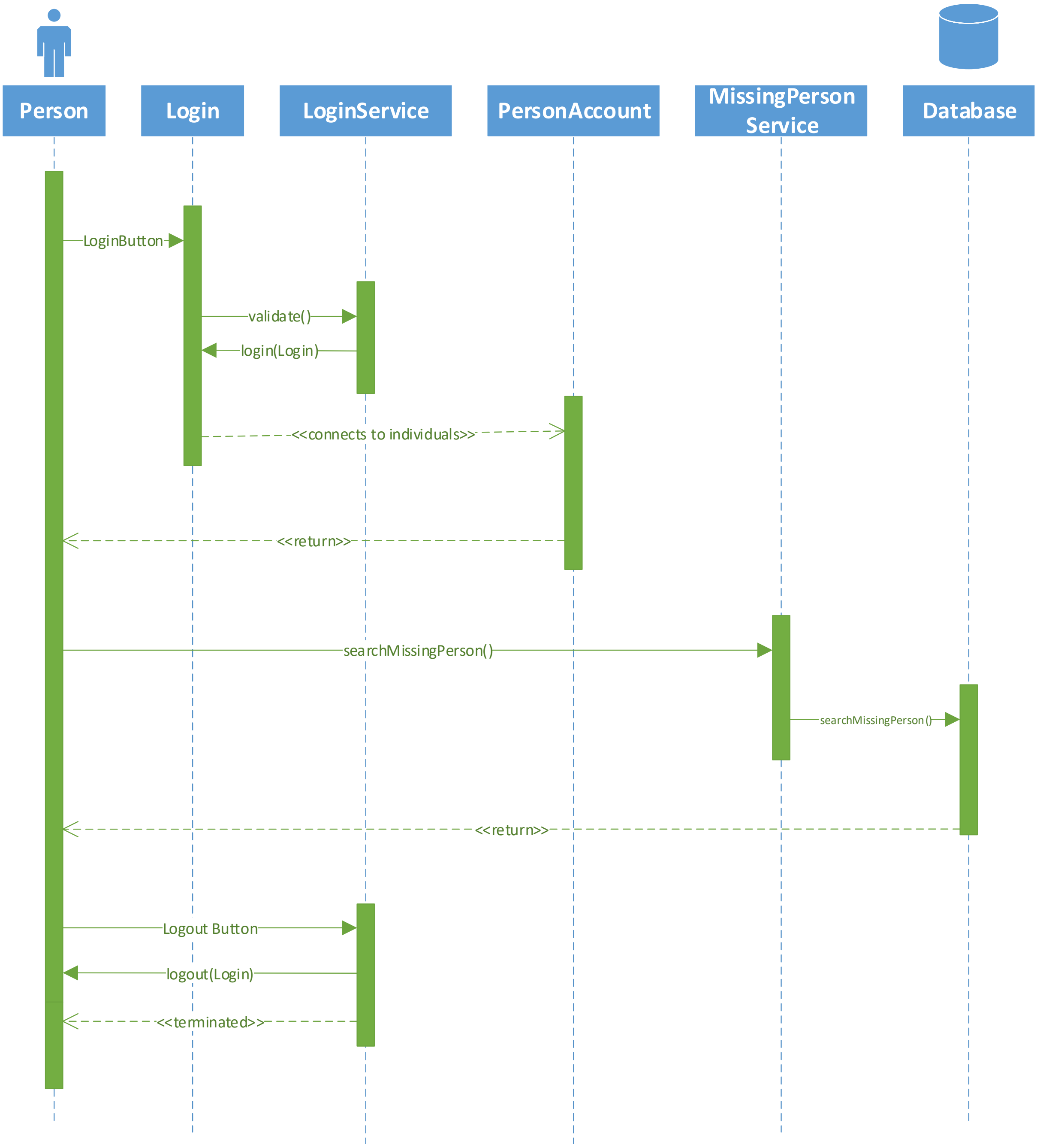
## 4. Remove Missing Person



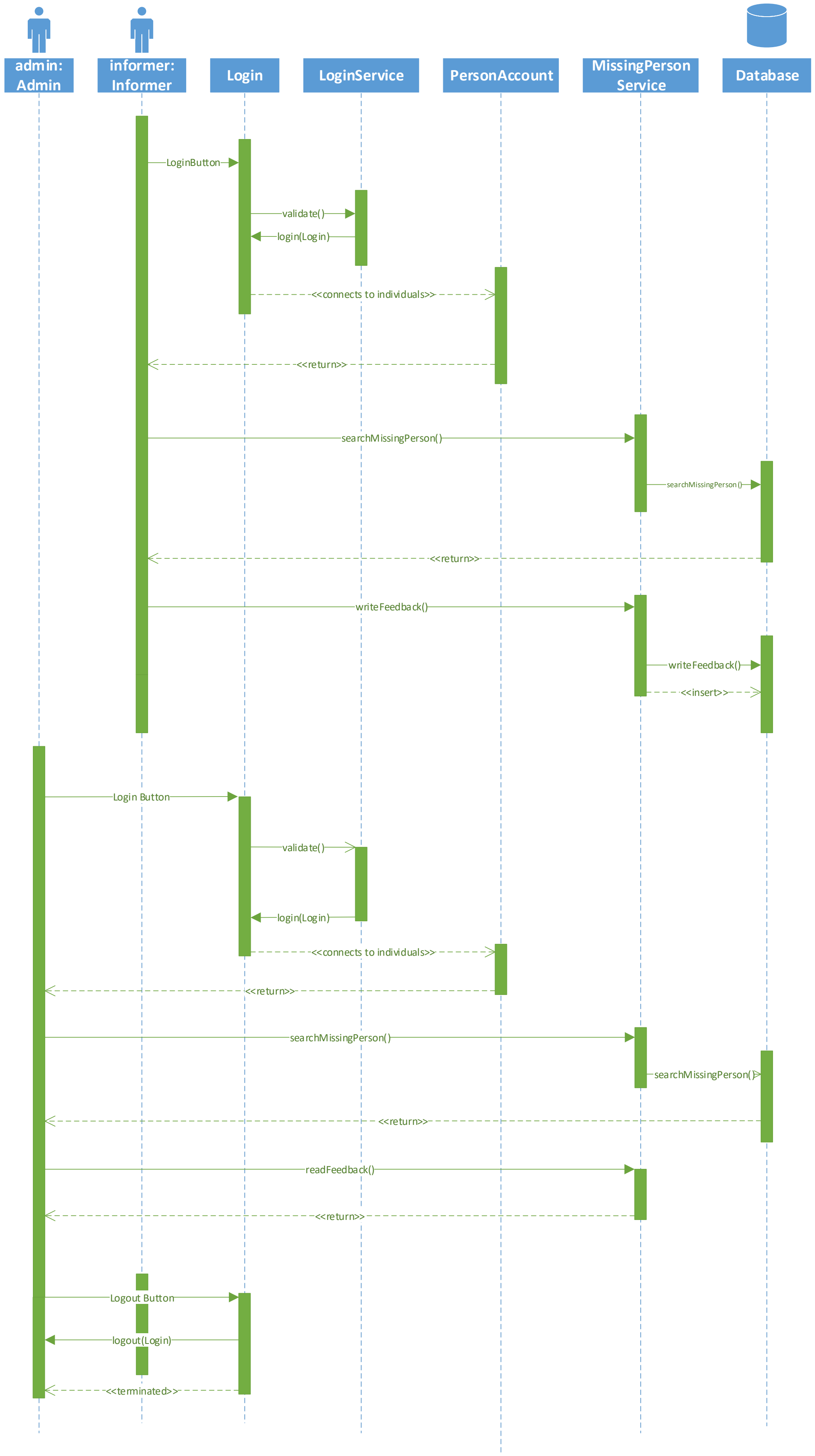
5. View Missing People



6. Search Missing Person

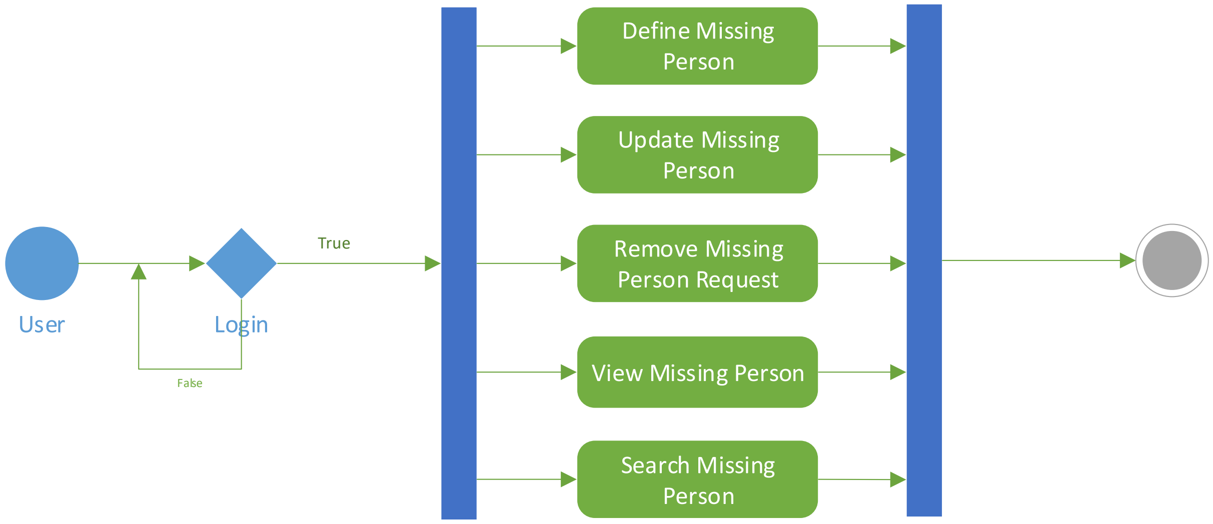


## 7. Write/Read Feedback

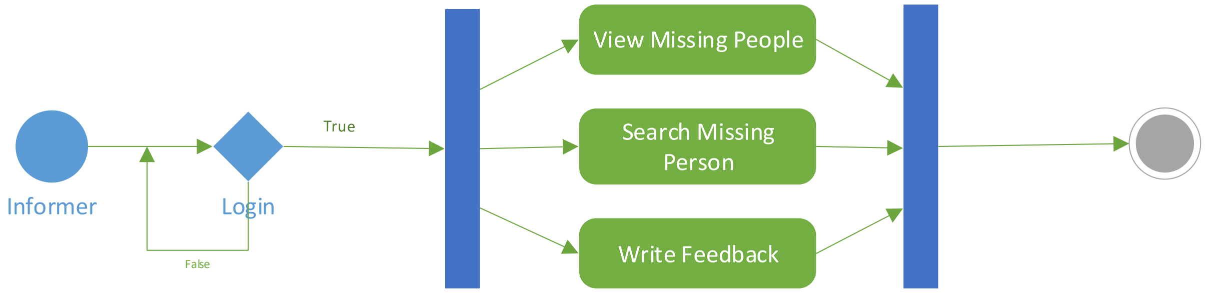


## Activity Diagrams

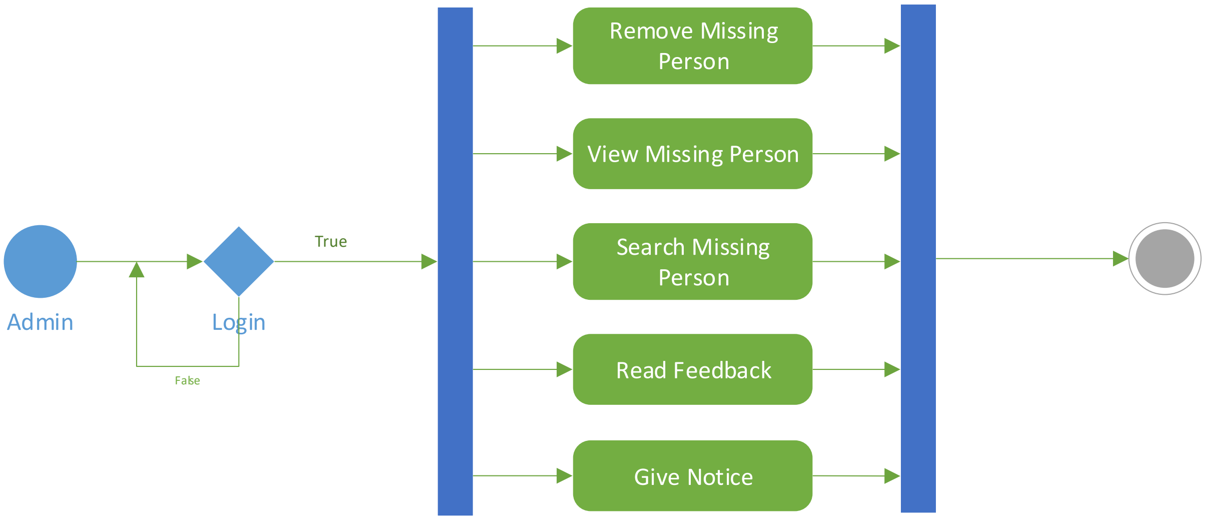
1. User Perspective Activity Diagram



2. Informer Perspective Activity Diagram



3. Admin Perspective Activity Diagram


Admin
Login
True
False
Remove Missing Person
View Missing Person
Search Missing Person
Read Feedback
Give Notice