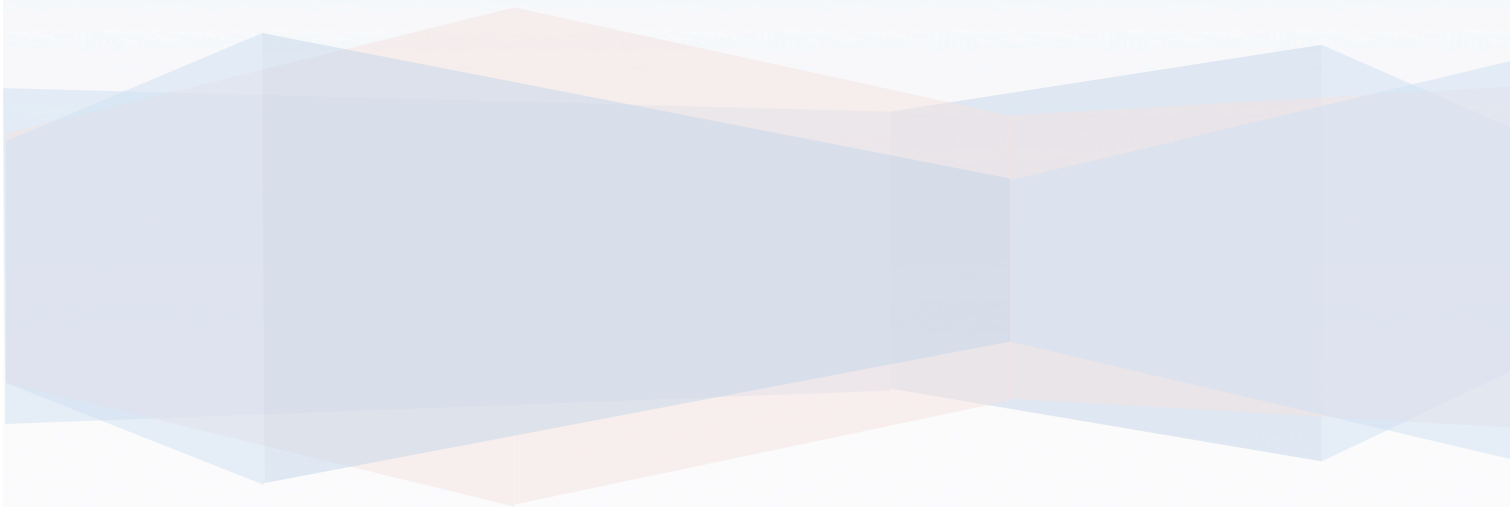


# Design Document

## Animal Tracker And Health Monitoring Application

Gearoid Lacey



## Table of Contents

<b>Abstract.....</b>	<b>3</b>
<b>Detailed Use Cases.....</b>	<b>4</b>
Register.....	4
Login .....	5
Logout.....	6
Change Password .....	7
Display Animal Path .....	8
Display Current Location .....	9
Isolate Animal .....	10
Add Animal Profile.....	11
Set GPS Intervals.....	12
Create Boundary.....	13
<b>System Sequence Diagrams .....</b>	<b>14</b>
Register System Sequence Diagram .....	14
Login System Sequence Diagram.....	15
Change Password System Sequence Diagram .....	16
Display Animal Path System Sequence Diagram .....	17
Display Current Location System Sequence Diagram .....	18
Isolate Animal System Sequence Diagram .....	19
Add Animal Profile System Sequence Diagram .....	20
Set GPS Intervals System Sequence Diagram .....	21
Create Boundary System Sequence Diagram .....	22

## **Abstract**

The purpose of this document is to provide a design specification for an animal tracker and health monitoring app. The aim of this document is to include detailed use cases, which describe the interactions between the application user, the application and the cloud. This document will also provide an overview of the layout of the applications code.

## Detailed Use Cases

### Register

**Name:** Register

**Actors:** Application User, Database System

**Description:** This use case begins when the application user chooses to create, an account on the android application. The application users will be required to create a user ID and password to register on the application. The use case ends when the application user has finished their work regarding their account.

#### Main Success Scenario:

1. The application user successfully opens the Animal Tracker And Health Monitoring Application.
2. The application displays the login/register screen.
3. The application user enters a user ID in the Username textbox.
4. The application user enters a password in the Password textbox.
5. The application user then clicks the Register button to submit their information.
6. The application then validates the information in the input fields
7. The database system then stores the inputted information.
8. The users information is added to the database.
9. The application then displays the home screen

#### Alternatives:

- 6a. The information in the input fields does not match the required format.
  1. The application user is prompted with an appropriate error.
  2. The application displays the change password screen again and are required to enter their information in the correct format and press the Change Password button
- 7a. The users credentials already exist in the database.
  1. The user is prompted with an appropriate error.
  2. The application reloads the login/register screen.
  3. The application user enters a user ID in the Username textbox.
  4. The application user enters a password in the Password textbox.
  5. The application user then clicks the Register button to resubmit their information.

## Login

**Name:** Login

**Actors:** Application User, Database System

**Description:** This use case begins when the application user opens the app. When the application user has opened the app they will be prompted to enter in their user ID and password. The user then clicks the login button located directly beneath the input fields and is redirected the application home screen.

### Main Success Scenario:

1. The application user successfully opens the Animal Tracker And Health Monitoring Application.
2. The application displays the login/register screen.
3. The application user enters a user ID in the Username textbox.
4. The application user enters a password in the Password textbox.
5. The application user then clicks the Login button to submit their information.
6. The application then validates the information in the input fields
7. The database system then checks the inputted information.
8. The application then displays the home screen

### Alternatives:

- 6a. The information input by the application user does not match the required format.
  1. The application user is prompted with an appropriate error
  2. The application displays the login/register screen again and the application user is required to enter their information in the correct format and press the Login button
- 7a. The users credentials do not exist in the database.
  1. The application user is prompted with an appropriate error.
  2. The application reloads the login/register screen
  3. The application user enters a user ID in the Username textbox.
  4. The application user enters a password in the Password textbox.
  5. The application user then clicks the Login button to resubmit their information

## Logout

**Name:** Logout

**Actors:** Application User, Database System

**Description:** This use case begins when the application user wishes to log out of the app. The application user will proceed by clicking on the menu button located on the top left corner of each screen. A menu bar will then appear on the left hand side of the screen with the logout option at the bottom of the menu. The application user will then click the logout option which will display a prompt requesting them to confirm that they wish to log out. The user will then click the confirm log out button on this prompt. The application user is now logged out of the application.

### **Main Success Scenario:**

1. The application user successfully navigates to the menu button in the top left corner of the screen.
2. The application user clicks the menu button.
3. A drop down menu is displayed.
4. The application user clicks the logout option on the drop down menu.
5. The application user is logged out.
6. The application displays the login/register screen.

### **Alternatives:**

- 5a. The application is unable to log out the application user.
  - 1 The application user is prompted with an appropriate error message.
  - 2 The application reloads the home screen

## Change Password

**Name:** Change Password

**Actors:** Application User, Database System

**Description:** This use case begins when the application user navigates to the login screen and selects the Change Password link. After clicking the Change Password link a new screen is displayed where the application user inputs their old password followed by their new one. The application user then clicks the Request Change button that changes their password and returns them to the login screen.

### Main Success Scenario:

1. The application user successfully opens the application.
2. The application displays the login/register screen.
3. The application user clicks the Change Password button.
4. The application displays three fields, the User ID field, an old password field and a new password field.
5. The application user enters their ID, old password and new password to the relevant fields.
6. The application user then clicks the Change Password button to submit their information.
7. The application validates the information input by the user.
8. The application users information is updated in the database
9. The application displays the login/register screen.

### Alternatives:

- 7a. The information input by the user does not match the required format.
1. The application user is prompted with an appropriate error
  2. The application displays the login/register screen again and the application user is required to enter their information in the correct format and press the Login button
- 8a. The information input by the user cannot be updated in the database as the user doesn't exist in the database.
1. The application user is prompted with an appropriate error.
  2. The application reloads the change password screen
  3. The application user enters a user ID in the Username textbox.
  4. The application user enters a password in the Password textbox.
  5. The application user then clicks the Login button to resubmit their information

## Display Animal Path

**Name:** Display Animal Path

**Actors:** Application User, Database System

**Description:** This use case begins when the application user selects the display animal path option in the application menu. The application user is then required to select the animal whose path data they wish to view. They then press the view button which displays the animals path data on a map.

### Main Success Scenario:

1. The application user successfully navigates to the menu button in the top left corner of the screen.
2. The application user clicks the menu button.
3. A drop down menu is displayed.
4. The application user clicks the Display Animal Path option on the drop down menu.
5. The application displays the Display Animal Path screen.
6. The application retrieves all animals associated with the current application user from the database.
7. The application user is prompted to select an animal from a drop down list.
8. The application user selects an animal from the drop down list.
9. The application retrieves all previous GPS information about the specified animal from the database.
10. The application user then presses the view button.
11. The application displays the retrieved information on a graph.

### Alternatives:

- 6a. No animals are associated with the current application user.
  1. The application user is prompted with an appropriate error message.
  2. The application reloads the Display Animal Path screen.
- 9a. No GPS information exists for the selected animal.
  1. The application user is prompted with an appropriate error message.
  2. The application user is prompted to select another animal from the drop down list of animals.



## Display Current Location

**Name:** Display Current Location

**Actors:** Application User, Database System

**Description:** This use case begins when the application user selects the display animal path option in the application menu. The application user is then required to select the animal whose current location they wish to view. They then press the view button which displays the animals current location data on a map.

### Main Success Scenario:

1. The application user successfully navigates to the menu button in the top left corner of the screen.
2. The application user clicks the menu button.
3. A drop down menu is displayed.
4. The application user clicks the Display Current Location option on the drop down menu.
5. The application displays the Display Current Location screen.
6. The application retrieves all animals associated with the current application user from the database.
7. The application user is prompted to select an animal from a drop down list.
8. The application user selects an animal from the drop down list.
9. The application retrieves the most recent GPS information about the specified animal from the database.
10. The application user then presses the view button.
11. The application displays the retrieved location on a graph.

### Alternatives:

- 6a. No animals are associated with the current application user.
  1. The application user is prompted with an appropriate error message.
  2. The application returns to the home screen.
- 9a. No GPS information exists for the selected animal.
  1. The application user is prompted with an appropriate error message.
  2. The application user is prompted to select another animal from the drop down list of animals.

## Isolate Animal

**Name:** Isolate Animal

**Actors:** Application User, Database System

**Description:** This use case begins when the application user selects the Isolate Animal functionality in the application menu. The application user selects the animal they wish to isolate. They then press the save button to add the animal to a list of sick/injured animals in isolation.

### Main Success Scenario:

1. The application user successfully navigates to the menu button in the top left corner of the screen.
2. The application user clicks the menu button.
3. A drop down menu is displayed.
4. The application user clicks the Isolate Animal option on the drop down menu.
5. The application displays the Isolate Animal screen.
6. The application retrieves all animals associated with the current application user from the database.
7. The application user is prompted to select an animal from a drop down list.
8. The application user selects an animal from the drop down list.
9. The application user then presses the Isolate button.
10. The application stores the animals details in an Isolation table in the database.

### Alternatives:

- 6a. No animals are associated with the current application user.
  1. The application user is prompted with an appropriate error message.
  2. The application returns to the home screen.

## **Add Animal Profile**

**Name:** Add Animal Profile

**Actors:** Application User, Database System

**Description:** This use case begins when the application user chooses to add an animal profile in the application. Details such as the animals tag number (optional), type, weight, breed, gender and age are stored for each animal. The use case ends when the application user has finished their work regarding their animals.

### **Main Success Scenario:**

1. The application user successfully navigates to the menu button in the top left corner of the screen.
2. The application user clicks the menu button.
3. A drop down menu is displayed.
4. The application user clicks the Add Animal Profile option on the drop down menu.
5. The application displays the Add Animal Profile screen.
6. The application user is then enters information about the animal such as the animals unique ID (a tag number if used on cattle), type (cat, dog or cow for example), weight, breed, gender and age into the appropriate fields.
7. The application user then clicks the Add Animal button
8. The application then validates the information in the input fields
9. The database system then stores the information input by the application user.

### **Alternatives:**

- 8a. The information in the input fields does not match the required format.
  1. The application user is prompted with an appropriate error.
  2. The application reloads the Add Animal Profile screen and the application user is required to re-enter the animals information
- 9a. The animals credentials already exist in the database.
  1. The user is prompted with an appropriate error.
  2. The application reloads the Add Animal Profile screen.

## Set GPS Intervals

**Name:** Set GPS Intervals

**Actors:** Application User, Tracking Device

**Description:** This use case begins when an application user wishes to set the GPS Intervals. From the drop down menu the application user will select the Set GPS Intervals option. The application user will then select an animal from a drop down list and enter the length of time they want between each tracking request. The use case ends when the application user has set the GPS Intervals.

### Main Success Scenario:

1. The application user successfully navigates to the menu button in the top left corner of the screen.
2. The application user clicks the menu button.
3. A drop down menu is displayed.
4. The application user clicks the Set GPS Intervals option on the drop down menu.
5. The application displays the Set GPS Intervals screen.
6. The application retrieves all animals associated with the current application user from the database which it populates in a drop down list.
7. The application user then selects an animal from the drop down list.
8. The application user then enters the length of time they want between each tracking request.
9. The application then validates the information in the input fields
10. The application user then clicks the save button
11. The application then sends the application users input via SMS to the tracking device.

### Alternatives:

- 6a. There are no animal profiles associated with the current application user.
  1. The application user is prompted with an appropriate error.
  2. The application returns to the Set GPS Intervals Screen.
- 9a. The application users input does not match the required format
  1. The application user is prompted with an appropriate error.
  2. The application returns to the Set GPS Intervals screen where the application user has to re-enter the information.

## Create Boundary

**Name:** Create Boundary

**Actors:** Application User, Database System, Tracking Device

**Description:** This use case begins when an application user wishes to set the boundary for their animal. From the drop down menu the application user will select the Create Boundary option. The application user will then select an animal from a drop down list and enter the radius of the boundary they want to set for the animal from their current location. The use case ends when the application user has created the Boundary for the specified animal.

### Main Success Scenario:

1. The application user successfully navigates to the menu button in the top left corner of the screen.
2. The application user clicks the menu button.
3. A drop down menu is displayed.
4. The application user clicks the Create Boundary option on the drop down menu.
5. The application displays the Create Boundary screen.
6. The application retrieves all animals associated with the current application user from the database which it populates in a drop down list.
7. The application user then selects an animal from the drop down list.
8. The application user then enters the size of the boundary their animal can travel.
9. The application then validates the information in the input fields
10. The application user then clicks the create boundary button
11. The database system then stores the information input by the application user.

### Alternatives:

6a. There are no animal profiles associated with the current application user.

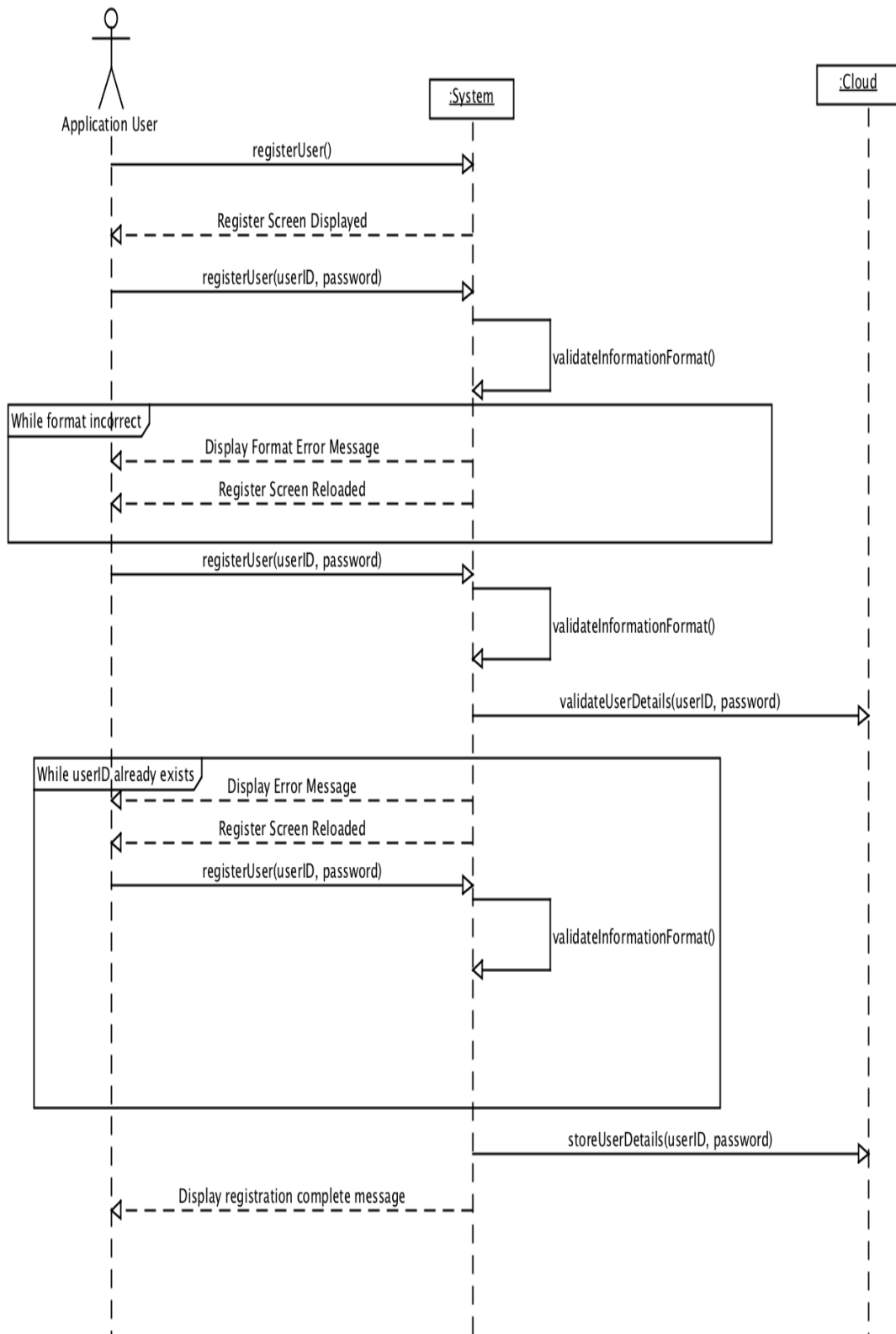
1. The application user is prompted with an appropriate error.
2. The application returns to the Create Boundary screen.

9a. The information in the input fields does not match the required format.

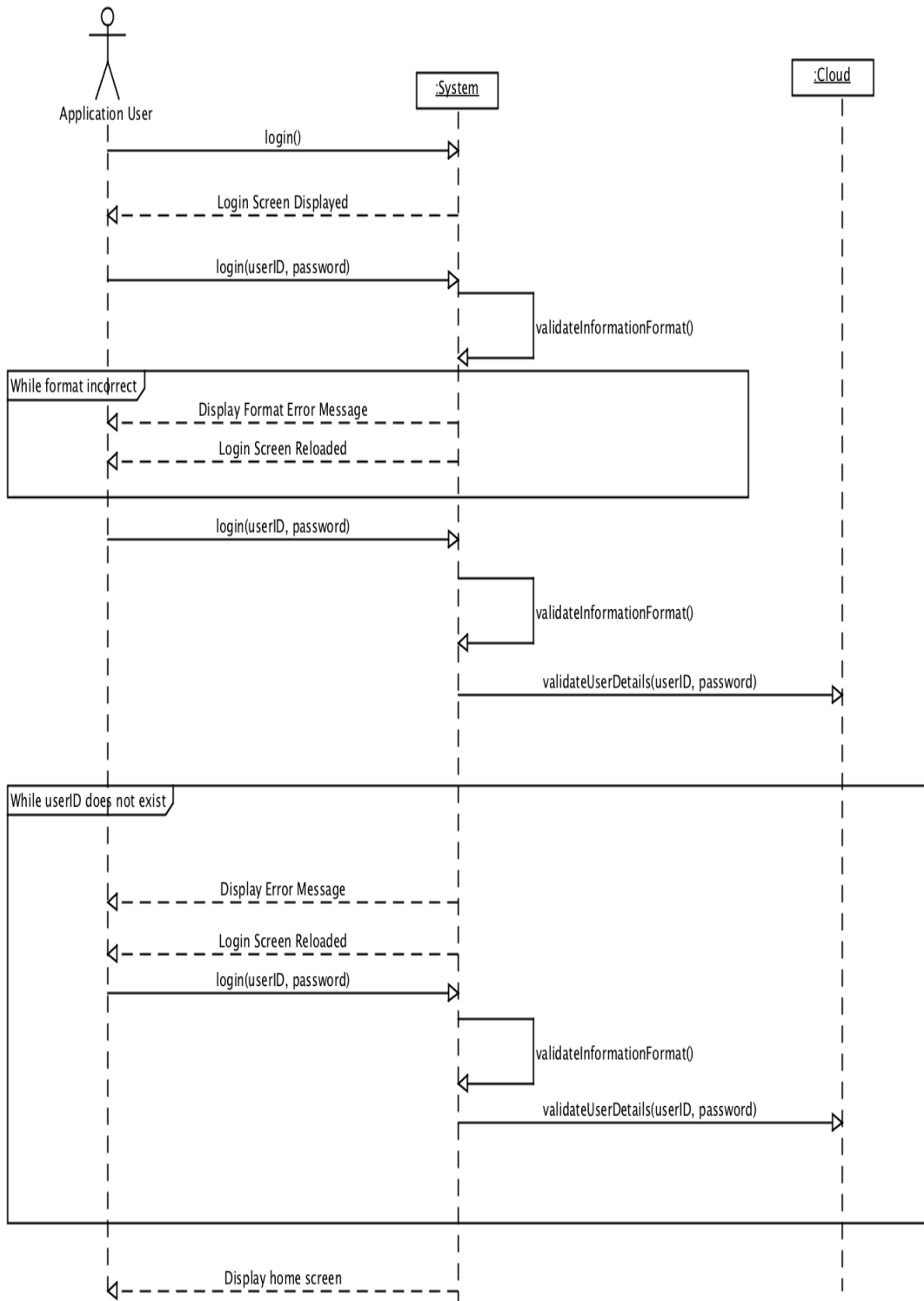
1. The application user is prompted with an appropriate error.
2. The application reloads the Create Boundary screen and the application user is required to re-enter the boundary information.

# System Sequence Diagrams

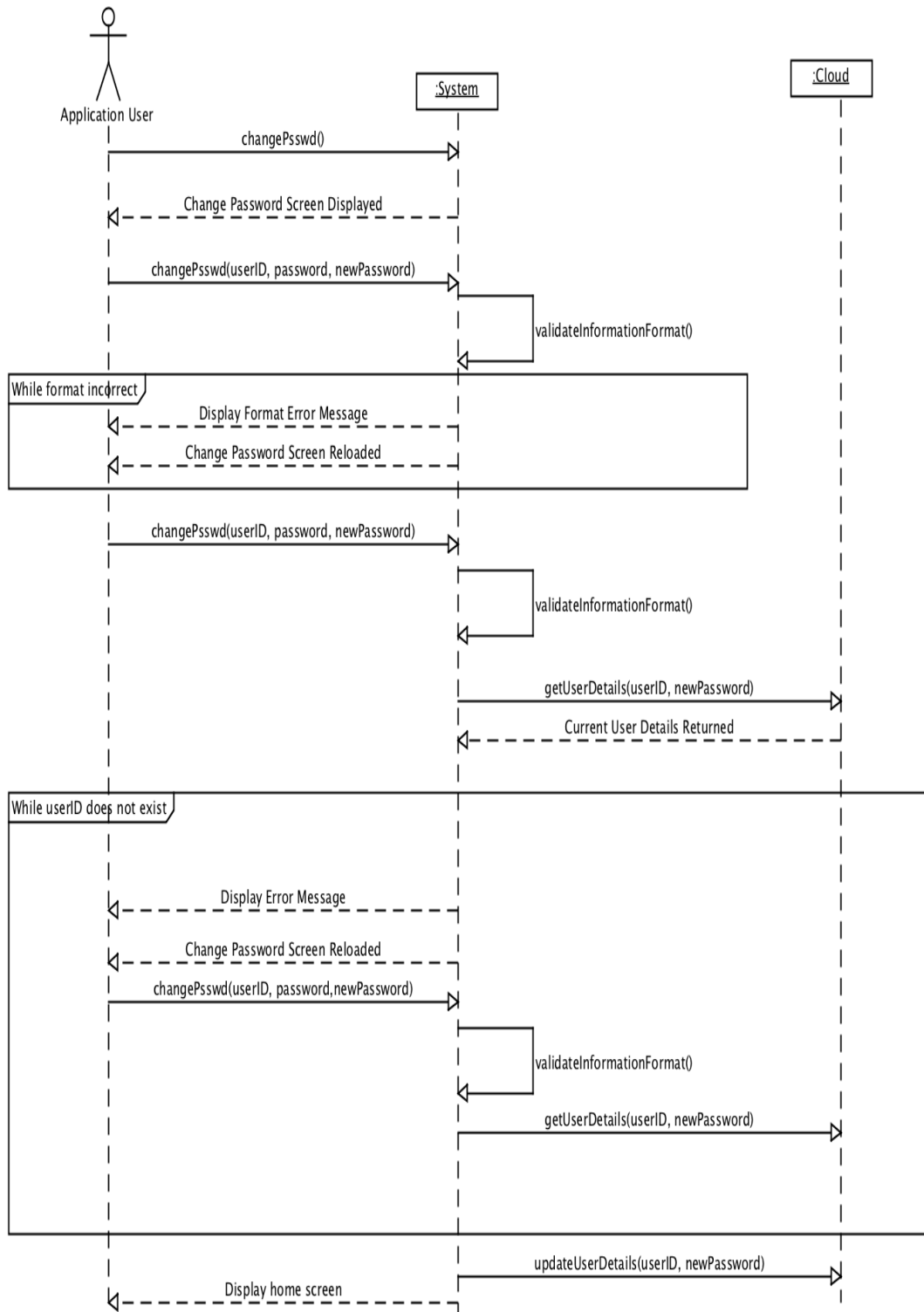
## Register System Sequence Diagram



# Login System Sequence Diagram

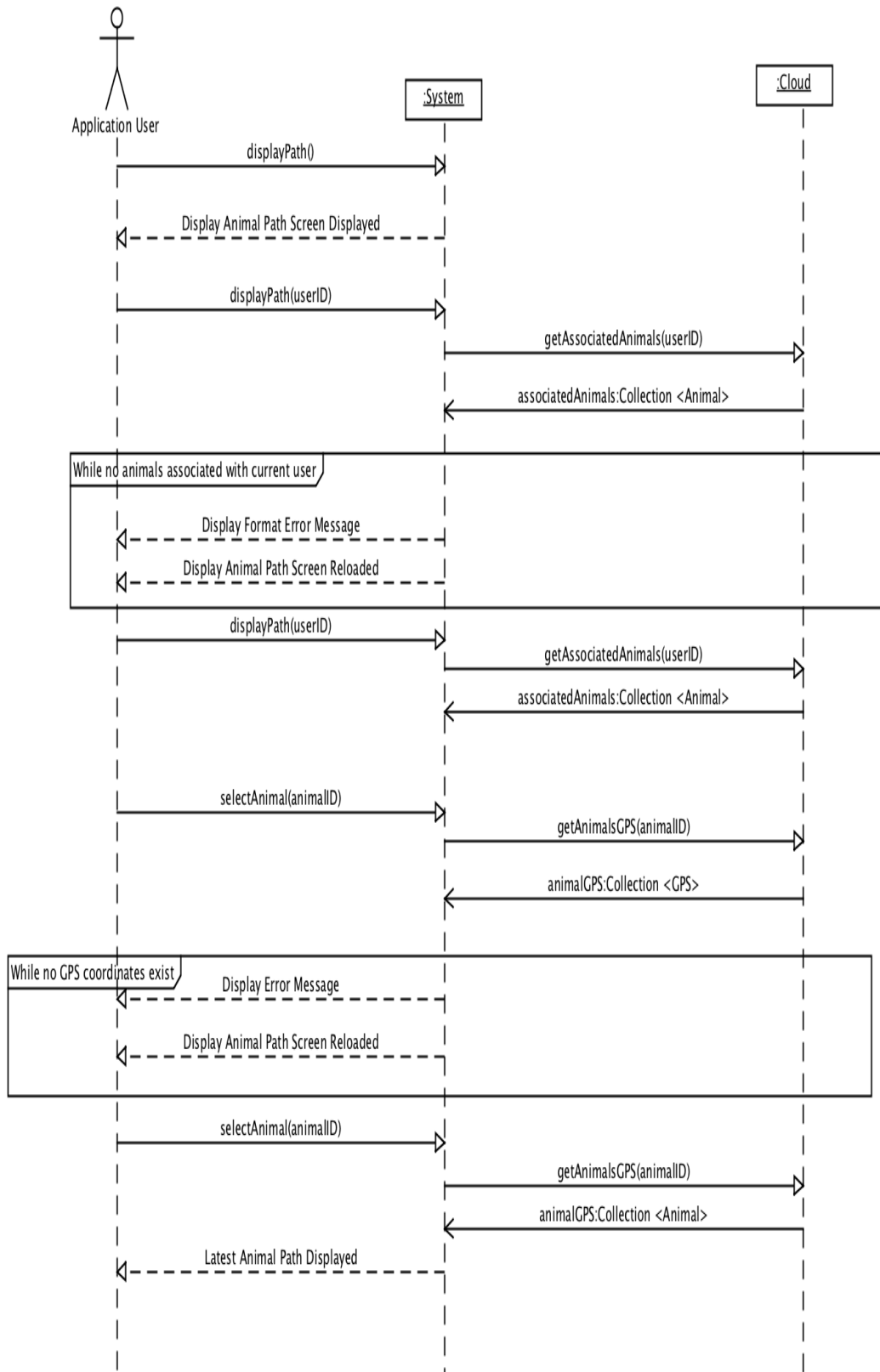


## Change Password System Sequence Diagram

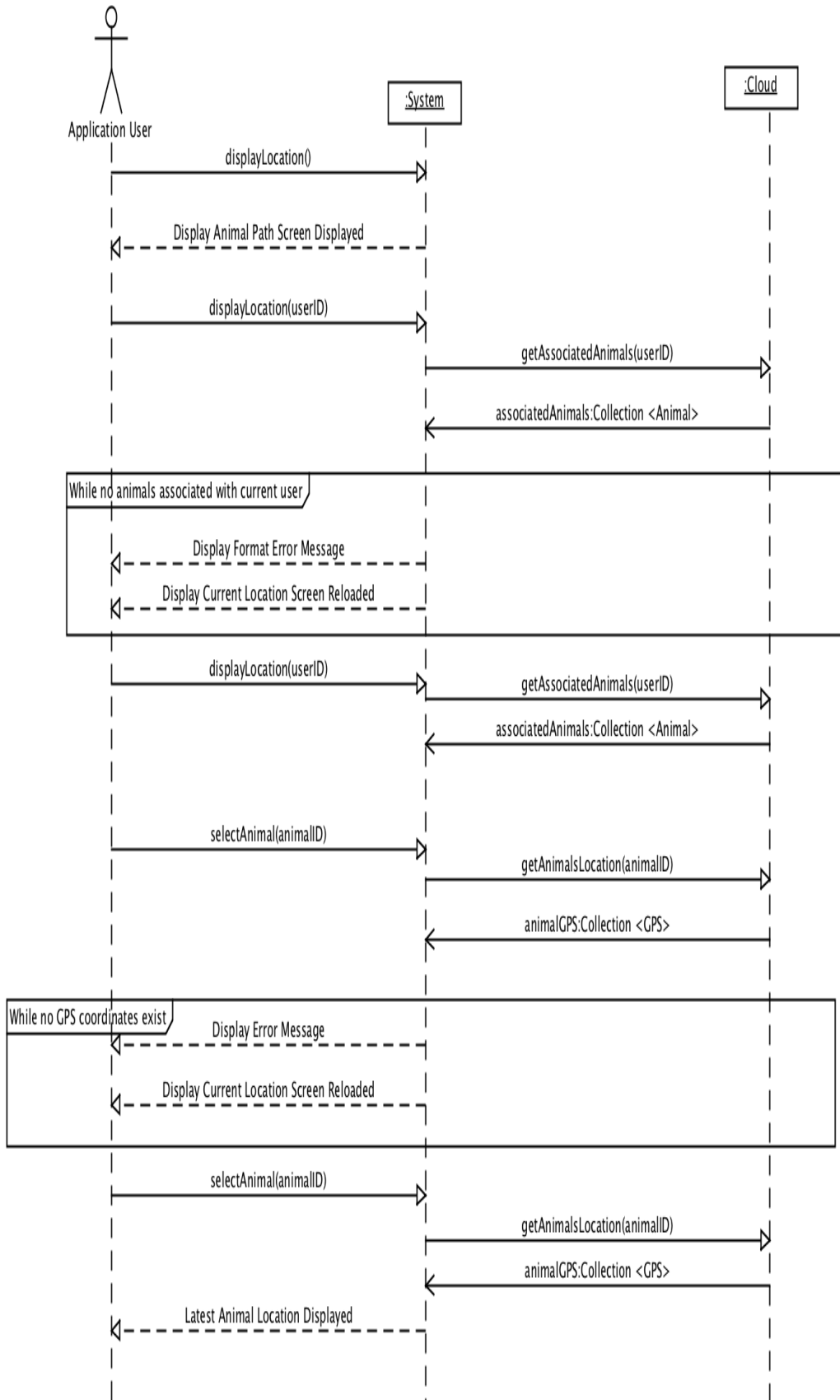




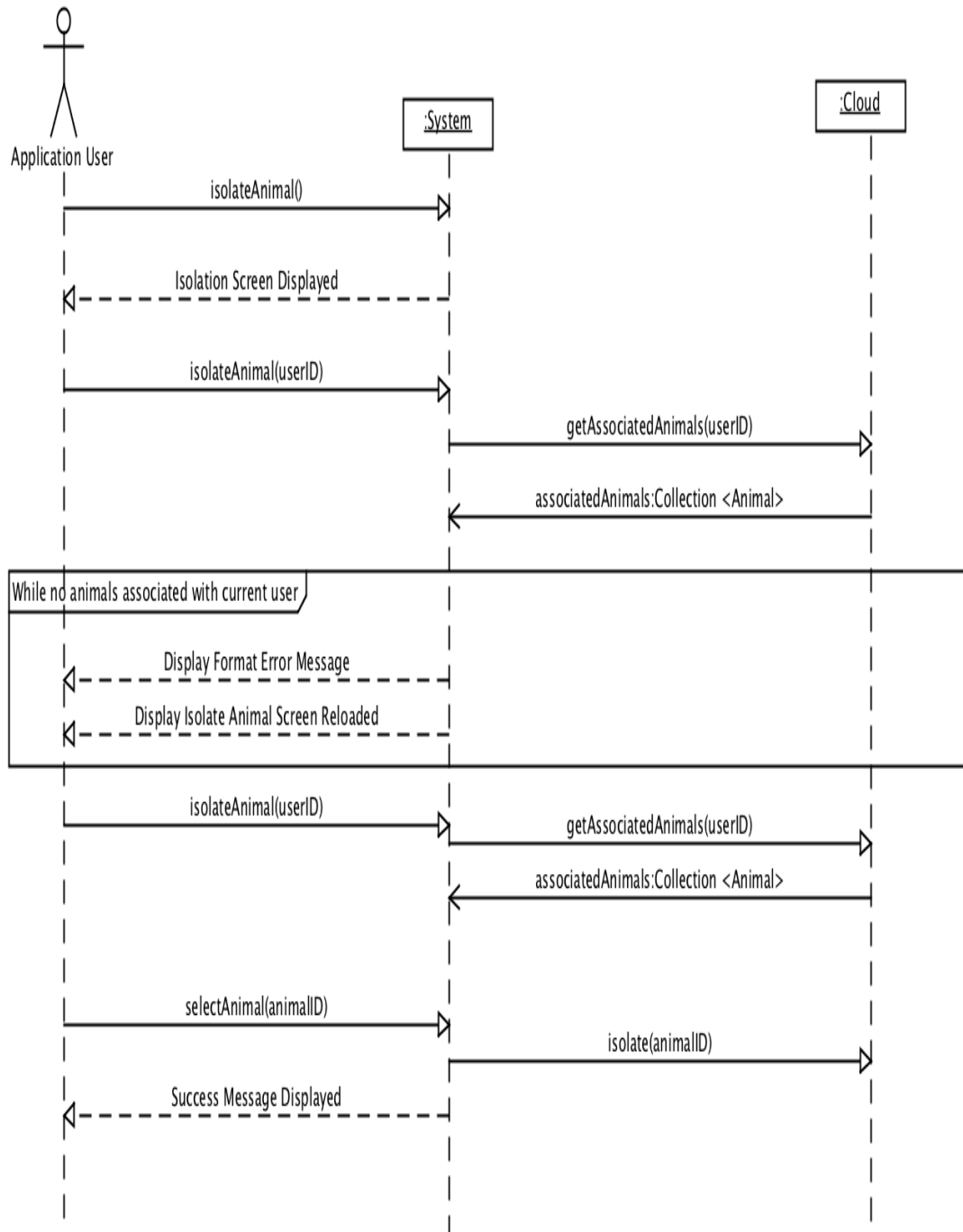
## Display Animal Path System Sequence Diagram



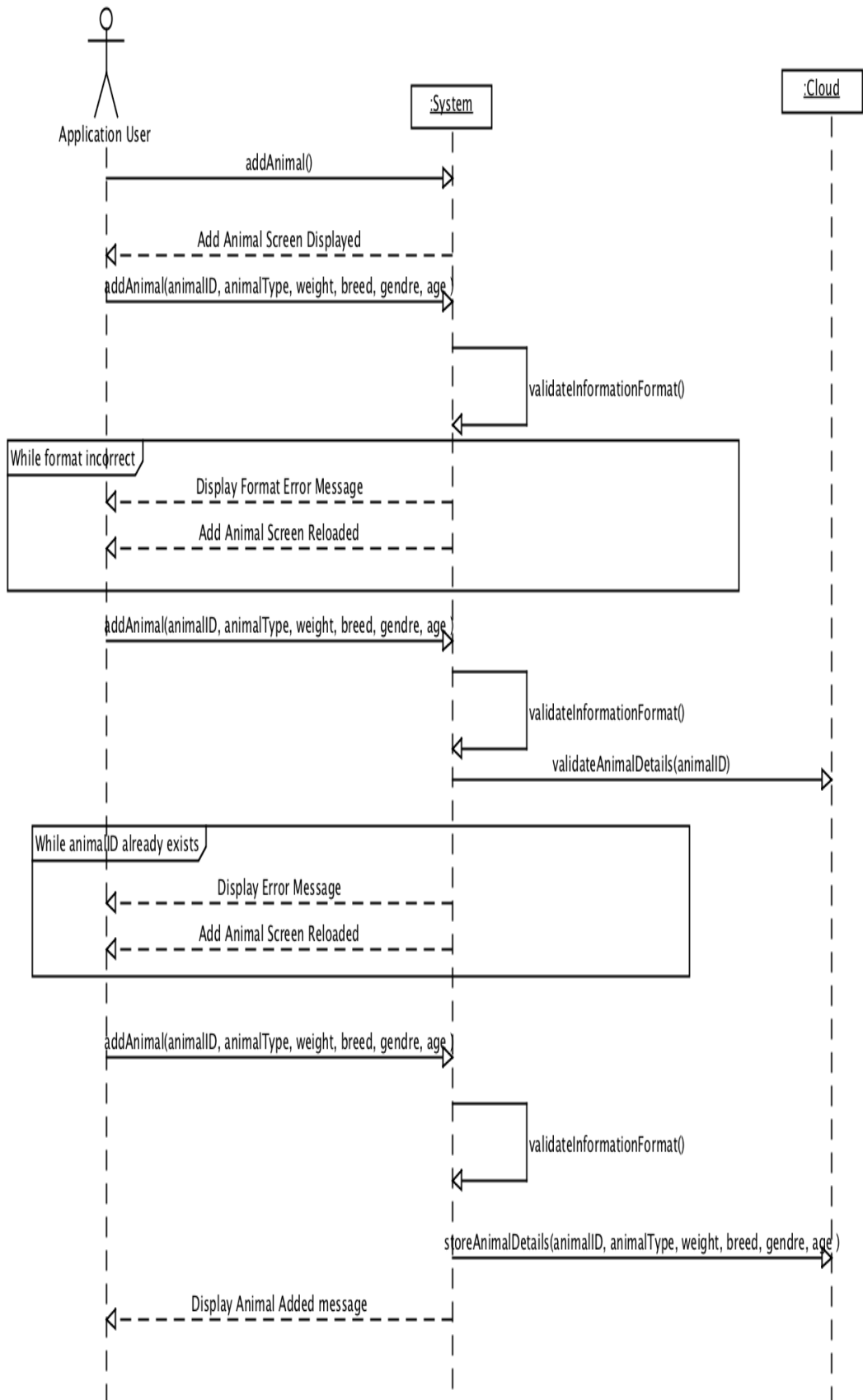
## Display Current Location System Sequence Diagram



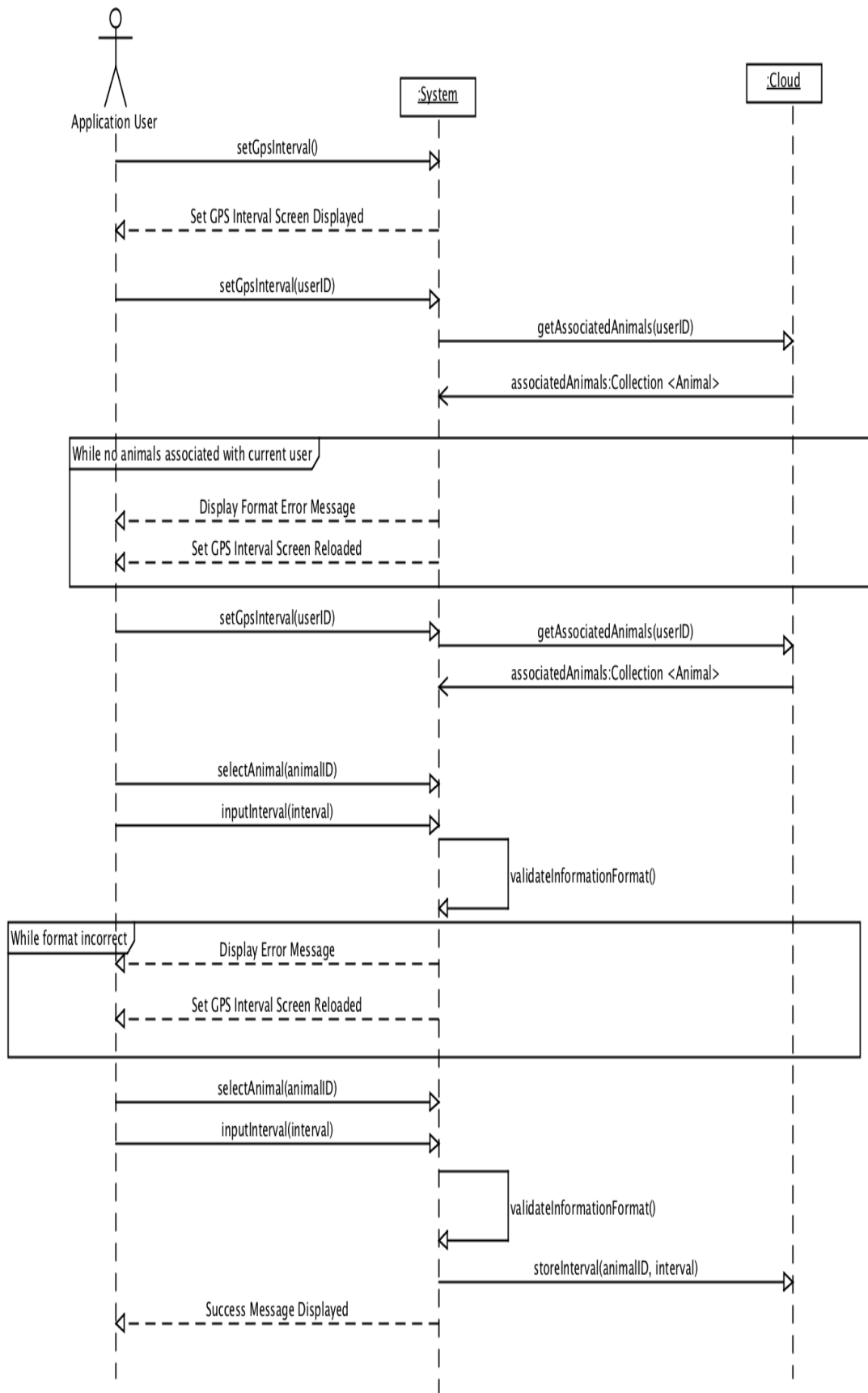
## Isolate Animal System Sequence Diagram

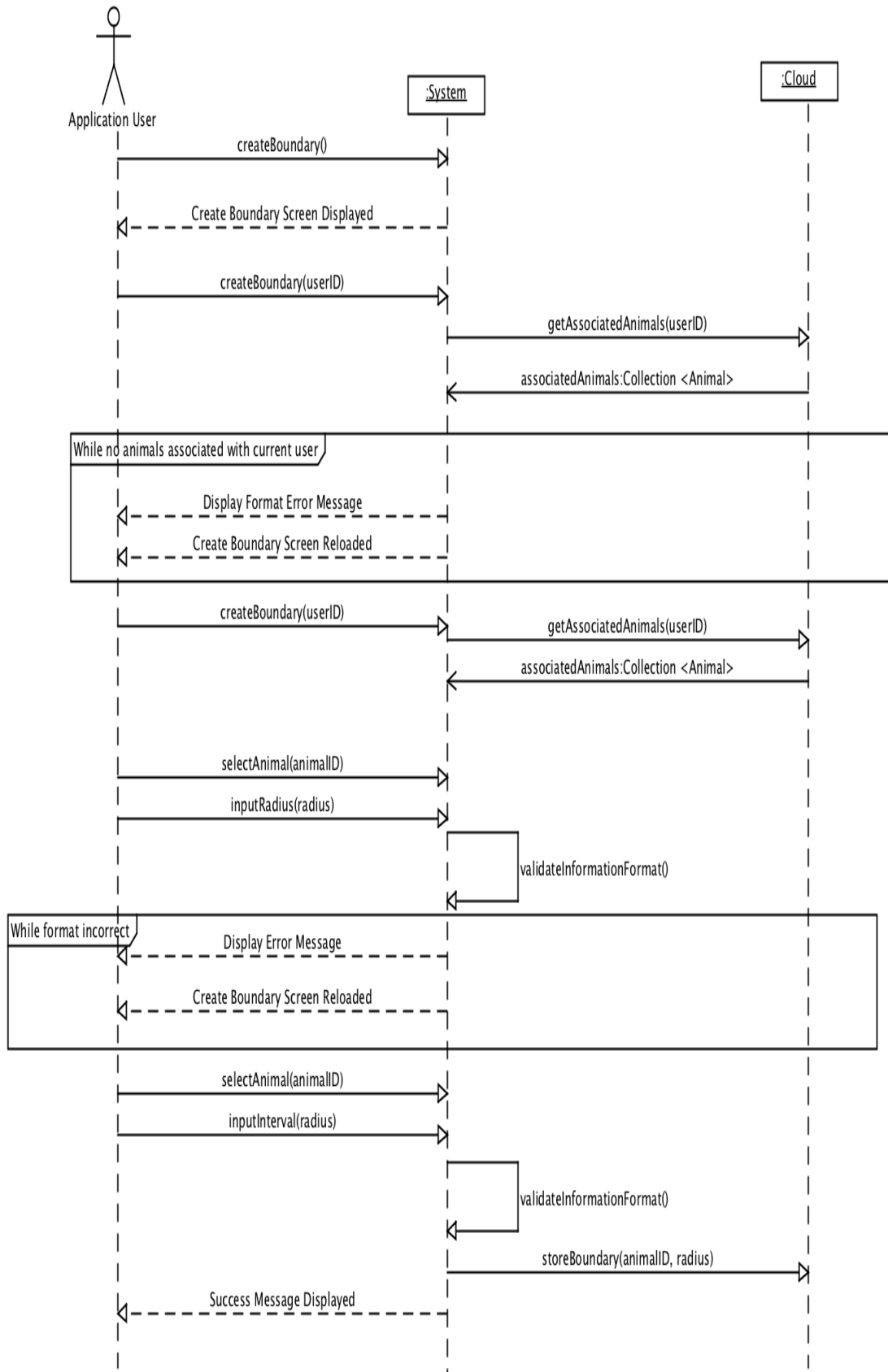


## Add Animal Profile System Sequence Diagram



## Set GPS Intervals System Sequence Diagram





## Database Design Diagram

