

## Pet Diaries

---

### ❖ Requirements:

#### 1. Manage Request:

- UR -013: Generate request
- UR-015: Respond to requests that turn up in their inbox based on preferences and location set while setting up profile
- UR- 019: Delete generated request
- UR -020: Edit generated request

#### 2. Authenticate Profile:

- UR- 008 : View new user profile requests on dashboard
- UR- 009 : Approve or reject new user requests

#### 3. Complete Transaction:

- UR -011: Give feedback to Caretaker
- UR- 018: Selectively block users based on negative feedback
- UR -026: Review feedback by Owner
- UR- 027: Assign Virtual Money
- UR -028: Use Virtual Money

#### 4. Maintain System:

- UR -010: View all activities between users
- UR -021: Profiles can be selectively deactivated
- UR -022: Requests can be selectively deleted

### ❖ Use Case Documents

<b>Use Case ID:</b>	UC-04
<b>Use Case Name:</b>	Manage Request
<b>Description:</b>	Owner can create request for his pets and admin manages all the requests

<b>Actors:</b>	Owner, Caretaker, Admin
<b>Pre-conditions:</b>	Owner wants a caretaker to take care of his pets and a request page is displayed
<b>Post-conditions:</b>	Admin allows only valid requests

<b>Frequency of Use:</b>	As need arises by the owner		
<b>Flow of Events:</b>		<b>Actor Action</b>	<b>System Response</b>
	1	Owner generates a request	Check if request is valid and within limit and Publish it
	2	Owner can Edit the request	Make required changes in the request
	3	Owner can delete the request	Deletes the request from system
	4	Admin can check and delete the request	Deletes the request from the system
<b>Variations:</b>	Admin can view all requests and check them from Database		
<b>Exceptions:</b>			
<b>Developer Notes:</b>			

<b>Use Case ID:</b>	UC-05
<b>Use Case Name:</b>	Authenticate Profile
<b>Description:</b>	Owner and Caretaker create their profiles and admin does checks for the validity of their profiles

<b>Actors:</b>	Owner, Caretaker, Admin		
<b>Pre-conditions:</b>	Owner and Caretaker wish to participate and create profiles		
<b>Post-conditions:</b>	Admin allows only valid profiles to stay in system		
<b>Frequency of Use:</b>	When a new profile is created		
<b>Flow of Events:</b>		<b>Actor Action</b>	<b>System Response</b>
	1	Owner/Caretaker create a new profile	Allot a userId to the and save the information in database
	2	Admin checks the profile and verifies the email	Sends an email to user to verify email
	3	Admin rejects the profile request if email not verified	Deletes the userId and profile from the system
	4	Admin accepts the profile and connects it to userId	Profile information si stored with userId in the database
	5	Admin notifies owner of request accepted	

<b>Variations:</b>	Owner and Caretaker have different pages to generate profiles
<b>Exceptions:</b>	
<b>Developer Notes:</b>	

<b>Use Case ID:</b>	UC-06
<b>Use Case Name:</b>	Complete Transaction
<b>Description:</b>	Owner sends feedback, admin checks feedback, if positive sends virtual money to caretaker

<b>Actors:</b>	Owner, Caretaker, Admin		
<b>Pre-conditions:</b>	Caretaker meets request and Owner provides a feedback		
<b>Post-conditions:</b>	Feedback is valid and positive		
<b>Frequency of Use:</b>	When a request is met and feedback is provided		
<b>Flow of Events:</b>		<b>Actor Action</b>	<b>System Response</b>
	1	Owner provides a feedback	Takes in the feedback and adds it to database
	2	Admin checks whether the feedback is positive	Checks whether the dates have passed the request date
	3	Admin assigns virtual money to caretaker	Assign virtual money
	4	Caretaker can use the virtual money	Virtual money balance is maintained
<b>Variations:</b>	Concept of virtual money is redeeming points		
<b>Exceptions:</b>			
<b>Developer Notes:</b>			

<b>Use Case ID:</b>	UC-07
<b>Use Case Name:</b>	Maintain System
<b>Description:</b>	Admin checks for active accounts

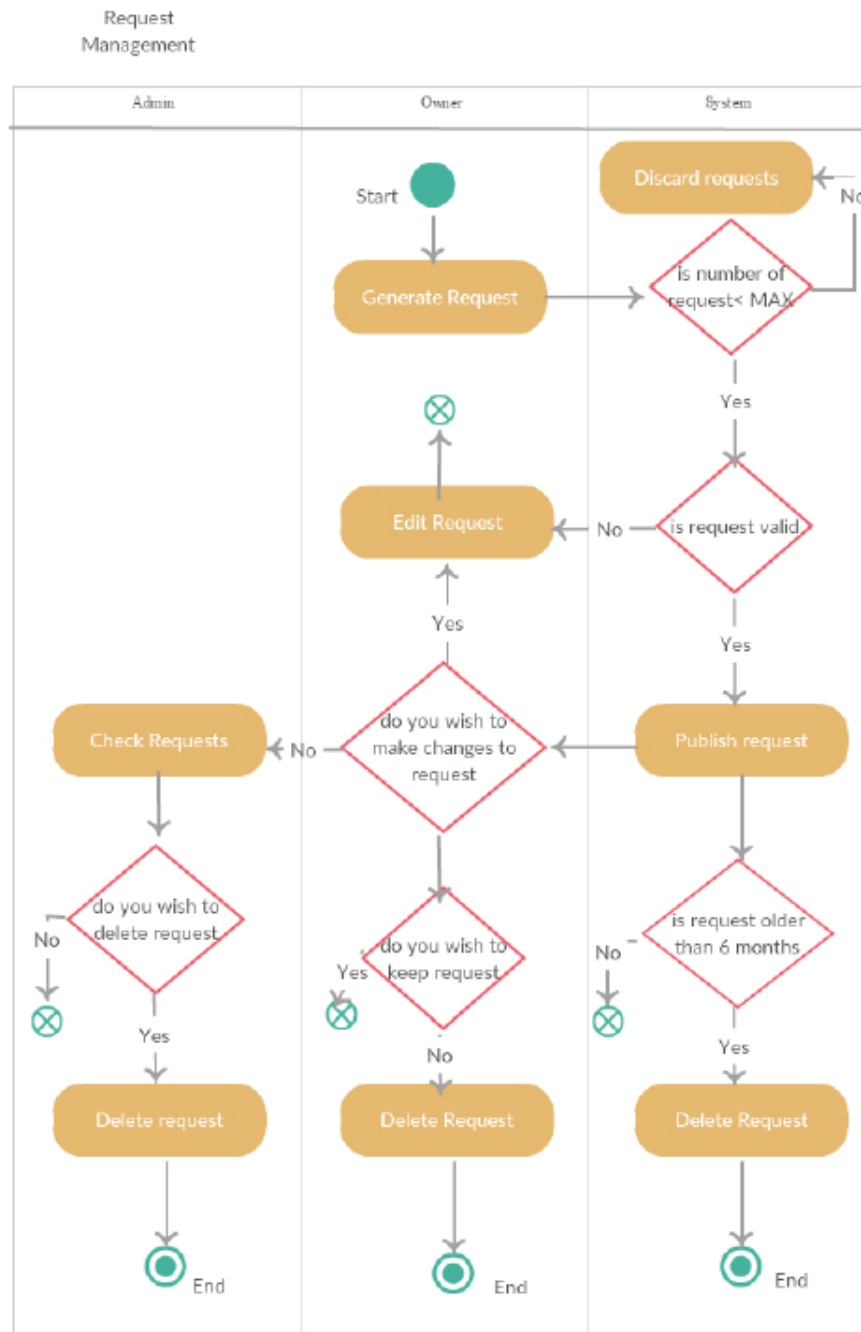
<b>Actors:</b>	Owner, Caretaker, Admin
<b>Pre-</b>	Owner and caretaker are regular users

<b>conditions:</b>			
<b>Post-conditions:</b>	The account should not be inactive for more than six months		
<b>Frequency of Use:</b>	When a request is met and feedback is provided		
<b>Flow of Events:</b>		<b>Actor Action</b>	<b>System Response</b>
	1	Admin checks the profile activity in database	Marks a profile for deletion
	2	Admin sends a reminder to user to activate their profile and marks the expiry of it upto 2 days from the email sent	Sends an email to the owner/caretaker
	3	Owner and caretaker choose whether to respond to the email	Database is updated for activity if the user logs in
	4	Admin updates the database	
	5	Admin deletes the user profile from database	Inactive user profile is deleted from system
<b>Variations:</b>	MySQL database used		
<b>Exceptions:</b>			
<b>Developer Notes:</b>			

## ❖ Activity Diagrams

### 1. Manage Request:

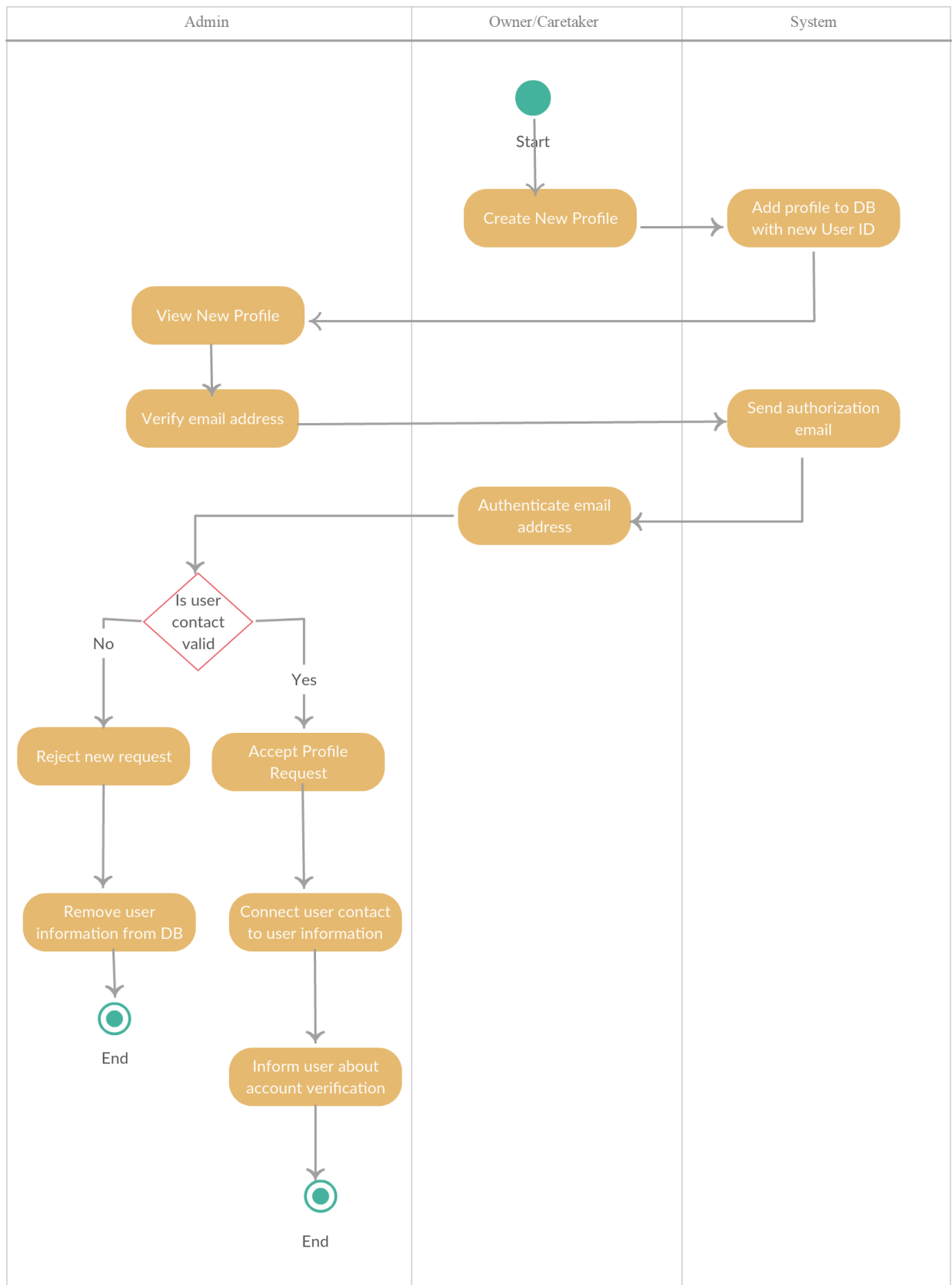
- Requirement IDs: UR-013, UR-015, UR-019, UR-020
- Use Case ID# : UC- 04
- Use Case description: Manage Request (Owner can create request for his pets and admin manages all the Requests)
- Name : Gayatri Mestry



## 2. Authenticate Profile:

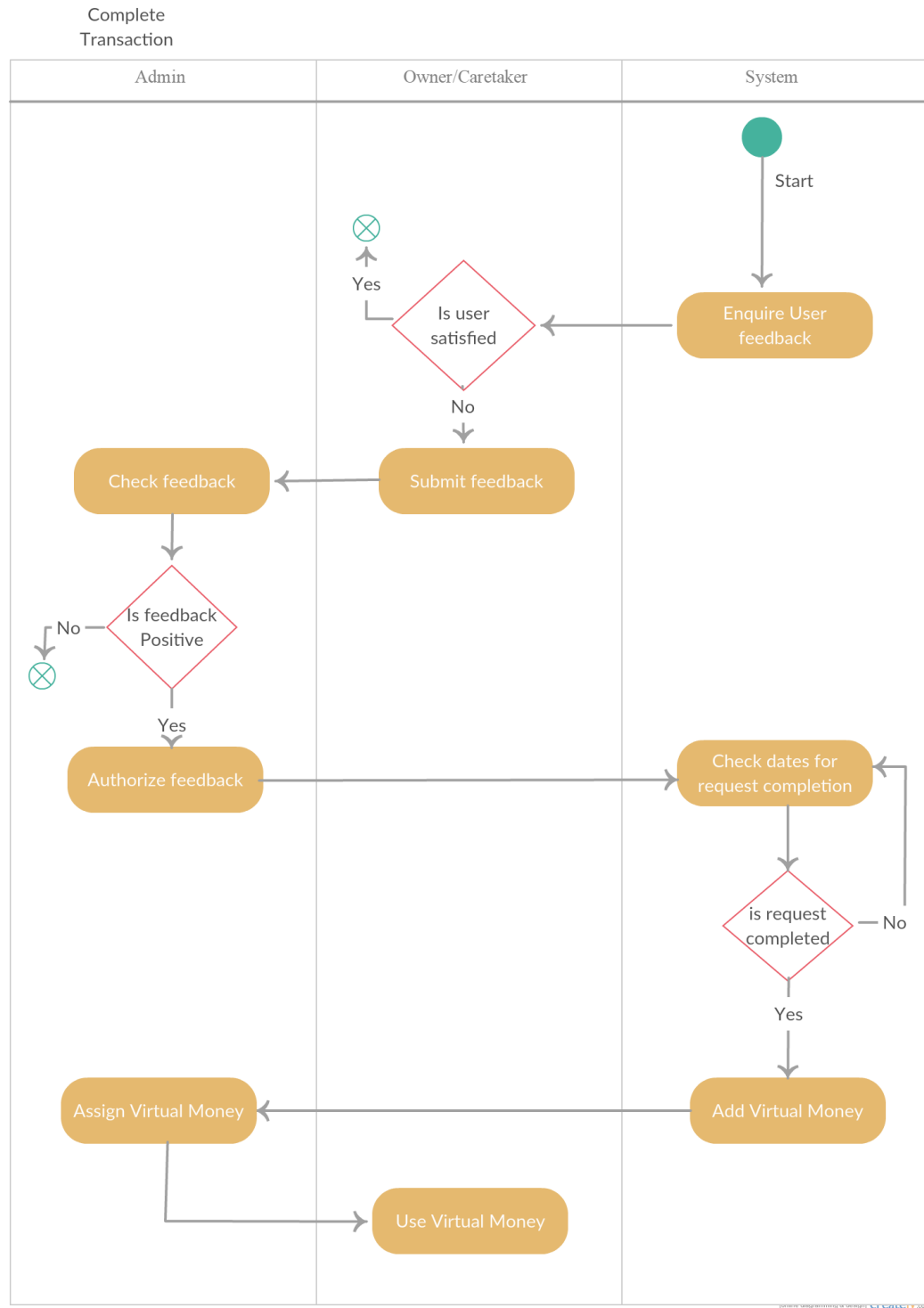
- Requirement IDs: UR-008, UR-009
- Use Case ID# : UC- 05
- Use Case description: Authenticate Profile (Owner and Caretaker create their profiles and admin does checks for the validity of their profiles)
- Name : Gayatri Mestry

## Authenticate Profile



3. Complete Transaction:

- Requirement IDs: UR-011, UR-018, UR-026, UR-027, UR-028
- Use Case ID# : UC- 06
- Use Case description: Complete Transaction (Owner sends feedback, admin checks  
feedback, if positive sends virtual money to caretaker)
- Name : Gayatri Mestry

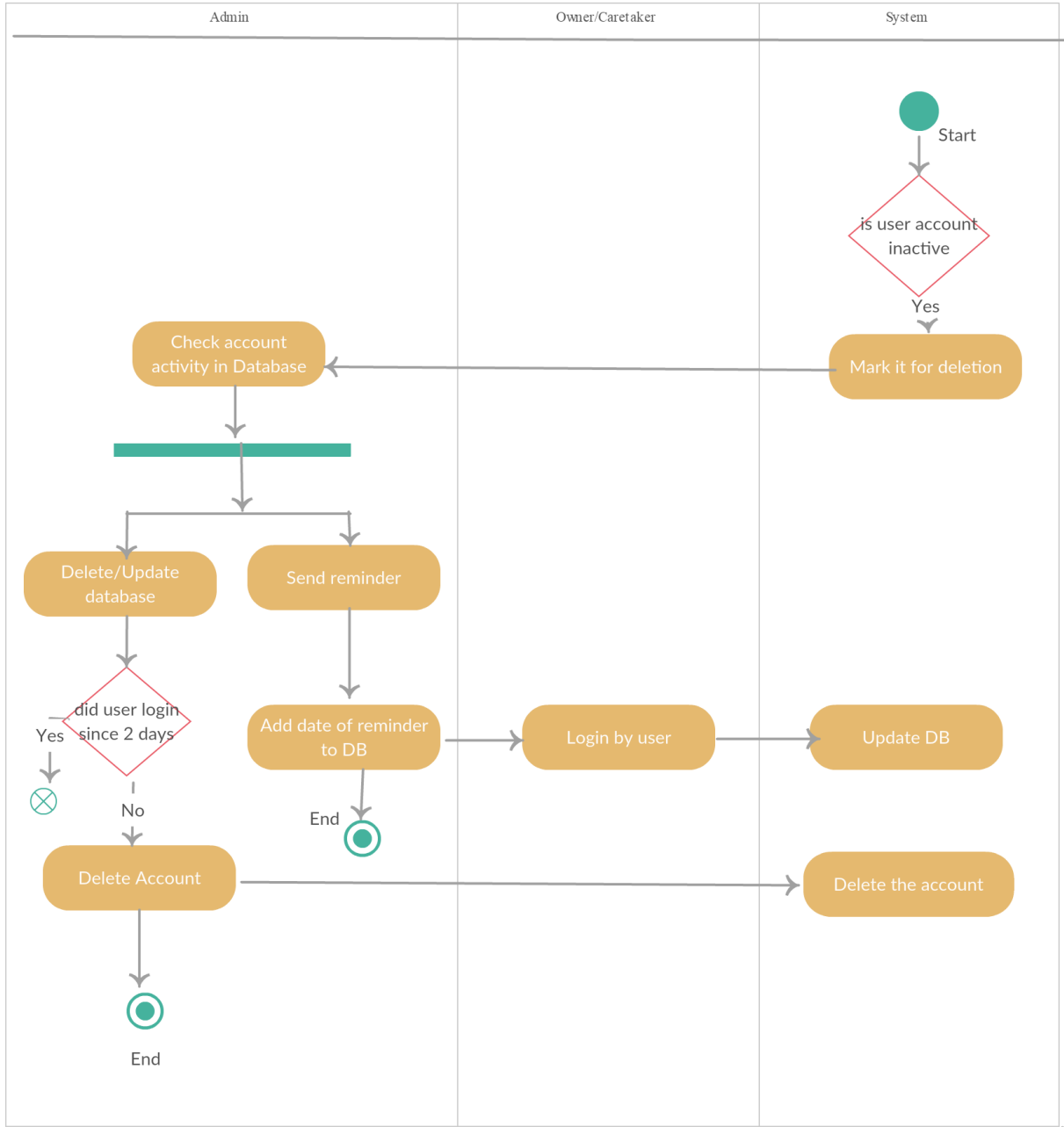


#### 4. Maintain System:

- Requirement IDs: UR-010, UR-021, UR-022
- Use Case ID# : UC- 07
- Use Case description: Complete Transaction (Admin checks for active accounts)
- Name : Gayatri Mestry



## Maintain System

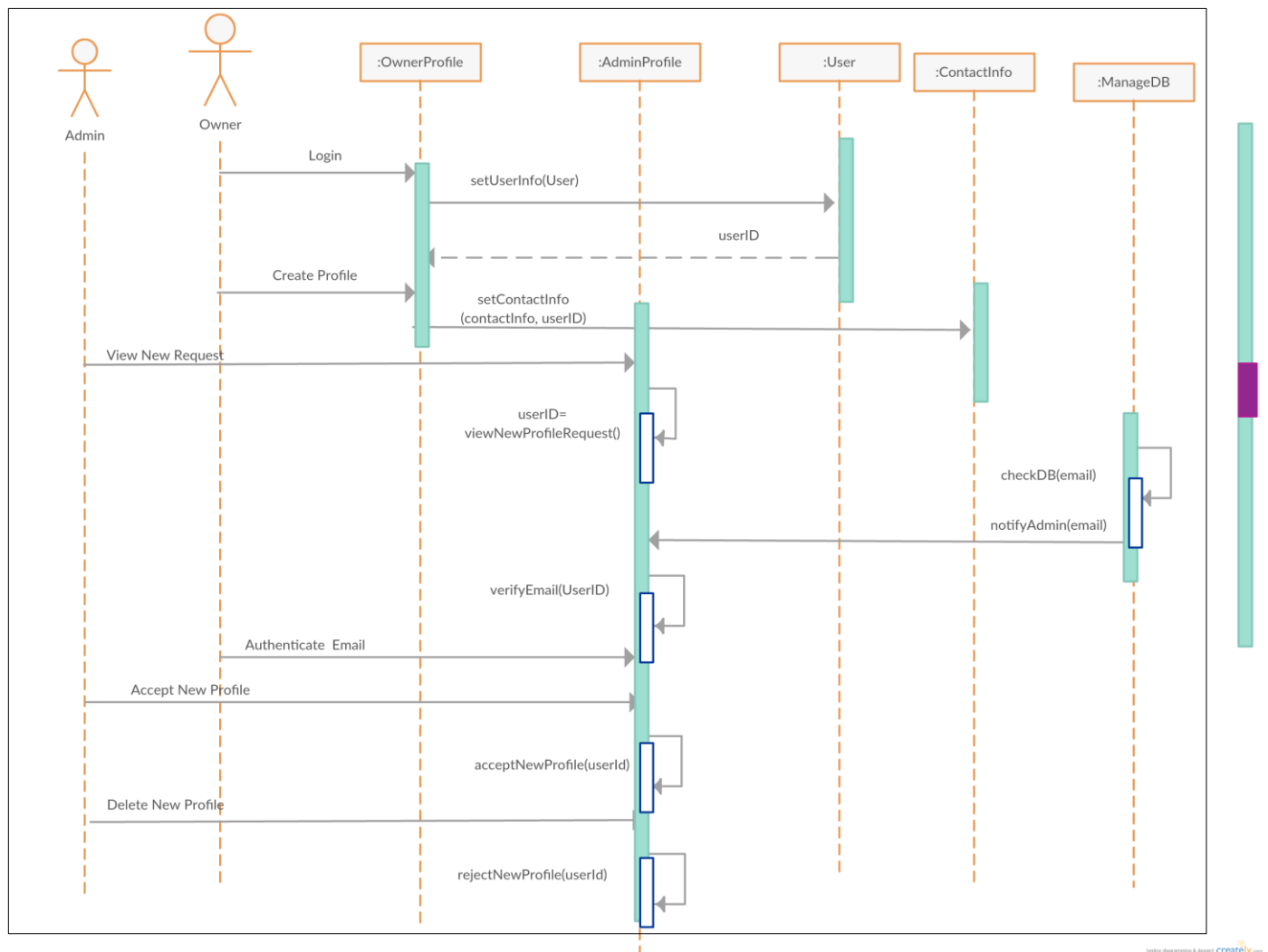


## ❖ Sequence Diagrams

### 1. Manage Request:

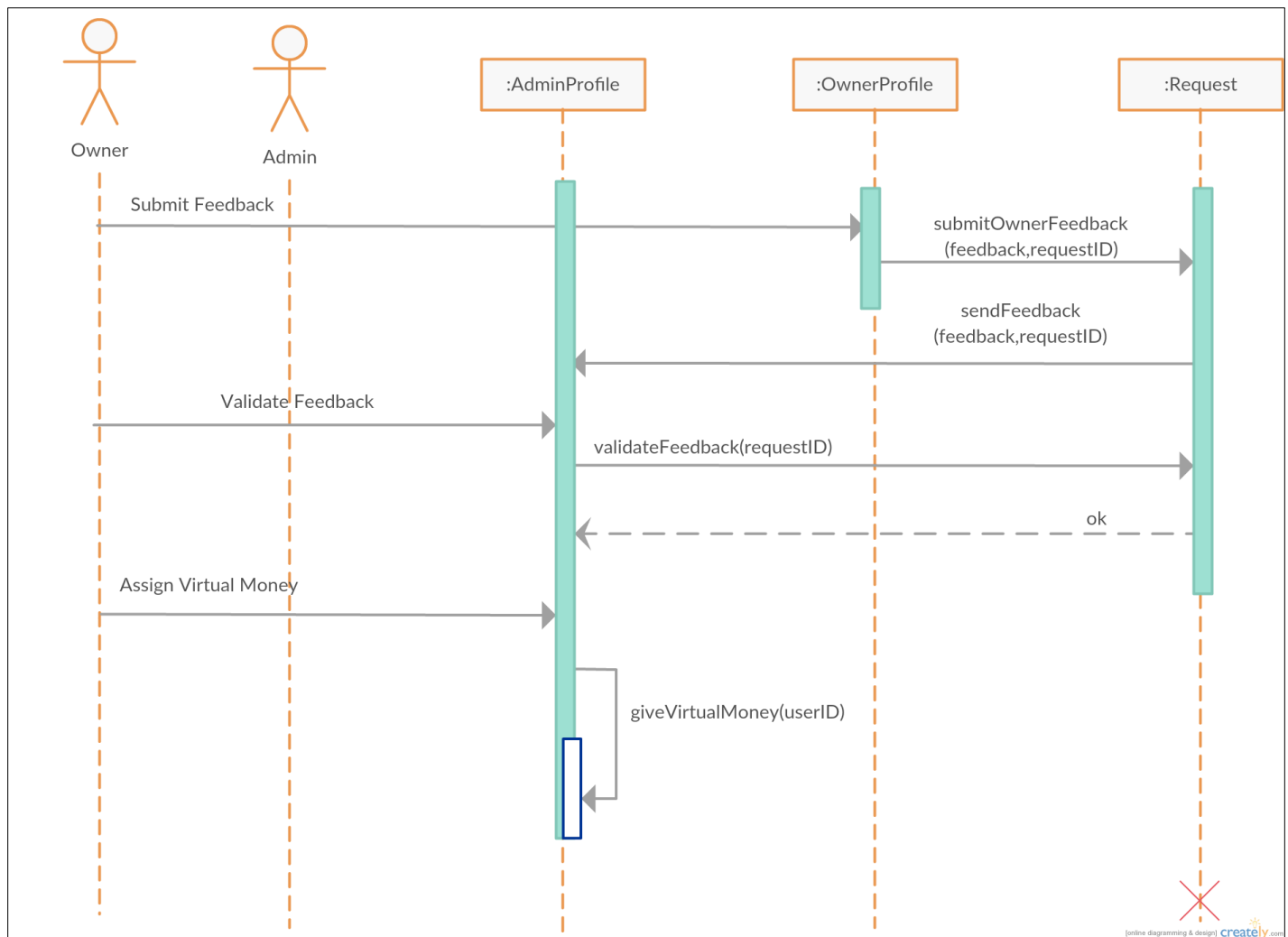
- Requirement IDs: UR-013, UR-015, UR-019, UR-020
- Use Case ID# : UC- 04
- Use Case description: Manage Request (Owner can create request for his pets and admin manages all the Requests)
- Name : Gayatri Mestry





### 3. Complete Transaction:

- Requirement IDs: UR-011, UR-018, UR-026, UR-027, UR-028
- Use Case ID# : UC- 06
- Use Case description: Complete Transaction (Owner sends feedback, admin checks feedback, if positive sends virtual money to caretaker)
- Name : Gayatri Mestry



#### 4. Maintain System:

- Requirement IDs: UR-010, UR-021, UR-022
- Use Case ID# : UC- 07
- Use Case description: Complete Transaction (Admin checks for active accounts)
- Name : Gayatri Mestry

