Instructor: Dr. Taghreed Bagies
Group project Cpit-251
Project Report
Mercy Application

Department of Information Technology
Faculty of Computing and Information Technology
King Abdulaziz University
Jeddah – Saudi Arabia

Student Name	ID
1. Manar Mutlq Altaiary	1906775
2. Jeelan Majid Alotaibi	1907030
3. Elaf Yousef Aloufi	1911265
4. Shatha Farhan Alfaifi	1808196
5. Reema Fahad Alghamdi	1905250
Section: IAR	

Table of Contents

Abstract

Mercy is a mobile application that aims to help as many animals, and people, as possible by providing the opportunity to offer pets for adoption for people who are interested in adopting instead of placing them in shelters. In addition, the application provides a feature for app users to donate money or supplies for any pet owners in need. This application is a non-profit application concerned with creating a good environment for the pet.

Introduction

An animal is in a good state of welfare if it is healthy, comfortable, well-nourished, safe, able to express innate behaviour, and if it is not suffering from unpleasant states such as pain, fear, and distress. Protecting an animal's welfare means providing for its physical and mental needs. Ensuring animal welfare is a human responsibility that includes consideration for all aspects of animal well-being. This application would make adoption easier and efficient for people.

Analysis

Problem description

We decided to build an application according to several problems that we observed:

- There are so many stray animals without home
- People may not know how to help stray animals
- People does not know where to offer their pets for adoption
- People may not have time to look for shops to adopt pets
- Pet owners may not have enough money to provide the needs for their pets such as surgery funds
- People may not know the right place to ask for money donation or supplies to help their pets
- People don't know where to donate to pet owners in need such as, donating pet supplies or money

Solution

The solution is to make a non-profit application that is a social platform. Users can create an account either a pet owner or as a customer, each provide a set of functions.

The application provides the following for customer users:

- Adopting pets from other pet owners on the application
- Donating money and supplies
- A chat service between users to communicate with each other for adoption and donation purposes such as location of pickup
- Report stray animals by location description and uploading images

The application provides the following for pet owner users:

- An opportunity for pet owners to offer their pets for adoption by filling a form of pet’s information
- Requesting money and supplies donation
- A chat service between users to communicate with each other for adoption and donation purposes such as location of pickup
- Report stray animals by location description and uploading images

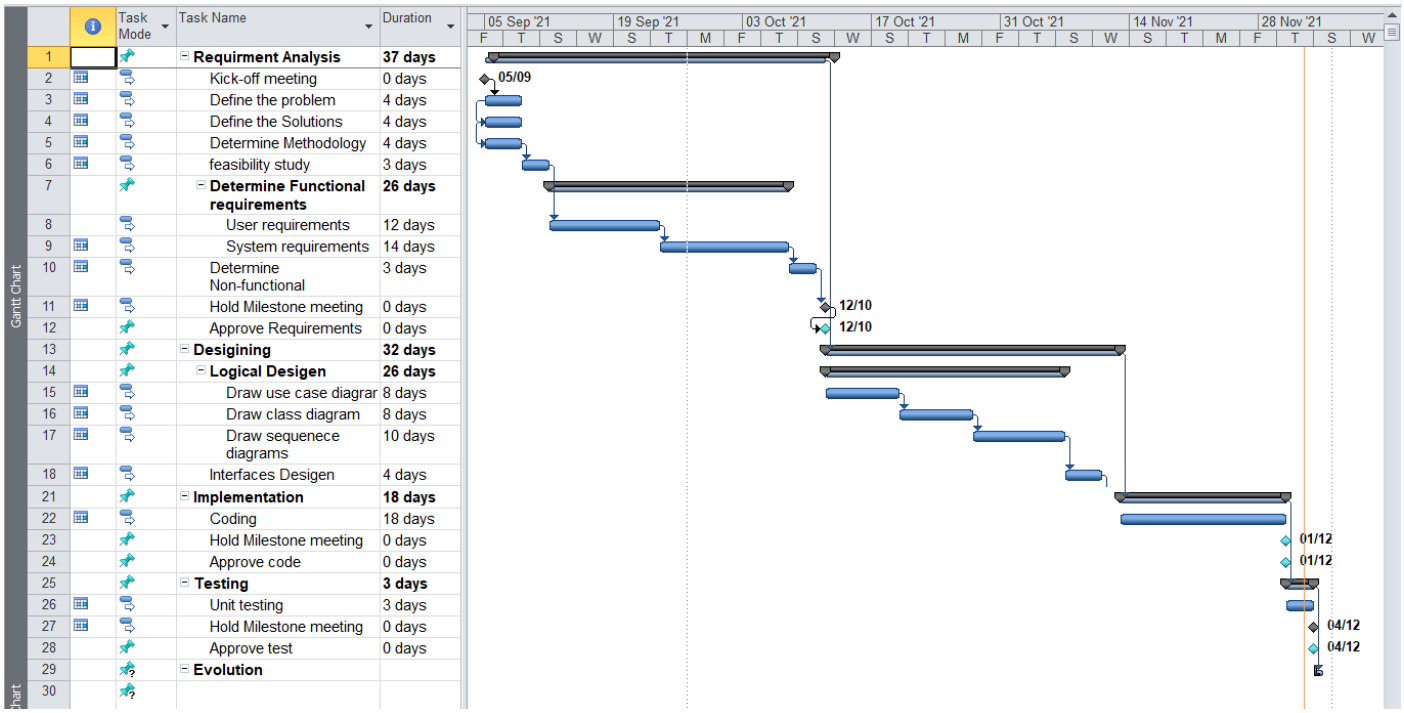
Methodology

The methodology that is used for this project is Waterfall Model, because Waterfall focuses most on a clear, defined set of steps. Its structure is simple works based on fixed dates. In addition, teams do not require consistent communication. Because the methodology is not an agile it does not require weekly meetings.
The requirement is fixed and known to all team members, so splitting work is efficient.

Feasibility study

Technical	Satisfied	The product is built to work on any mobile device. We designed the application to be a compatible on any device operating system
Economical	Not satisfied	The project requires a lot of budgets to complete
Scheduling	Satisfied	The team is working on a plan-driven model. The product of each phase is delivered based on the planned time

Process



	Task Mode	Task Name	Duration
1		**Requirment Analysis**	**37 days**
2		Kick-off meeting	0 days
3		Define the problem	4 days
4		Define the Solutions	4 days
5		Determine Methodology	4 days
6		feasibility study	3 days
7		**Determine Functional requirements**	**26 days**
8		User requirements	12 days
9		System requirements	14 days
10		Determine Non-functional	3 days
11		Hold Milestone meeting	0 days
12		Approve Requirements	0 days
13		**Desigining**	**32 days**
14		**Logical Desigen**	**26 days**
15		Draw use case diagrar	8 days
16		Draw class diagram	8 days
17		Draw sequenece diagrams	10 days
18		Interfaces Desigen	4 days
21		**Implementation**	**18 days**
22		Coding	18 days
23		Hold Milestone meeting	0 days
24		Approve code	0 days
25		**Testing**	**3 days**
26		Unit testing	3 days
27		Hold Milestone meeting	0 days
28		Approve test	0 days
29		**Evolution**	
30			

Functional requirements

User requirement

- The user shall be able to register as pet owner or a customer
- The user shall be able to login by username and password
- The user shall be able to chat with other users
- The user shall be able to report stray pets
- The customer shall be able to browse offered pet
- The customer shall be able to search offered pets by type
- The customer shall be able to adopt a pet from offered pet
- The customer shall be able to donate money to pet owners from donation request
- The customer shall be able to donate supplies to pet owners from donation request
- The pet owner shall be able to offer their pets for adoption by filling a form
- The pet owners shall be able to acquire donated supplies from the customers
- The pet owner shall be able to request supply donation by filling a form
- The pet owner shall be able to request money donation by specifying money goal and description as well as uploading images if they may
- The pet owner shall be able to confirm of reject adoption request from customers
- The pet owners shall be able to confirm donated supplies from customers

System Requirement

- The system shall ask customer input of supply information once the customer wants to donate a supply
- The system shall alert the pet owner by notification once a customer donates a supply
- The system shall send customers' donated supply form automatically to the pet owner user once the customer done submitting the form
- The system shall require customer input of card information once the customer opens the donate money form
- The system shall use the bank’s service to verify the validity of the customer’s credit card to complete the money donation to the intended pet owner
- The system shall send notification to the pet owner user once a customer has donated money to the pet owner
- The system shall display an error if the credit card balance is insufficient
- The system shall update available offered pet for adoption page, once a pet owner confirms the adoption of the pet
- The system shall display adoption form once the customer chooses a pet to adopt
- The system shall send the adoption form to the pet owner once the customer submits the form
- The system shall display pets according to the customer search by type
- The system shall send notification to the pet owner user once the customer request adoption of a pet

- The system shall ask the pet owner to confirm or reject the customer's adoption request
- The system shall open a chat between the pet owner and the customer once the pet owner accepted the pet adoption request
- The system shall send a notification once the pet owner rejects customer’s pet adoption request
- The system shall only accept unique username from users
- The system shall verify user password before login
- The system shall display offered pet on adoption page once the pet owner submits the offer adoption form
- The system shall update displayed pets once a pet owner user confirms an adoption
- The system shall display donated supply to the pet owner user once a pet owner chooses acquire supply
- The system shall open a chat between pet owner user and customer user once the pet owner accepts the donated supply
- The system shall remove the donated supply from the donated supplies list once the pet owner user declines the donated supply
- The system shall ask the pet owner to fill supply donation information request once pet owner wants to request supply donation
- The system shall display the pet owner’s supply donation request to the supply donation request page once the pet owner submit the request
- The system shall ask the pet owner to fill money donation information request once pet owner wants to request money donation

- The system shall display the pet owner’s money donation request to the money donation request page once the pet owner submit the request
- The system shall display features once the user login by username and password
- The system shall allow users to register only as pet owner user or customer user
- The system shall ask the user to fill information of location description and upload a picture of the animal once the user chooses to report a stray animal
- The system shall display the report of stray animal to report page once the user submits the form of report
- The system shall remove the displayed report once a user submits a confirm of rescue

Non-Functional requirements

- **Portability:** The application is available for download and usage on any operating system.

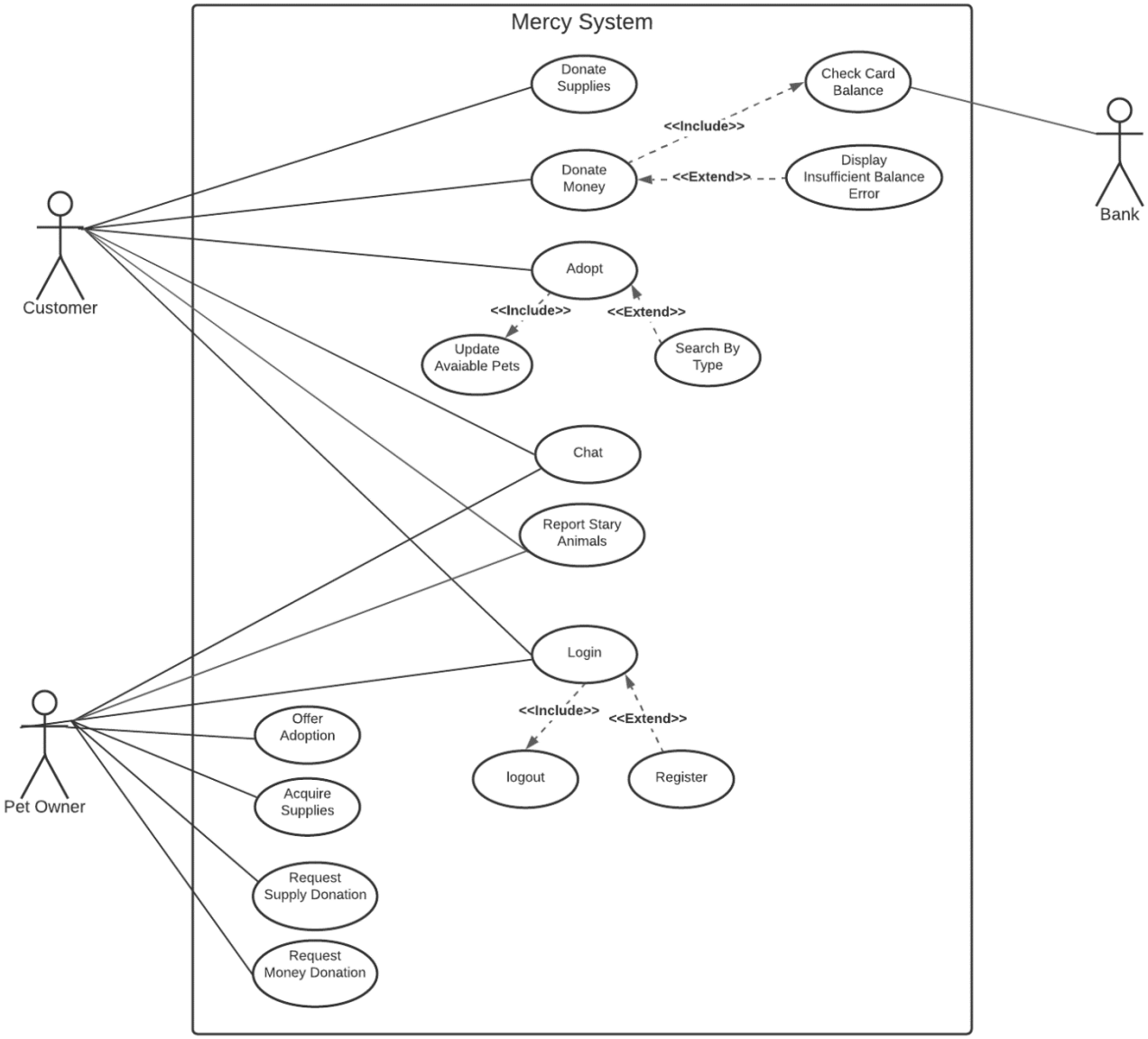
- **Usability**: The application is simple and straightforward to use, so users will not encounter any challenges.

- **Security**: User accounts' private login data is stored in the application, and users can access it after logging in.

- **Performance:** The software allows for rapid switching between features and pages.

- **Maintainability**: The application can be easily maintained, and the system will be maintained on a regular basis to ensure that it remains operational for a long period.

Use Case Diagram



Use case descriptions

Use case title: Offer Adoption
Primary actor: Pet Owner
Level: Kite(summary)
Stakeholders: Customer, Pet owner
Precondition: pet owner needs to access the app
Minimal guarantee: The Adoption offer does not get uploaded to the app
Success guarantee: Adoption offers uploaded to the app
Trigger: Pet owner enters the app and completes the adoption process for his pet
Main success scenario: **1.** Pet owner enters the app **2.** Pet owner searches for the option of adoption offers **3.** Pet owner fills out a form to offer his pet for adoption **4.** Adoption offers successfully uploaded
Extensions: **1.** Adoption form is not accepted a. Ensure the data entered is correct and complete b. Verify that all data has been filled out completely c. Check the condition for adoption **2.** Offer Adoption option is not working a. Pet owner opens the option at another time b. Pet owner quits app

Use case title: Request supply donation
Primary actor: Pet Owner
Level: Kite(summary)
Stakeholders: Customer, Pet owner
Precondition: application must connect to network; pet owner needs to access the application
Minimal guarantee: No transaction completed
Success guarantee: Request supply donation has been completed on pet owner account
Trigger: Pet owner click on request supply donation button
Main success scenario: **1.** Pet owner enters the app **2.** Pet owners click on the request supply donation button **3.** Pet owners write pet's information and the cause he need donation with the documents and bank account **4.** Request supply donation successfully uploaded
Extensions: **1.**Error in filling information a. Ensure the data entered is correct and complete b. Verify that all data has been filled out completely c. Verify bank information **2.** Request supply donation advertising did not upload a. Reload the page b. Pet owner quit the application

Use case title: Check card balance
Primary actor: Customer, Bank
Level: Kite(summary)
Stakeholders: Customer, Bank
Precondition: customer needs to access the app
Minimal guarantee: The system could not check the card balance
Success guarantee: Balance verified successfully
Trigger: Customer chooses the donate money option and the donation process
Main success scenario: **1.** Customer input card information to donate money **2.** The system fetches user's card information from the specified bank **3.** The system checks if there is enough balance on the card to donate **4.** Money has been donated successfully if enough balance is available
Extensions: **1.** Card balance check page is unavailable a. Customer tries another time b. Customer refresh the page **2.** The available balance is insufficient a. Customer tries again to make sure of the unavailability of balance b. Change donation card

Use case title: Display insufficient balance error
Primary actor: Customer, Bank
Level: Kite(summary)
Stakeholders: Customer, Bank
Precondition: Customer needs to access the app
Minimal guarantee: The donation process is incomplete
Success guarantee: The donation process has been completed
Trigger: Customer chooses the donate money option and the donation process
Main success scenario: **1.** Customer input card information to donate money **2.** The system fetches checks the availability of sufficient balance to donate **3.** The system displays a message to customer that the donation process has not been completed due to insufficient balance
Extensions: **1.** The system didn't display any feedback a. Customer tries another time b. Customer refresh the page

Use case title: Chat
Primary actor: Customer, Pet owner
Level: kite
Stakeholders: Customer, Pet owner
Precondition: Log in to account
Minimal guarantee: Unsend message
Success guarantee: Communication with each other
Trigger: Enter to chat page of the user
Main success scenario: **1.** Log in to account. **2.** Enter to user account. **3.** Enter to chat icon. **4.** Send message. **5.** reply to user message
Extensions: **1a.** unsend message 1a1. Error message “Lost connection of the internet”

Use case title: report stray pets
Primary actor: Customer, Pet owner
Level: kite
Stakeholders: Customer, Pet owner
Precondition: Log in to account
Minimal guarantee: Not loaded report to application
Success guarantee: loaded report to application
Trigger: Enter to report stray pets page
Main success scenario: **1-** Log in to account. **2-** Enter to to report stray pet’s page. **3-** Fill report information. **4-** Uploaded the report
Extensions **1a.** Not loaded report to application 1a1. Error message “Lost connection of the internet”

Use case title: Log out
Primary actor: Customer, Pet owner
Level: kite
Stakeholders: Customer, Pet owner
Precondition: Log in to account
Minimal guarantee: The account is not logged out
Success guarantee: Log out of the account
Trigger: Enter account seating
Main success scenario: **1.** Log in to account **2.** Enter to account seating **3.** Log out
Extensions: **1a.** not logged out 1a1. Error message “Lost connection of the internet”

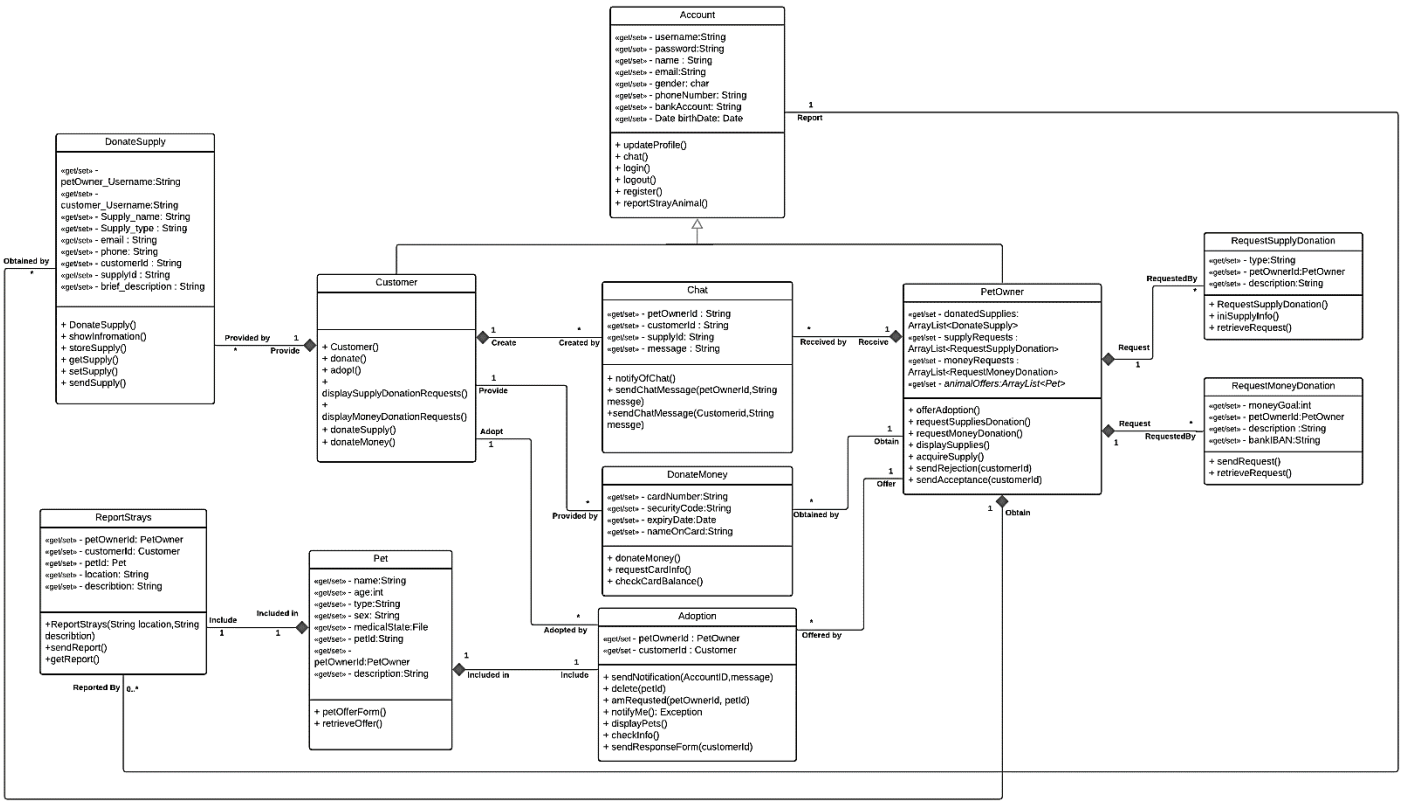
Use Case Title: Donate Money
Primary Actor: Customer
Level: kite(summary)
Stakeholders: Pet Owner, Customer
Precondition: Customer access the application
Minimal Guarantee: Rollback of any incomplete transaction
Success Guarantee: Money donation completed successfully
Trigger: Customer click on the donate money option
Main Success Scenario: **1.** Customer chooses money donation type either (Money or Supplies) **2.** Customer chooses the intended Pet Owner to donate **3.** Customer enters the desired amount of donation money **4.** Customer fills the credit card information and **5.** Customer clicks on the donate button **6.** The system requests conformation of the transaction **7.** The transaction completed successfully
Extensions: **a.** Donation form function is not available a1. Customer quits site a2. Customer reloads the donation page **b.** Transaction Rolled back b1. Customer checks if there is enough money available in the card b2. Customer starts again b3. Customer quits application **c.** Confirmation of donation is interrupted c1. Customer seeks other means of confirmation c2. Customer quits application **d.** The donation is rejected d1. User re-enter the information correctly

Use Case Title: Update available pets
Primary Actor: pet owner
Level: kite(summary).
Stakeholders: customer, pet owner
Precondition: pet owner confirms the receipt of the pet
Minimal Guarantee: nothing happened
Success Guarantee: system removes the pet successfully from the database
Trigger: pet owner clicks on confirm receipt
Main Success Scenario: **1.** pet owner logs in successfully **2.** pet owner clicks on confirm receipt **3.** system deletes the pet from database
Extensions: **a.** Internet connection is cut off while updating is processed 1a. system asks the pet owner again to confirm receipt

Use Case Title: Search by type
Primary Actor: Customer
Level: kite(summary).
Stakeholders: customer, pet owner
Precondition: being in adoption page
Minimal Guarantee: display empty adoption page
Success Guarantee: available animals are displayed in order of the chosen type correctly
Trigger: customer click on filter
Main Success Scenario: **1.** Customer logs in successfully **2.** Customer click adopt button **3.** System displays list of available animals **4.** Customer click on filter button **5.** Customer chooses one of the present types
Extensions: **a.** no animals are available from this type 1a. User chooses another type 2a. System recommends other types 3a. Customer checks later

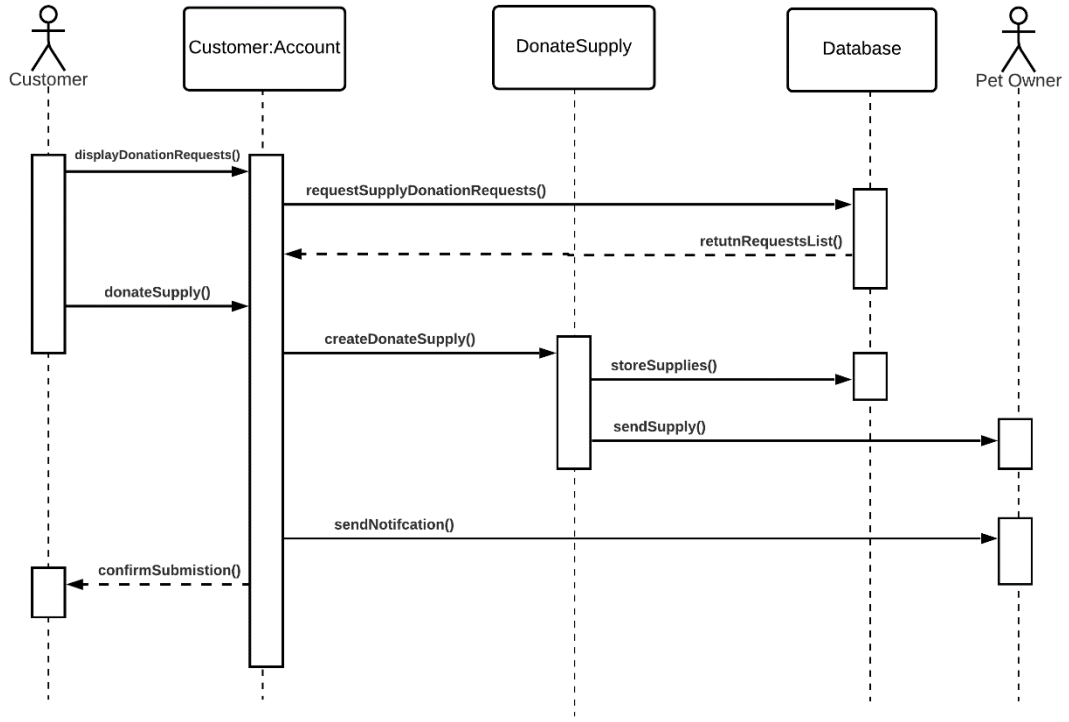
Use Case Title: Adopt
Primary Actor: customer, pet owner
Level: blue(see-level).
Stakeholders: customer, pet owner
Precondition: successfully logged in
Minimal Guarantee: display not available message
Success Guarantee: the adopter filling required information successfully
Trigger: user click the adopt button
Main Success Scenario: **1.** Customer logs in successfully **2.** Customer click adopt button **3.** System displays list of available animals **4.** Customer either choose an animal or click quit **5.** Customer fills the required adoption information **6.** System sends notification to the pet owner **7.** System asks the pet owner to confirm or reject the adoption request **8.** if accept → chat page will be opened between them if reject → System just sends excuse message to the customer
Extensions: **a.** Customer reverses the decision to adopt 1a.pet owner reoffer the adaptation **b.** More than adoption request at the same time 1b.pet owner have the freedom to choose who will owns the pet

Class Diagram

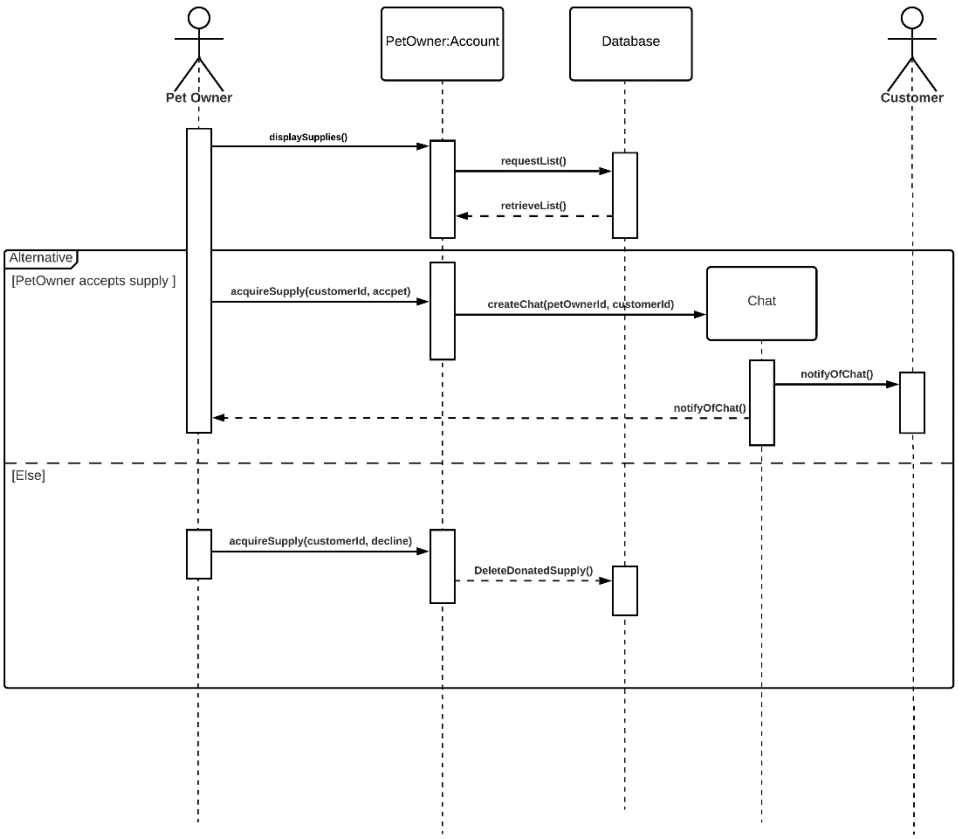


Sequence Diagram

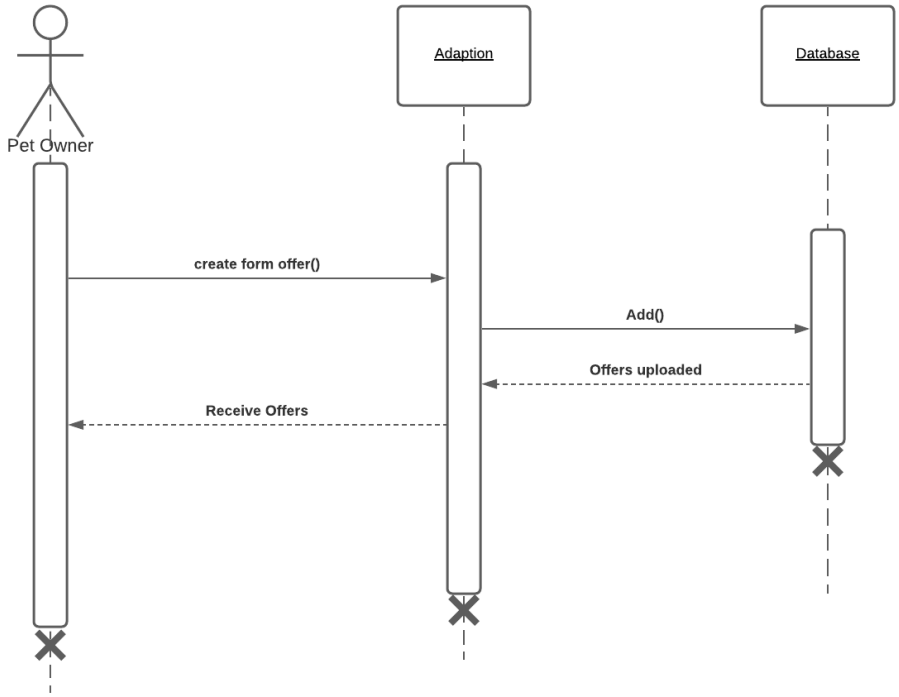
Use case: Donate Supply



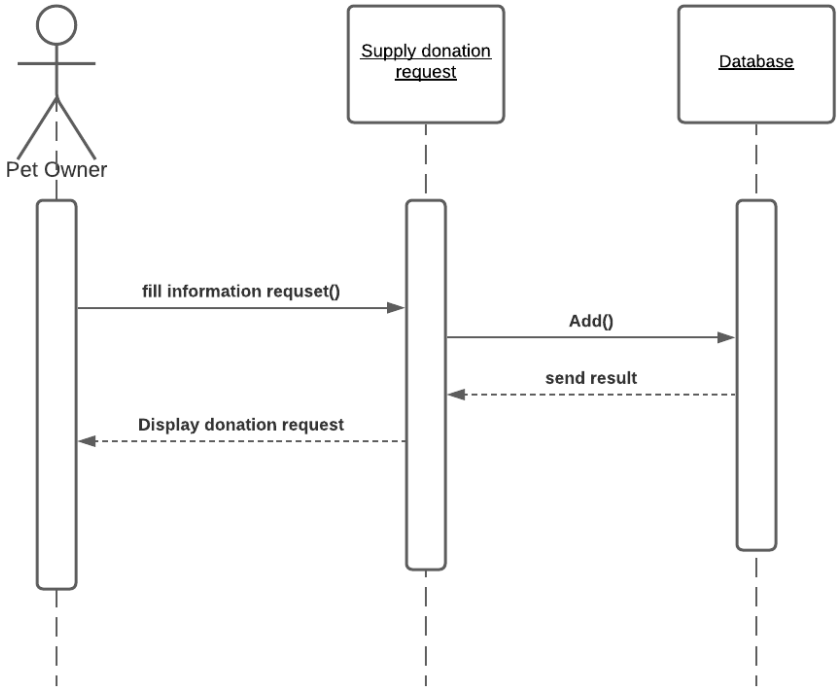
Use case: Acquire Supply



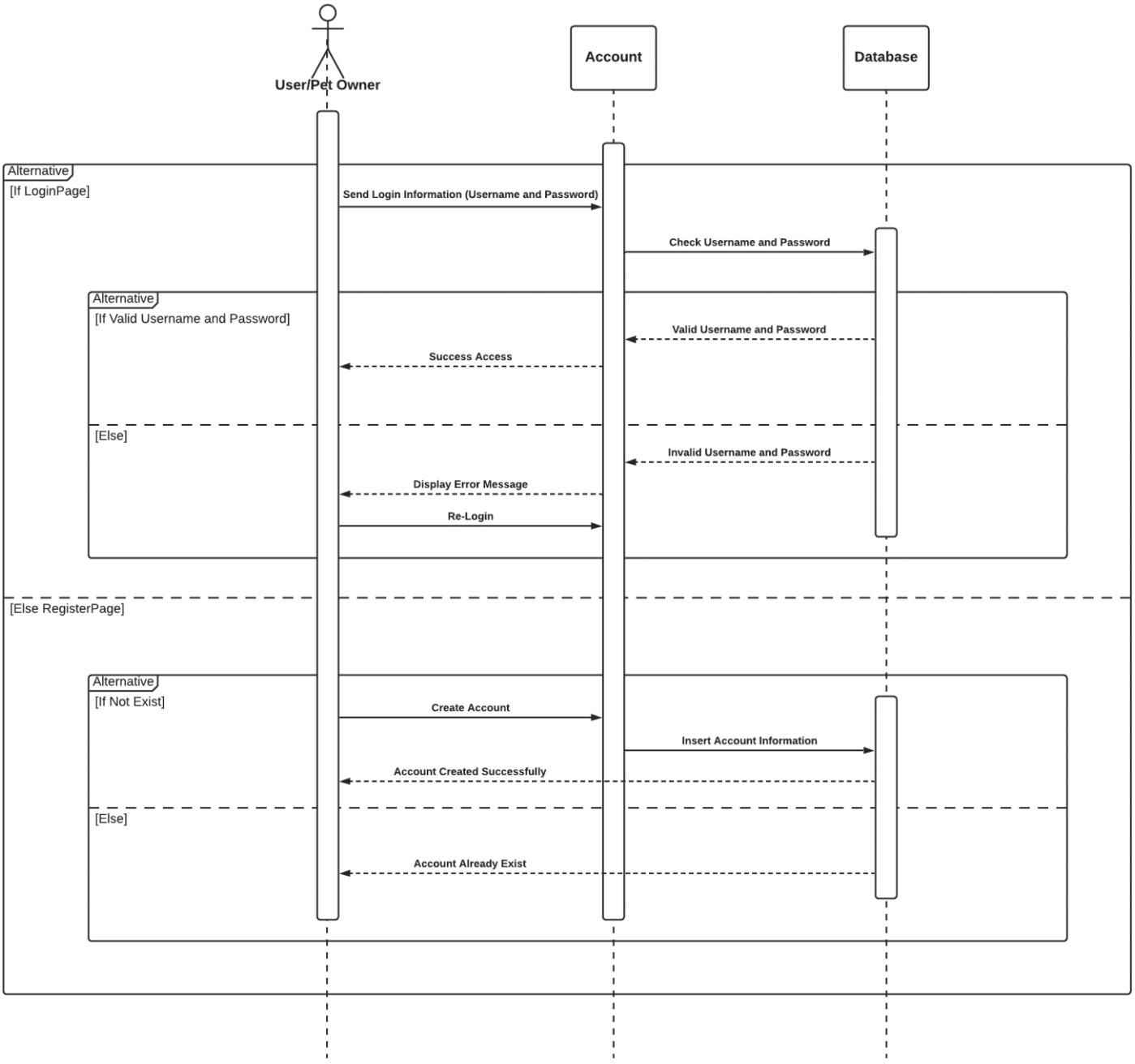
Use case: Offer adoption



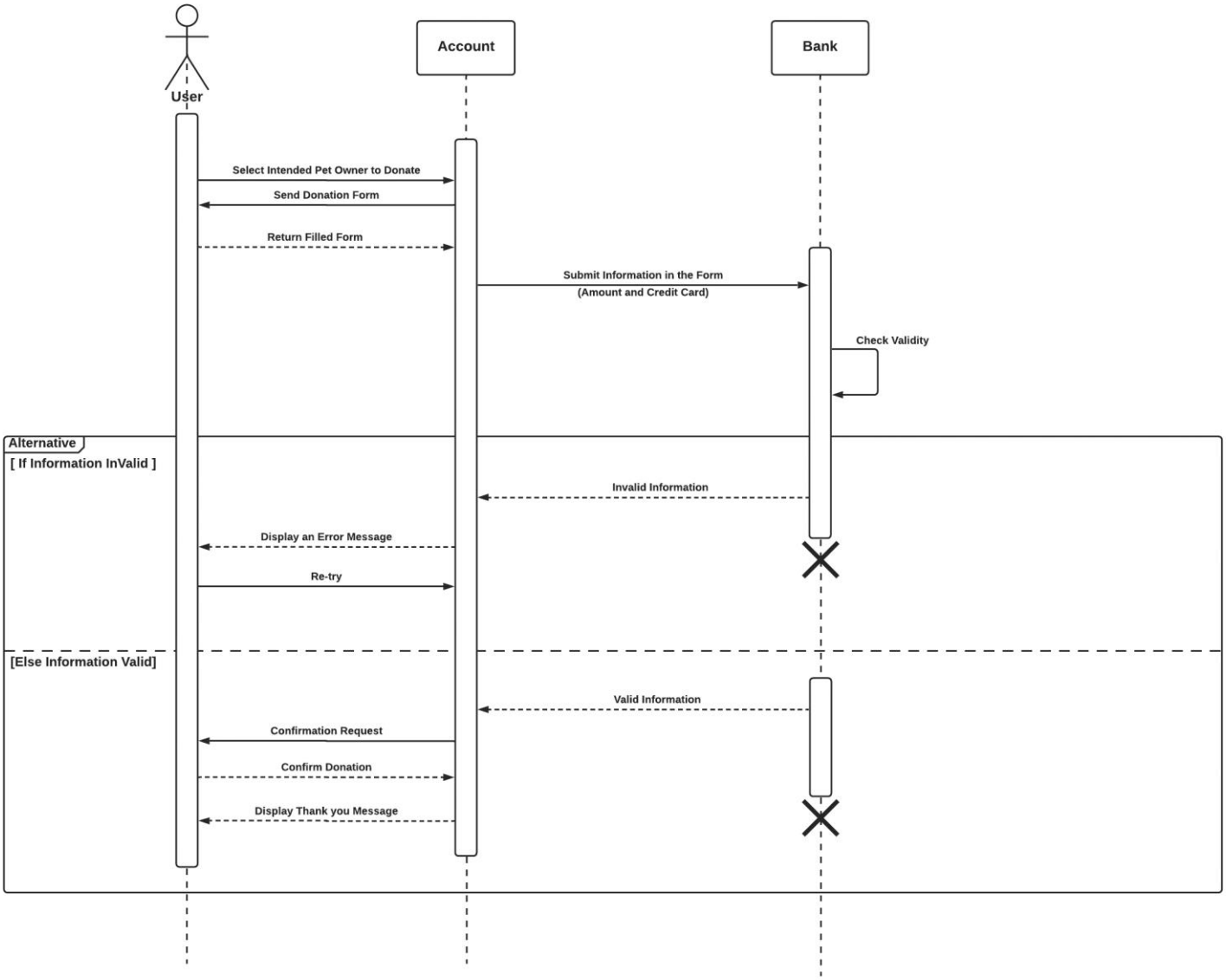
Use case: Request Supply Donation



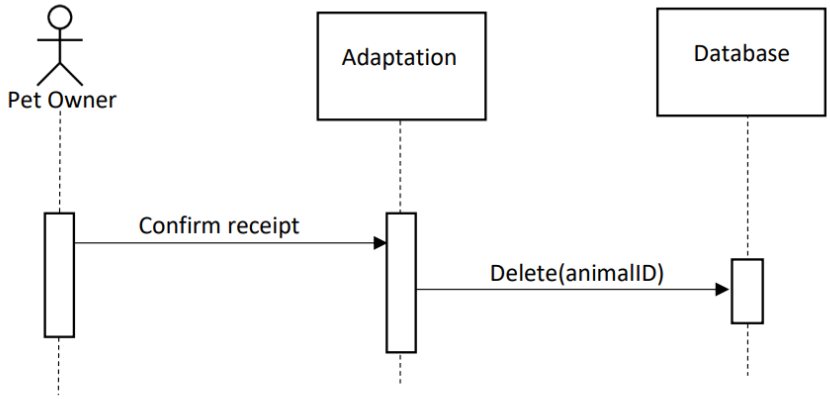
Use case: Log in



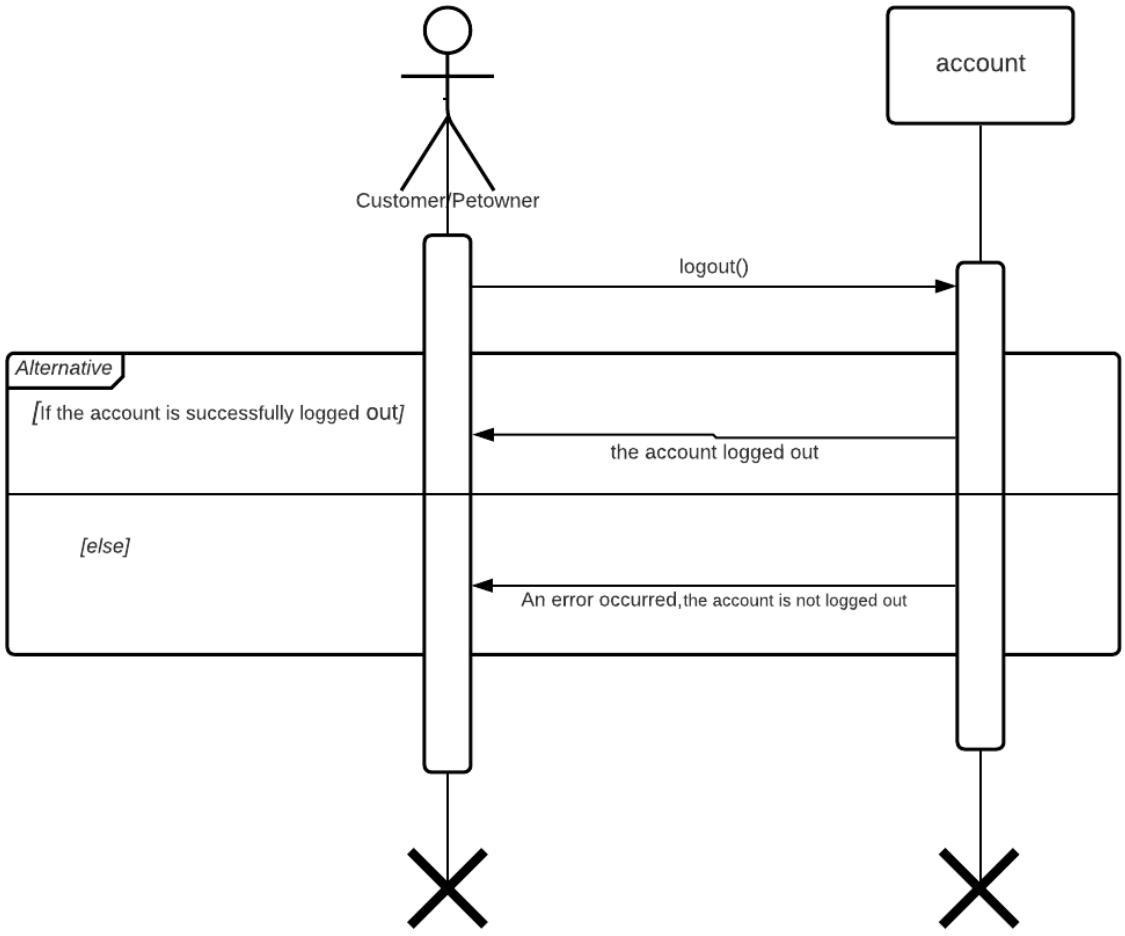
Use case: Donate Money



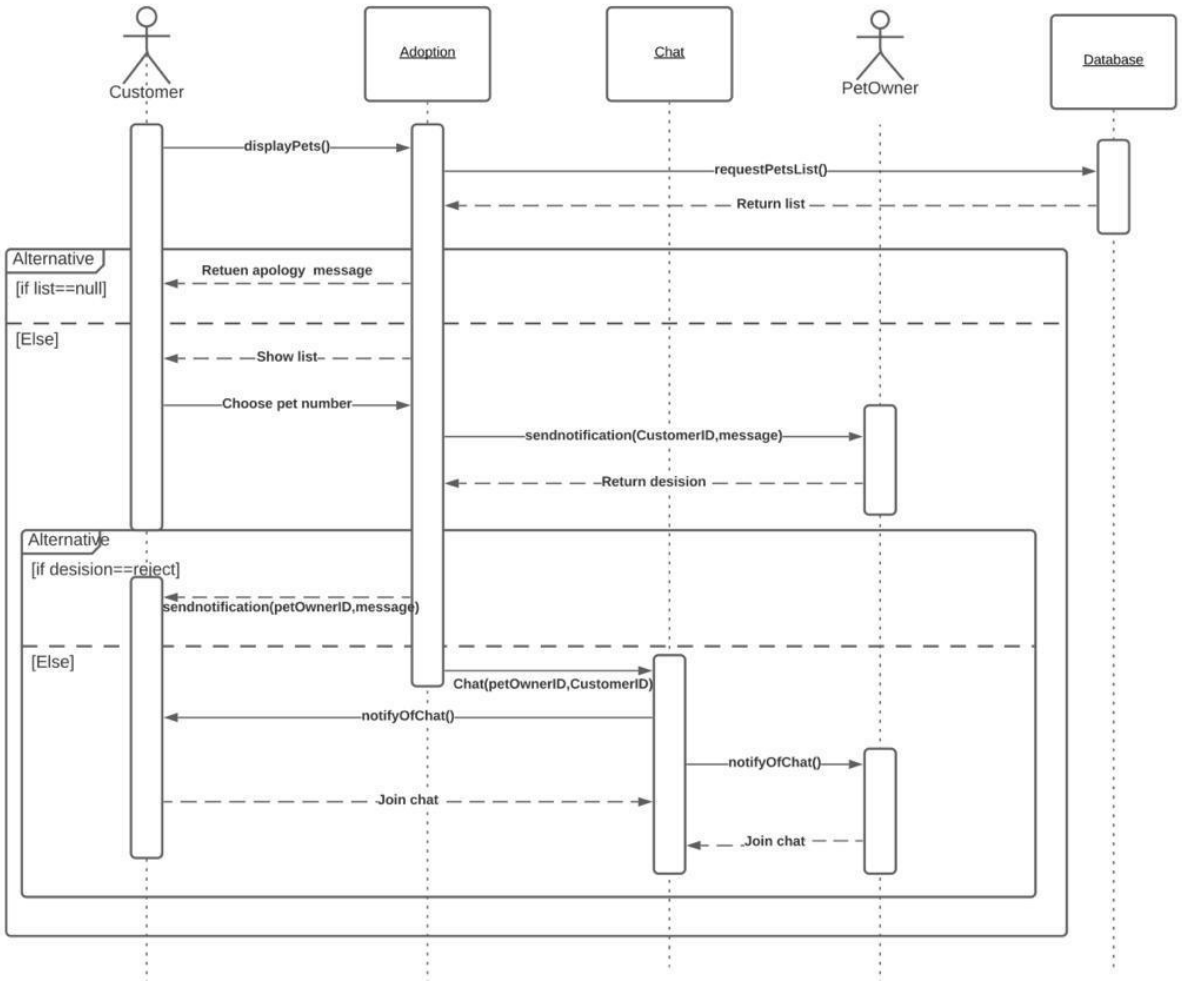
Use case: Update by type



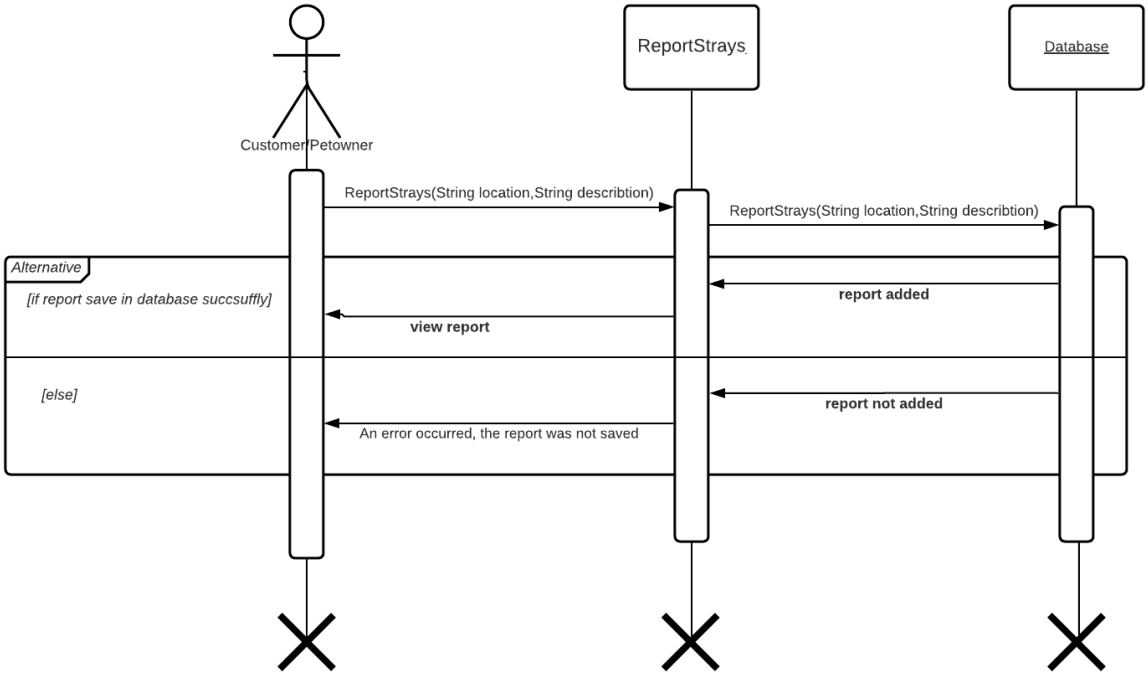
Use case: Log out



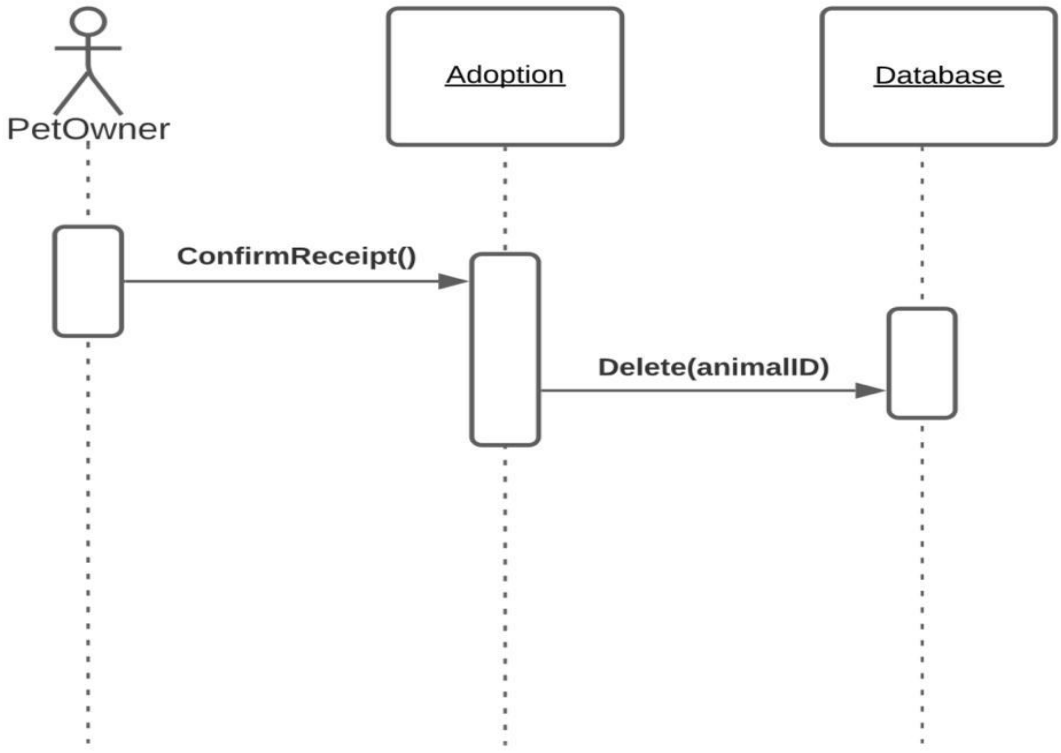
Use case: Adopt



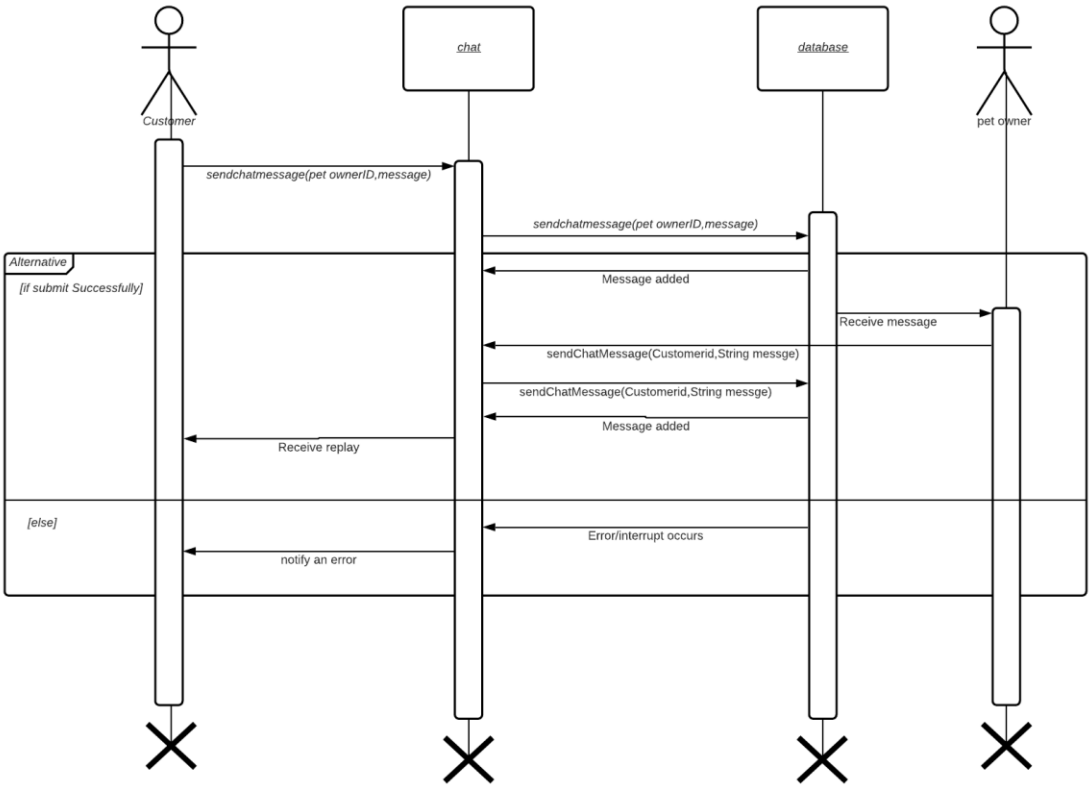
Use case: Report Stray



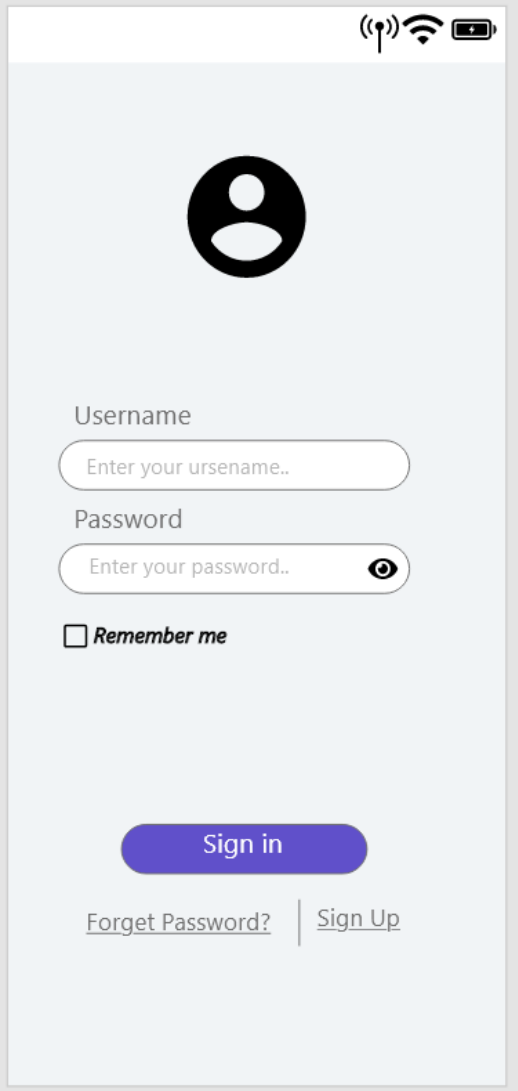
Use case: Update Availability

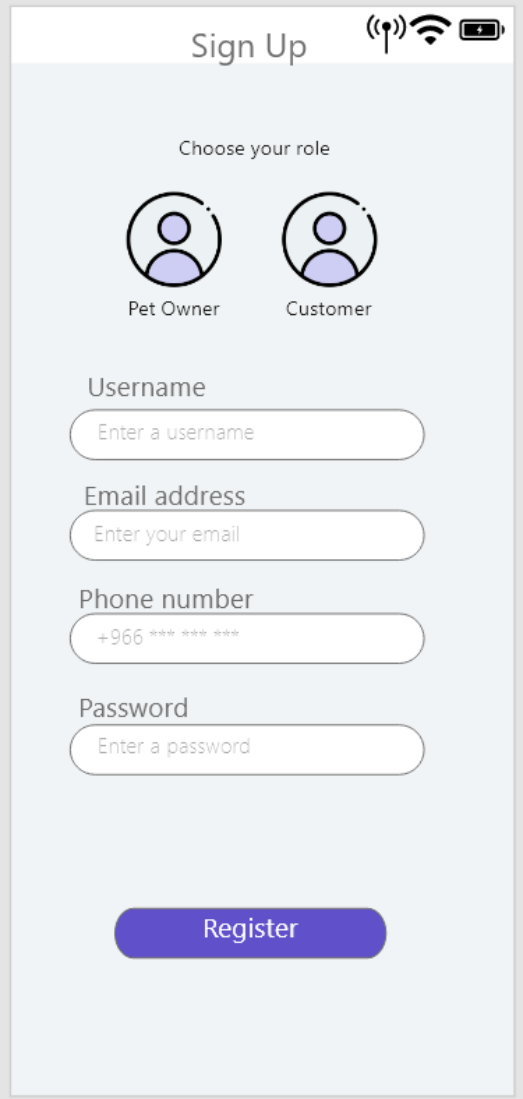


Use case: Chat

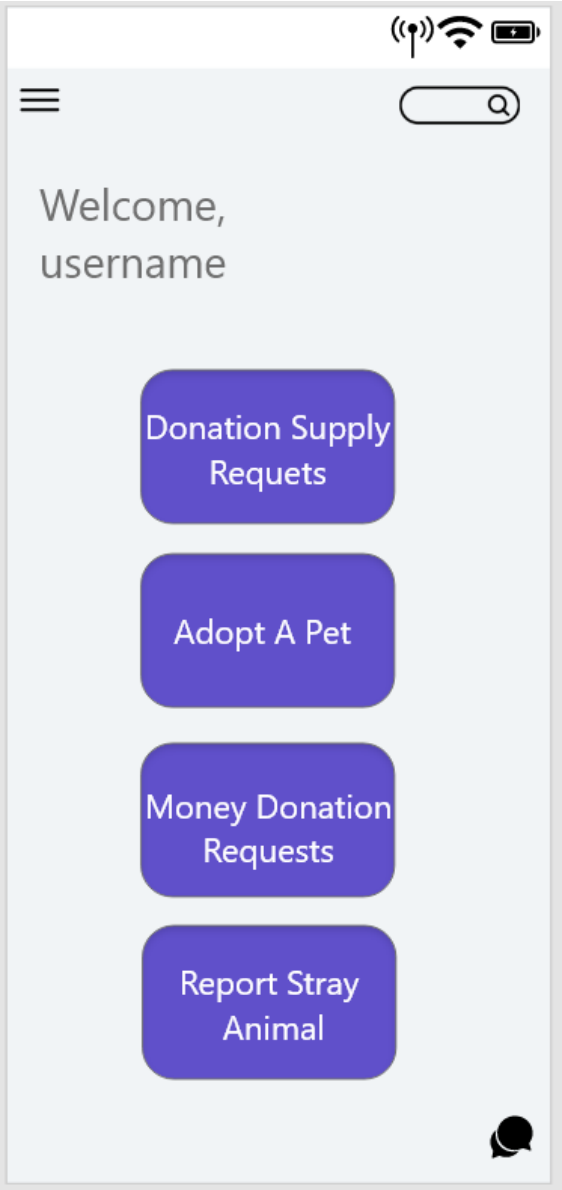


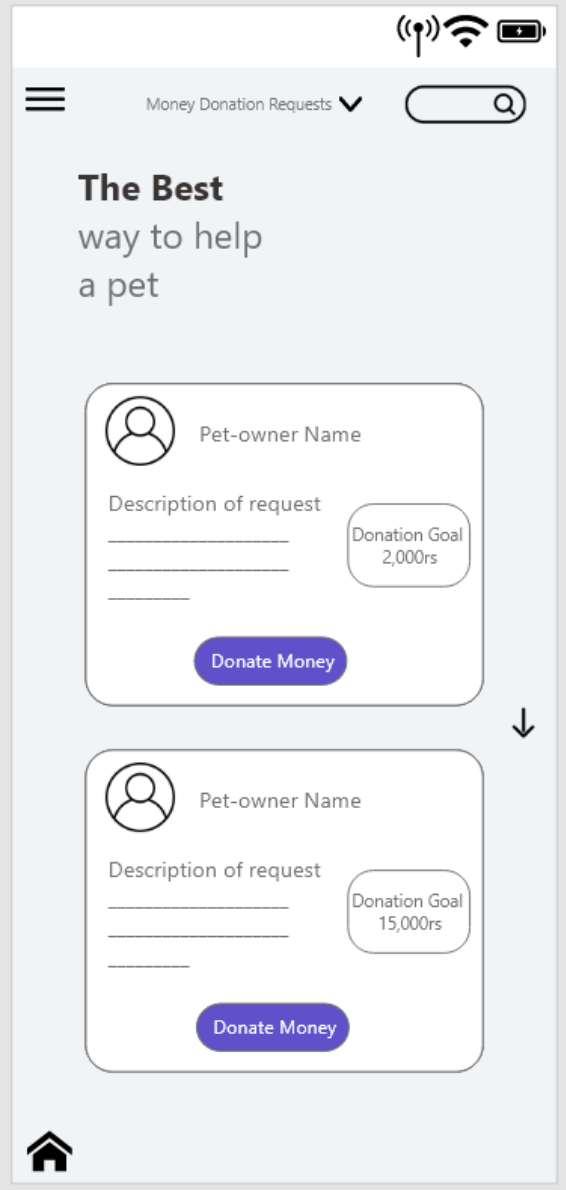
Interfaces Design

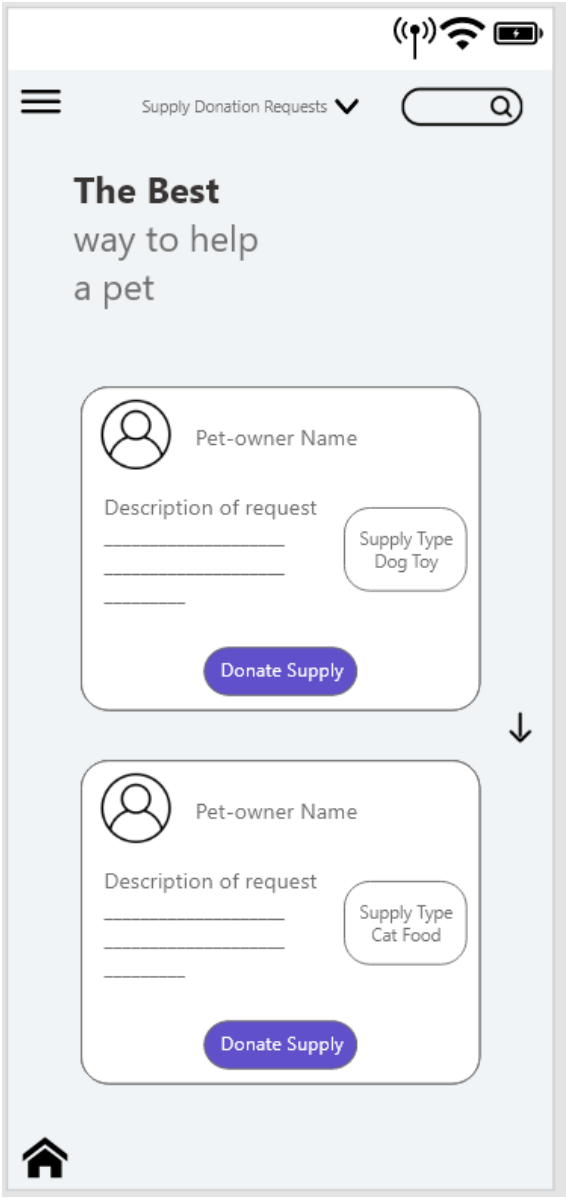


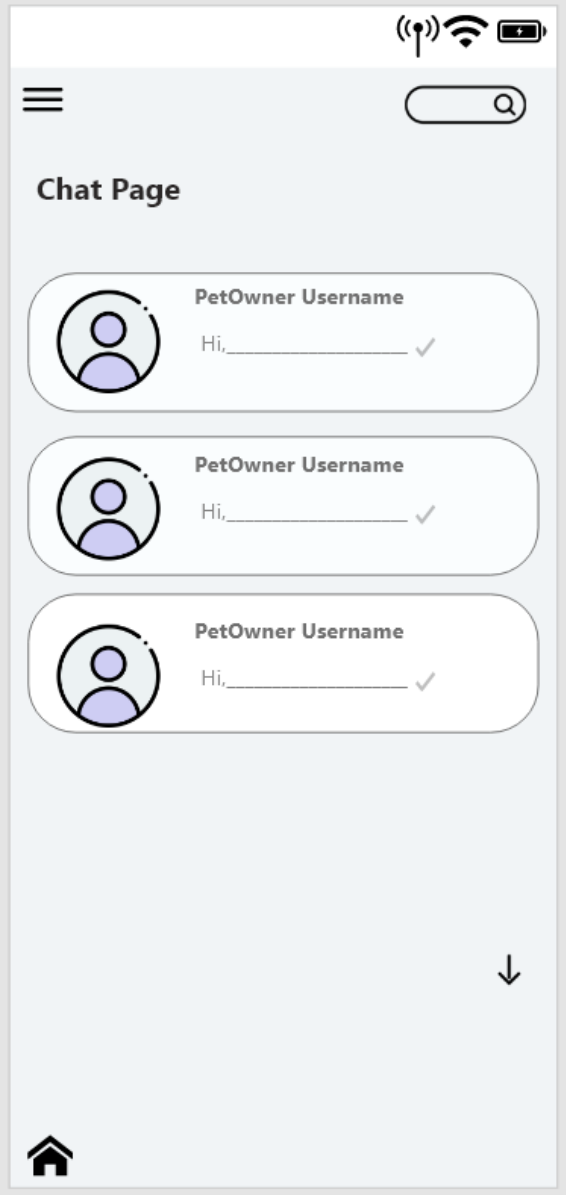


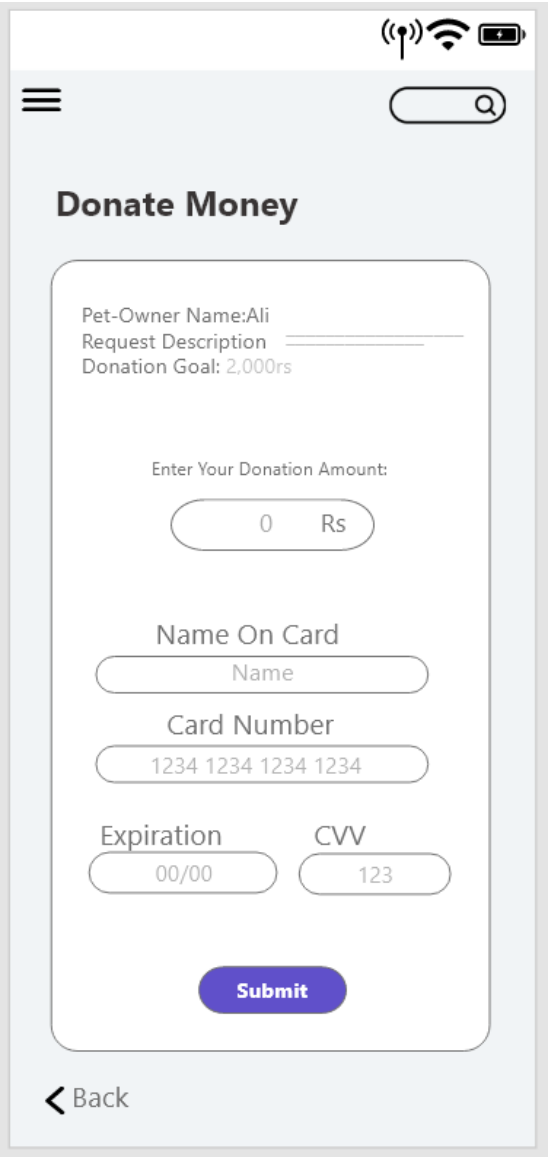
Customer Interfaces

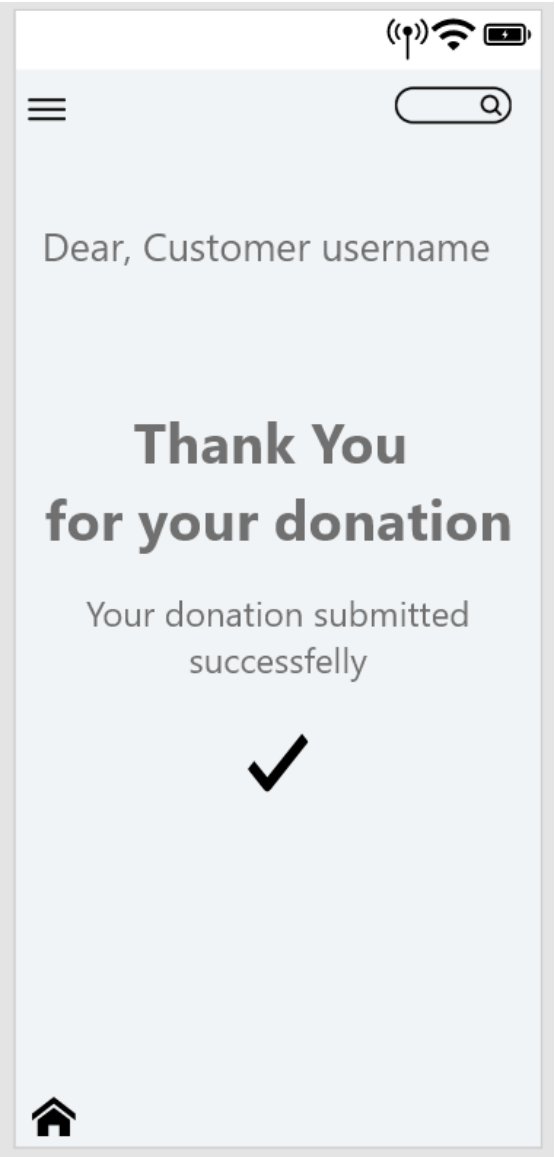


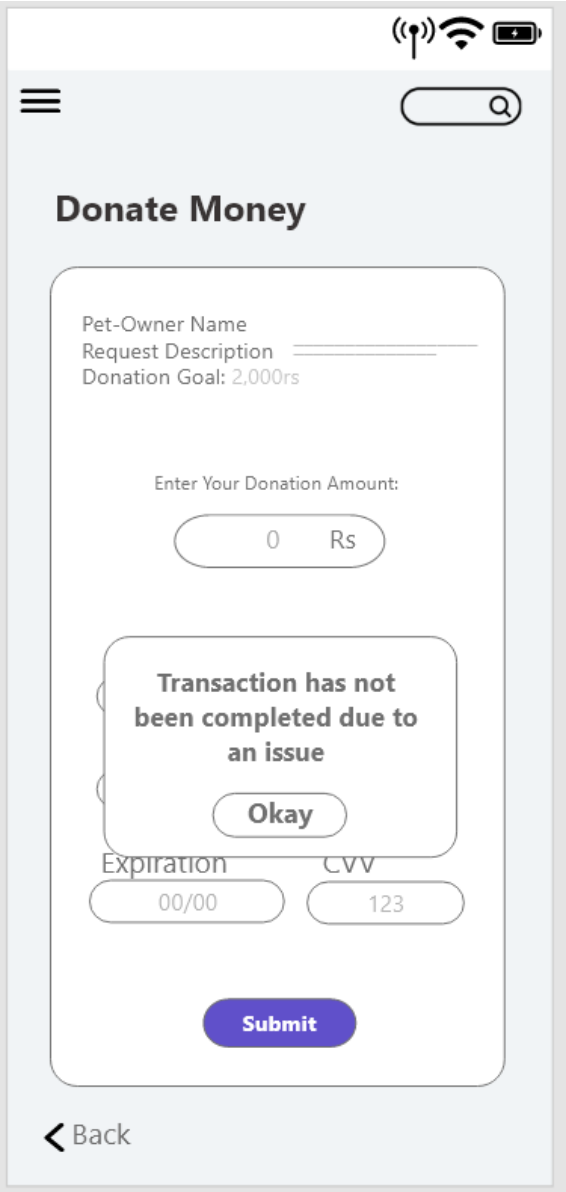


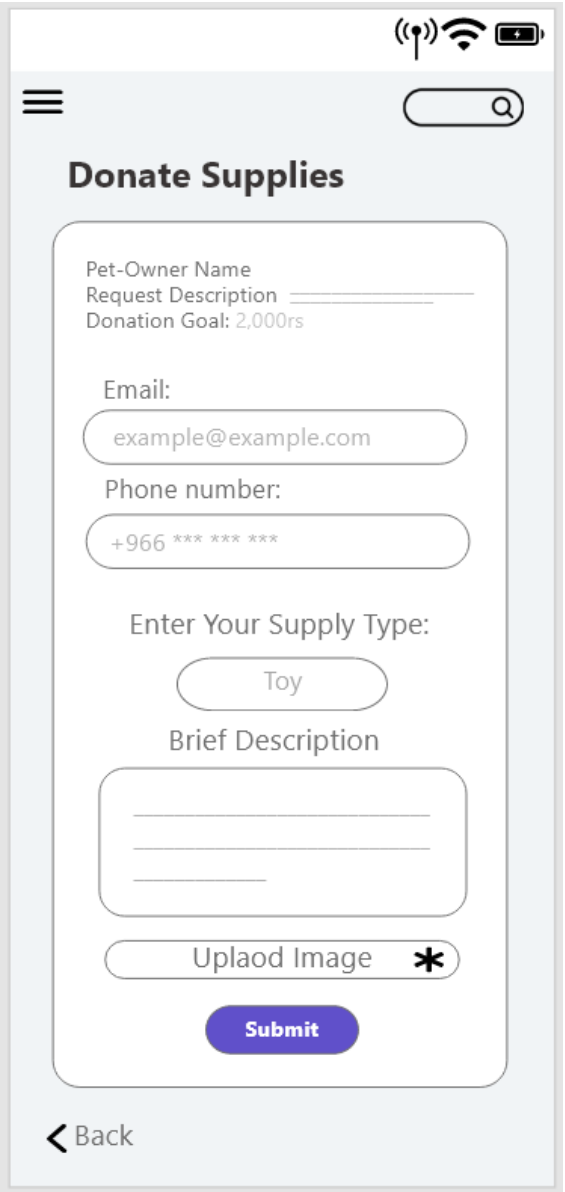

Supply Donation Requests
The Best
way to help
a pet
Pet-owner Name
Description of request
Supply Type
Dog Toy
Donate Supply
Pet-owner Name
Description of request
Supply Type
Cat Food
Donate Supply

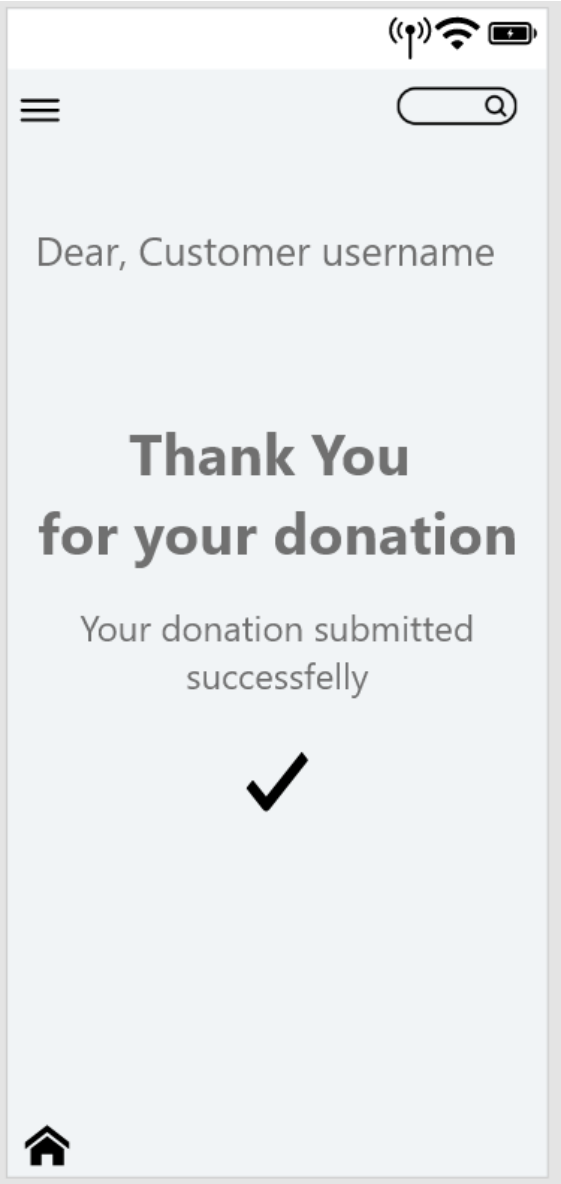

Chat Page
PetOwner Username
Hi,
PetOwner Username
Hi,
PetOwner Username
Hi,

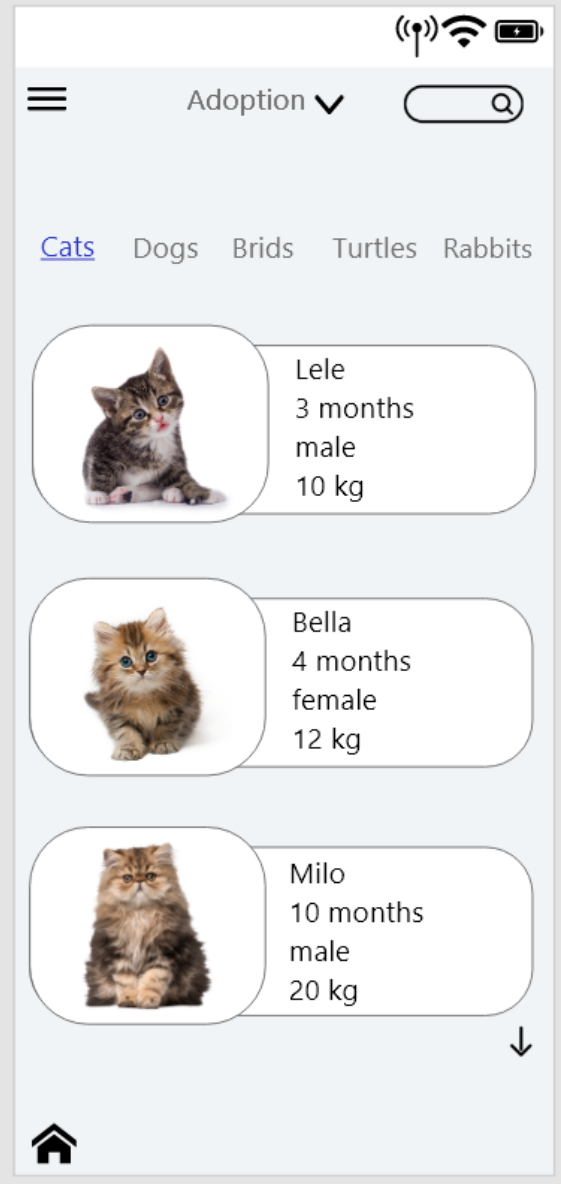

Donate Money
Pet-Owner Name:Ali
Request Description
Donation Goal: 2,000rs
Enter Your Donation Amount:
0 Rs
Name On Card
Name
Card Number
1234 1234 1234 1234
Expiration
00/00
CVV
123
Submit
Back

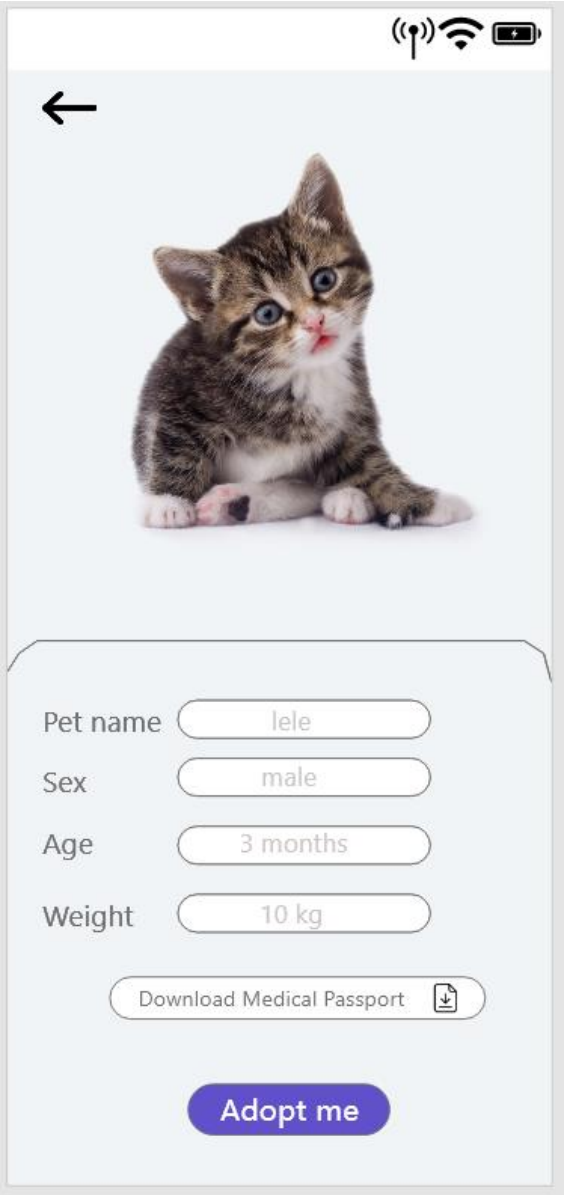

Dear, Customer username
Thank You
for your donation
Your donation submitted
successfelly

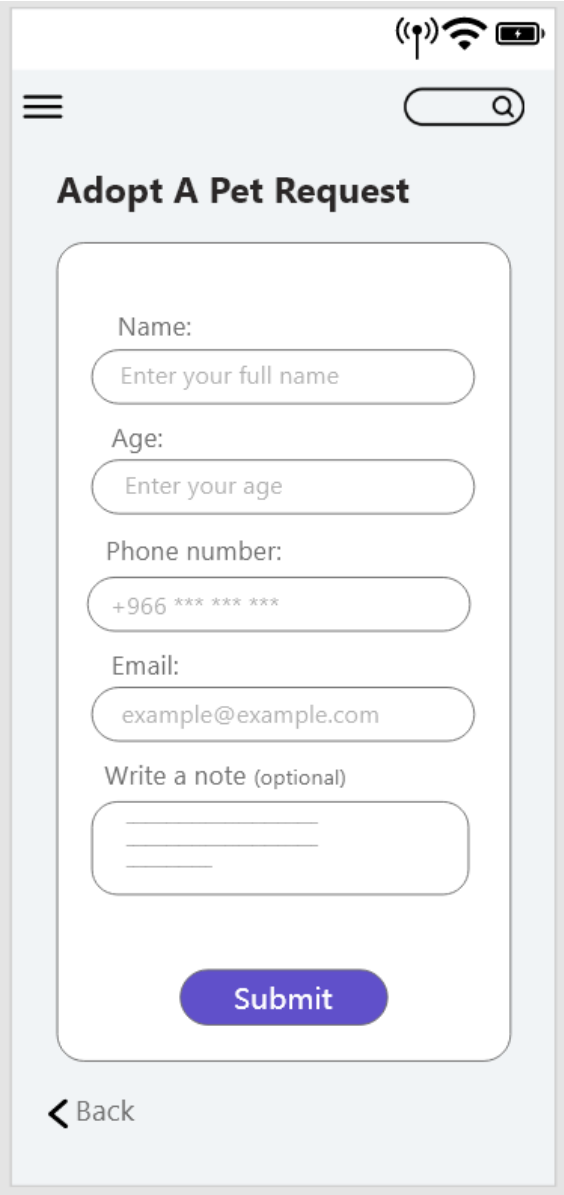

Donate Money
Pet-Owner Name
Request Description
Donation Goal: 2,000rs
Enter Your Donation Amount:
0 Rs
Transaction has not been completed due to an issue
Okay
Expiration
CVV
00/00
123
Submit
Back

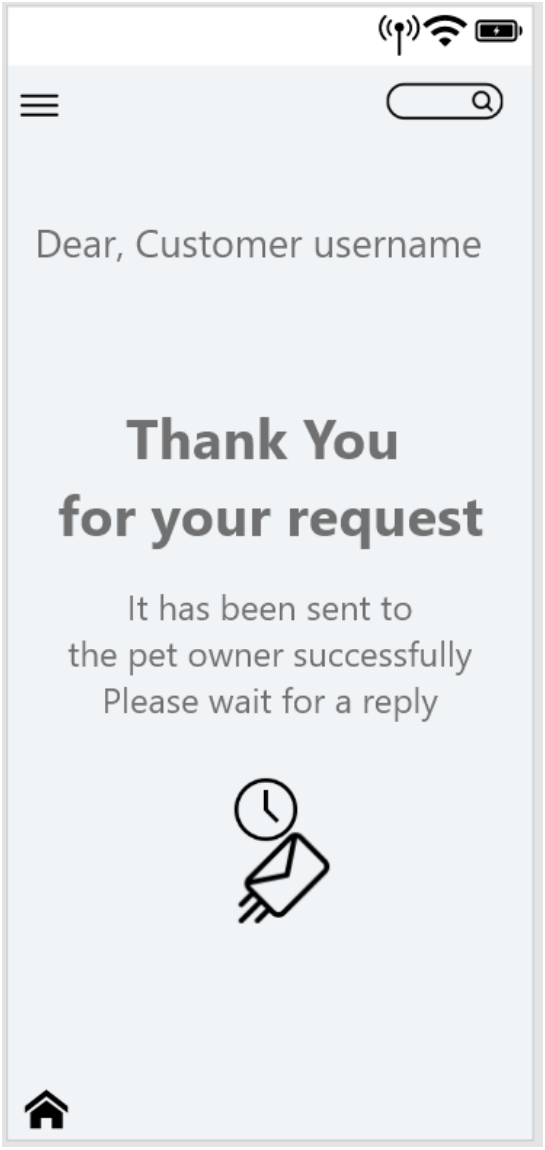

Donate Supplies
Pet-Owner Name
Request Description
Donation Goal: 2,000rs
Email:
example@example.com
Phone number:
+966 *** *** ***
Enter Your Supply Type:
Toy
Brief Description
Uplaod Image *
Submit
Back


Dear, Customer username
Thank You
for your donation
Your donation submitted
successfelly

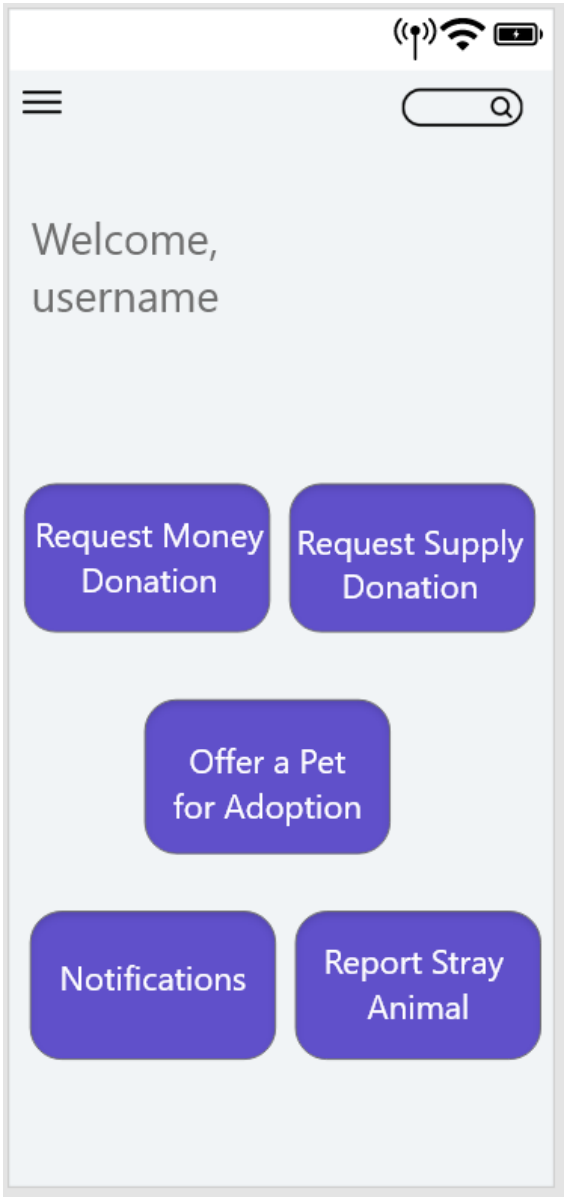

Adoption
Cats
Dogs
Brids
Turtles
Rabbits
Lele
3 months
male
10 kg
Bella
4 months
female
12 kg
Milo
10 months
male
20 kg

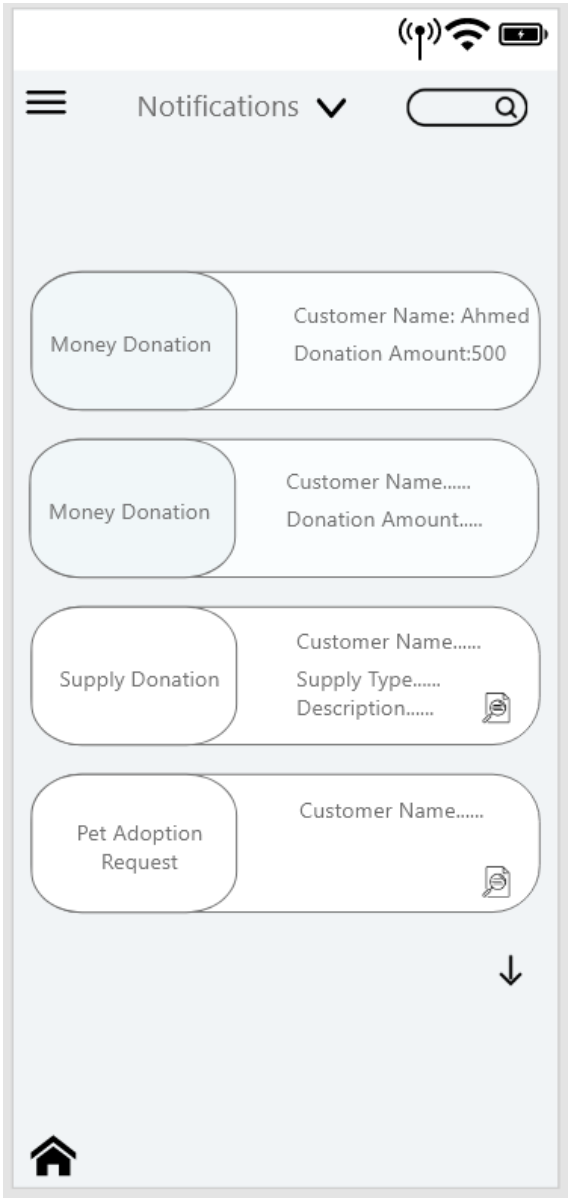

Pet name
lele
Sex
male
Age
3 months
Weight
10 kg
Download Medical Passport
Adopt me

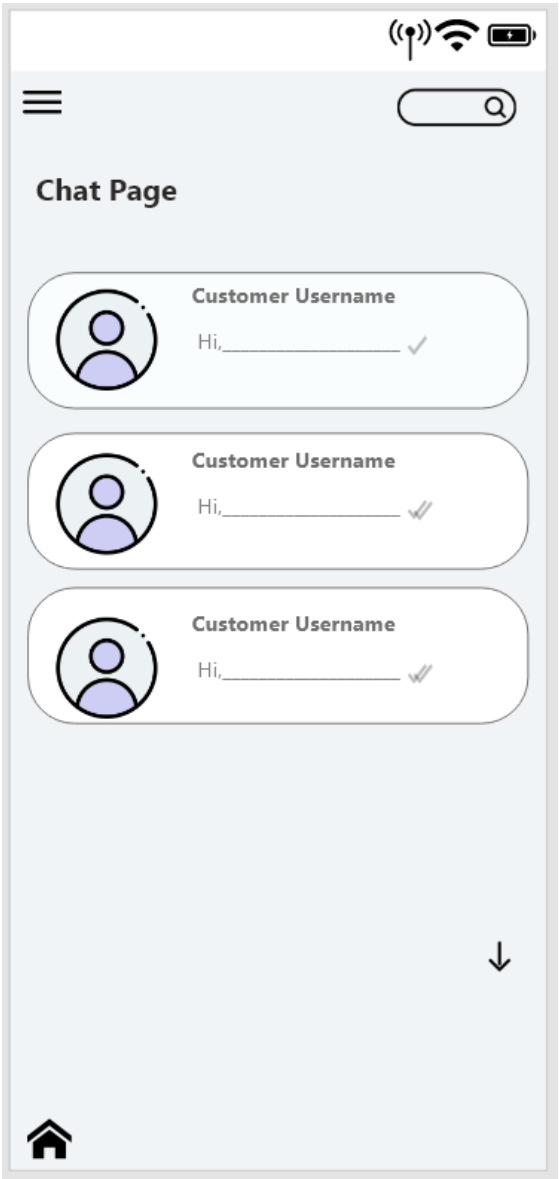

Adopt A Pet Request
Name:
Enter your full name
Age:
Enter your age
Phone number:
+966 *** *** ***
Email:
example@example.com
Write a note (optional)
Submit
Back

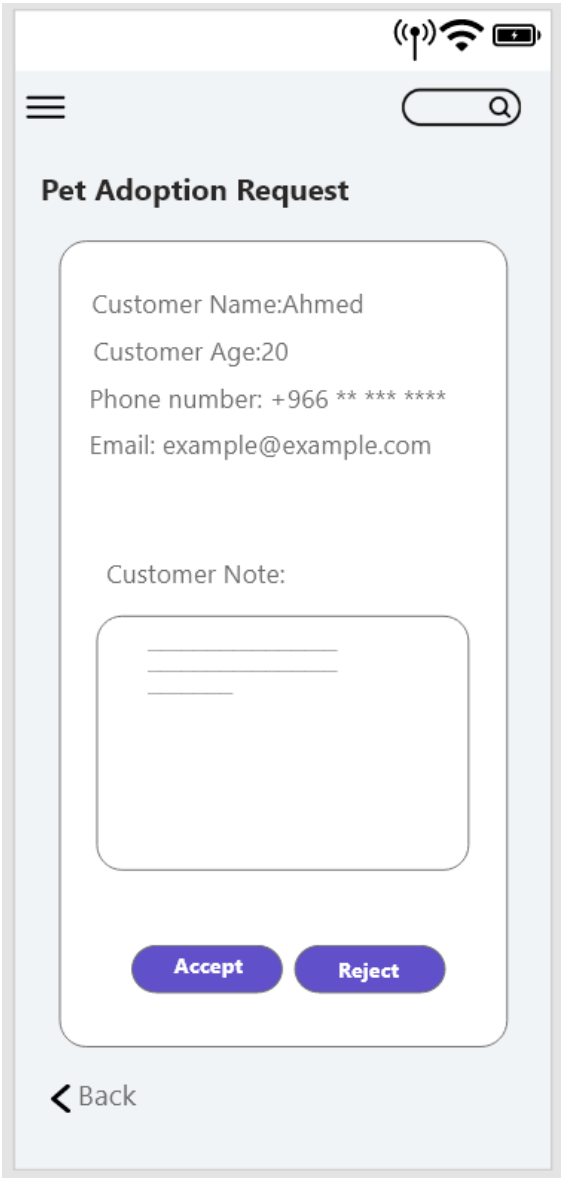

Dear, Customer username
Thank You
for your request
It has been sent to
the pet owner successfully
Please wait for a reply

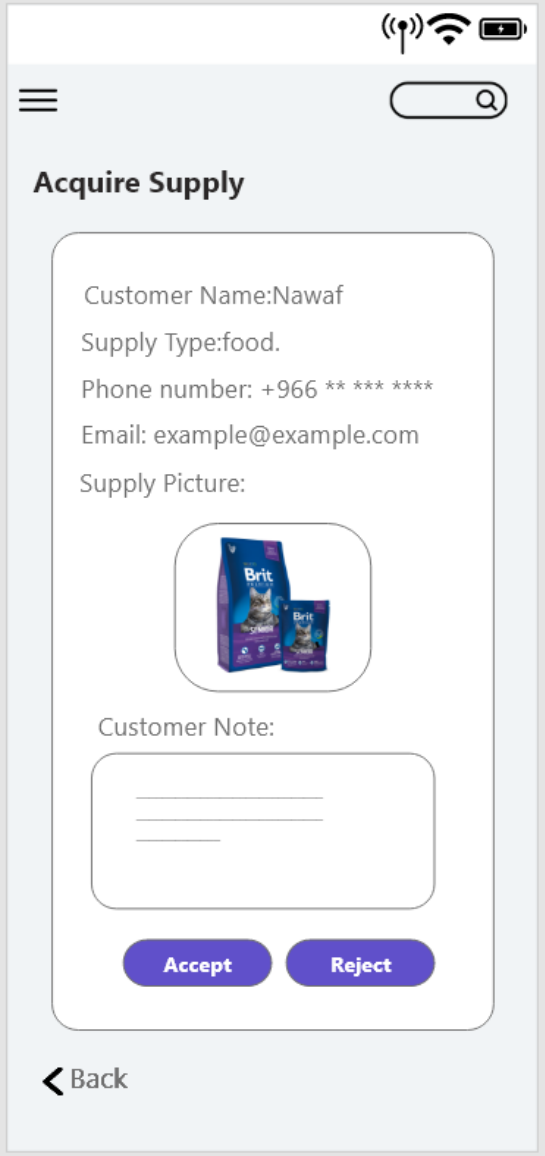
Pet Owner Interfaces

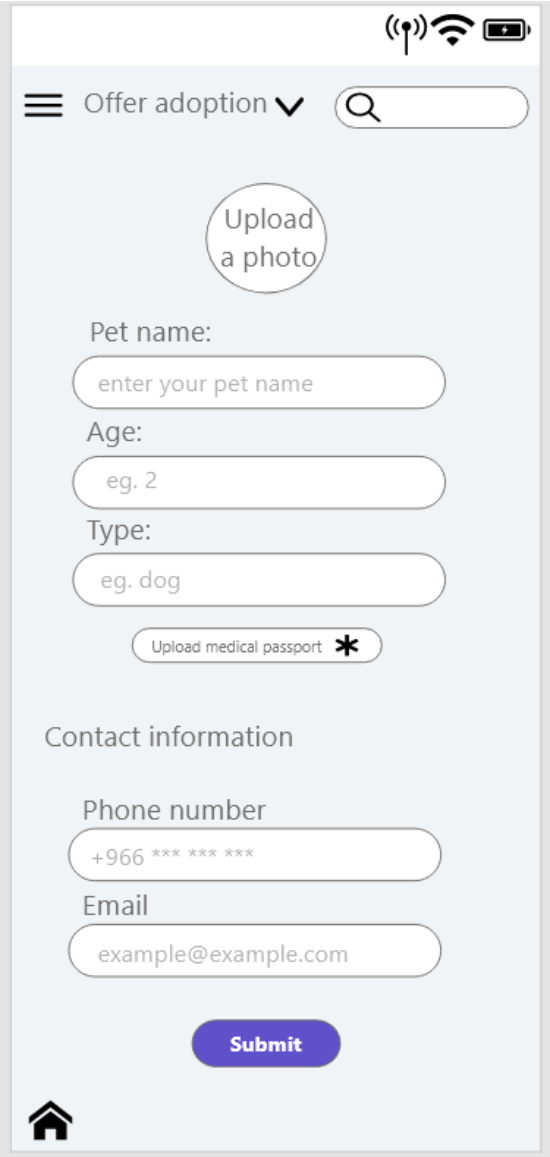


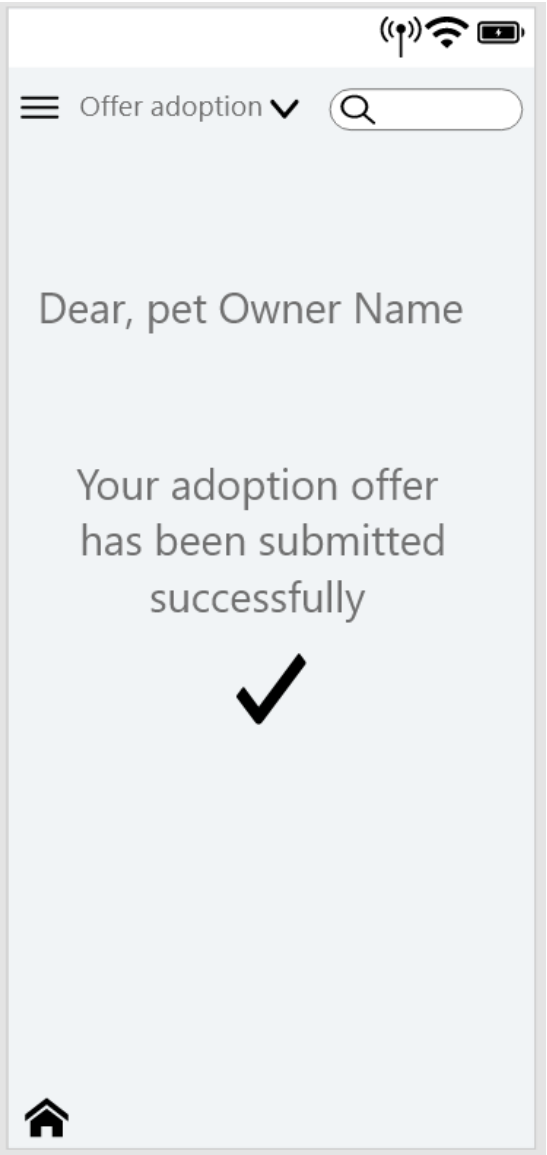


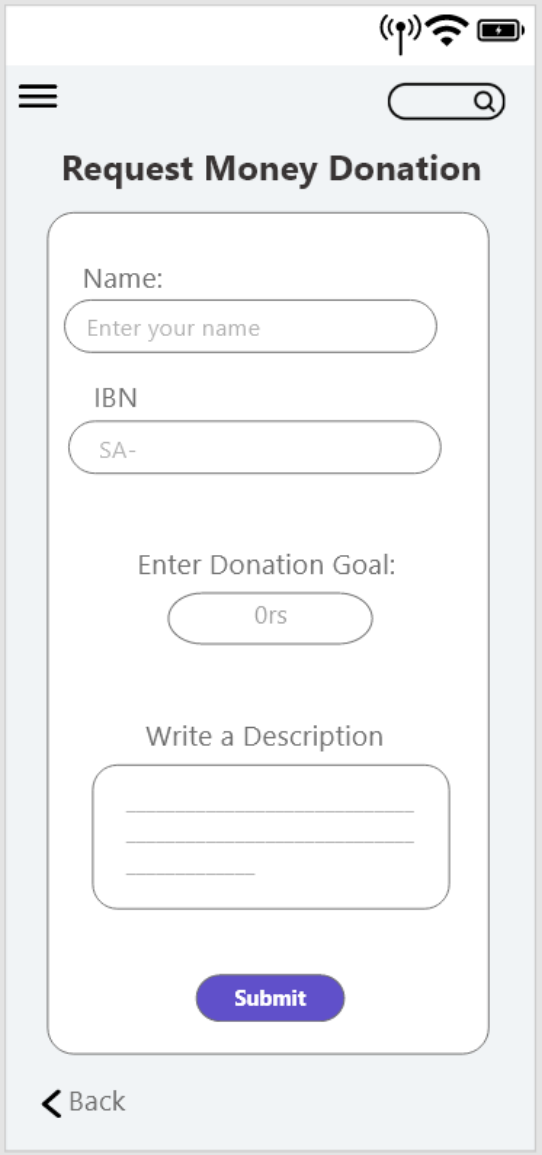

Chat Page
Customer Username
Hi,_____
Customer Username
Hi,_____
Customer Username
Hi,_____

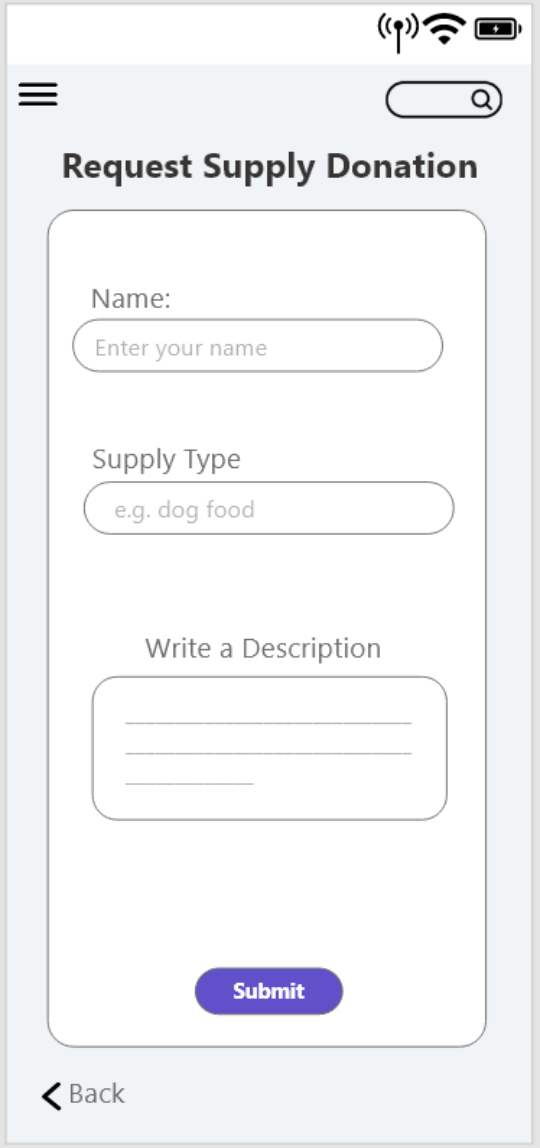

Pet Adoption Request
Customer Name:Ahmed
Customer Age:20
Phone number: +966 ** *** ****
Email: example@example.com
Customer Note:
Accept
Reject
Back

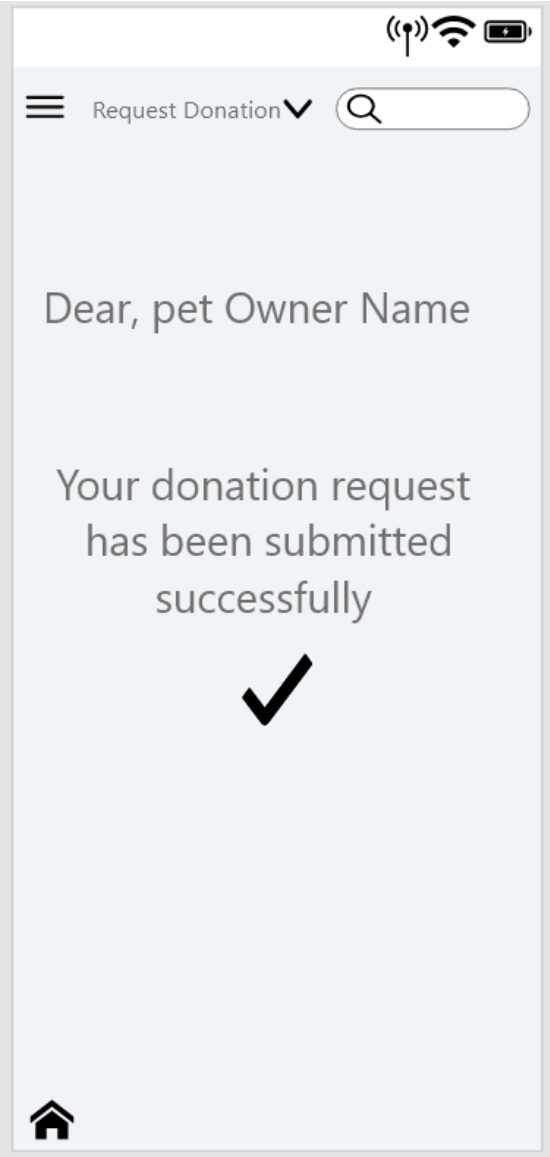

Acquire Supply
Customer Name:Nawaf
Supply Type:food.
Phone number: +966 ** *** ****
Email: example@example.com
Supply Picture:
Brit
Customer Note:
Accept
Reject
Back


Offer adoption
Upload a photo
Pet name:
enter your pet name
Age:
eg. 2
Type:
eg. dog
Upload medical passport *
Contact information
Phone number
+966 *** *** ***
Email
example@example.com
Submit

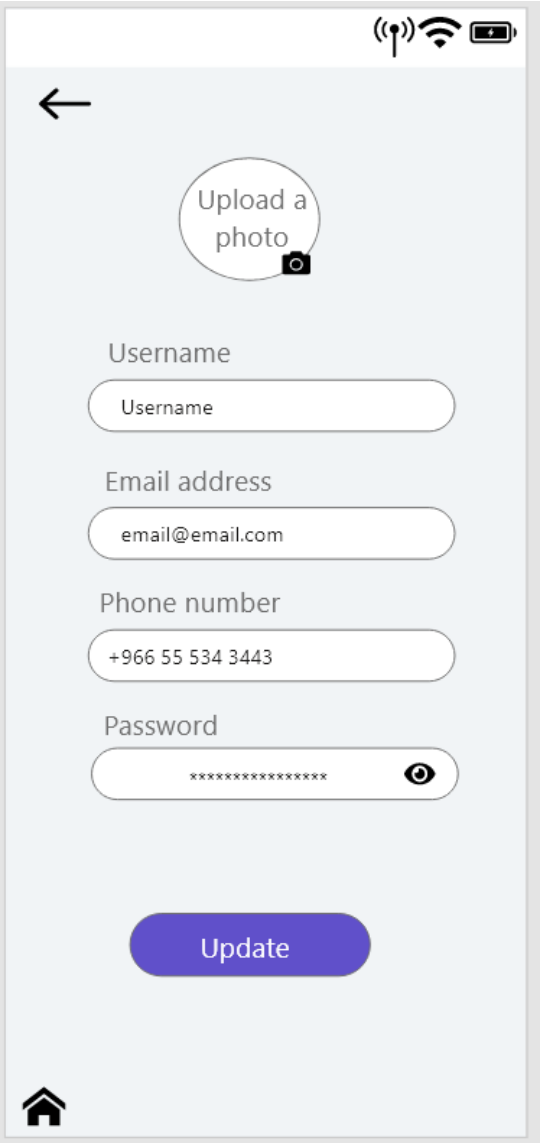

Offer adoption
Dear, pet Owner Name
Your adoption offer has been submitted successfully

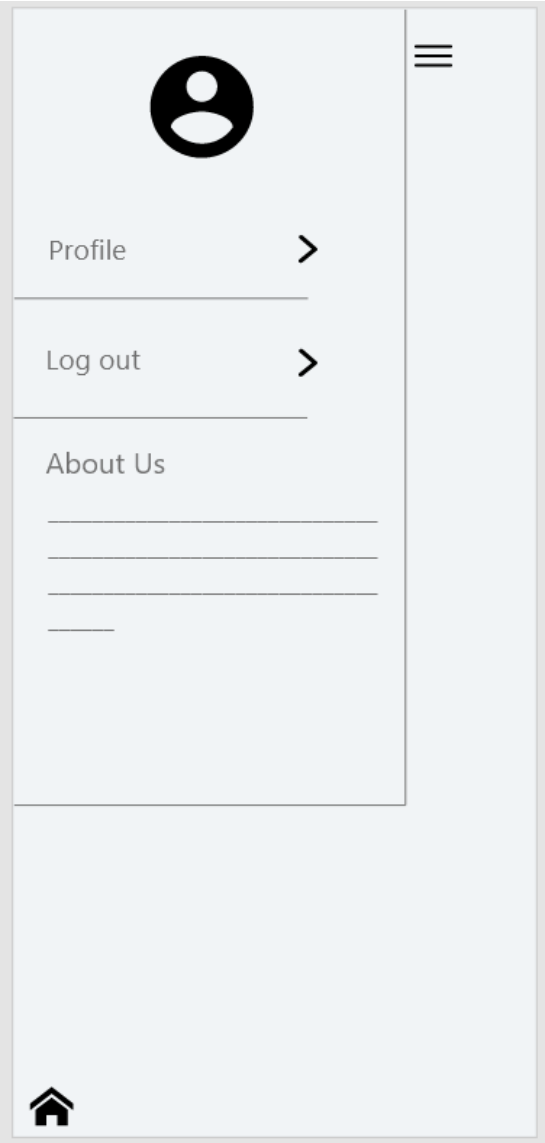

Request Money Donation
Name:
Enter your name
IBN
SA-
Enter Donation Goal:
0rs
Write a Description
Submit
Back

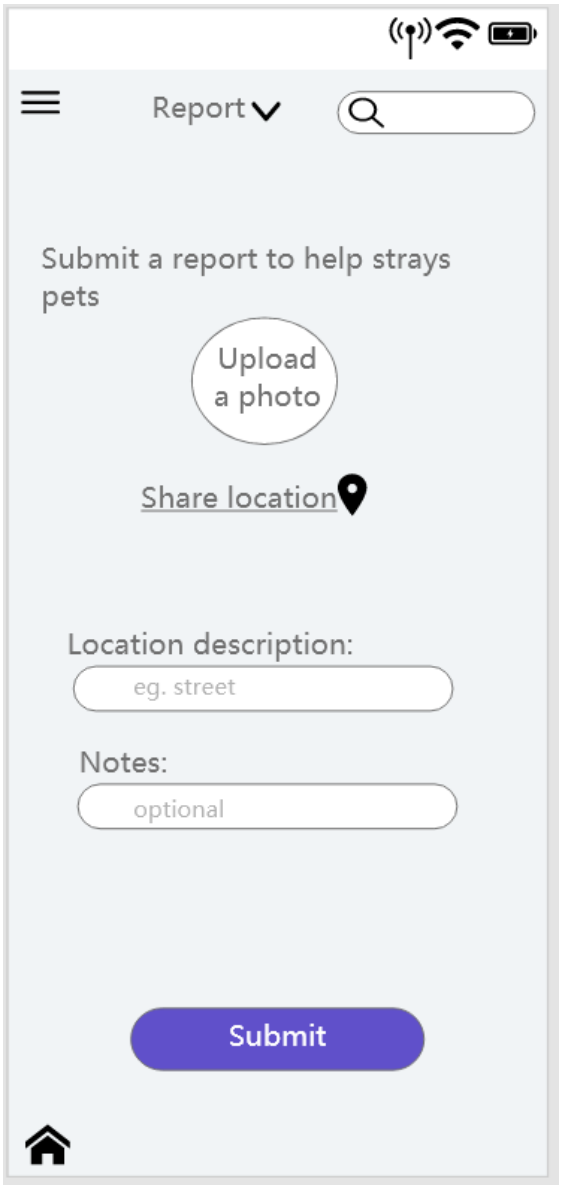

Request Supply Donation
Name:
Enter your name
Supply Type
e.g. dog food
Write a Description
Submit
Back

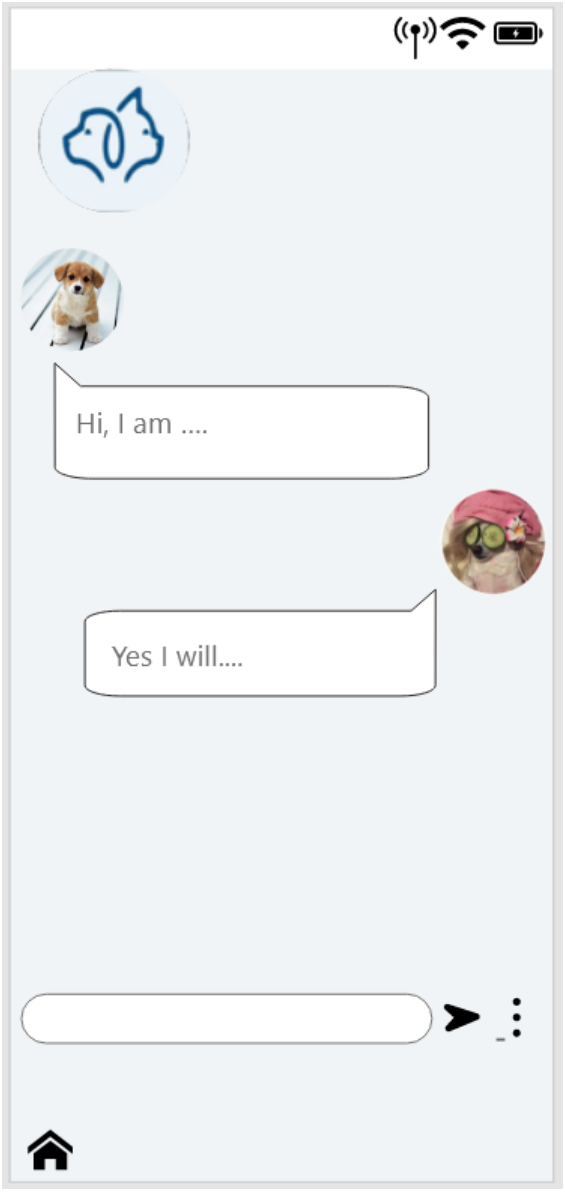

Request Donation
Dear, pet Owner Name
Your donation request has been submitted successfully

Shared Interfaces





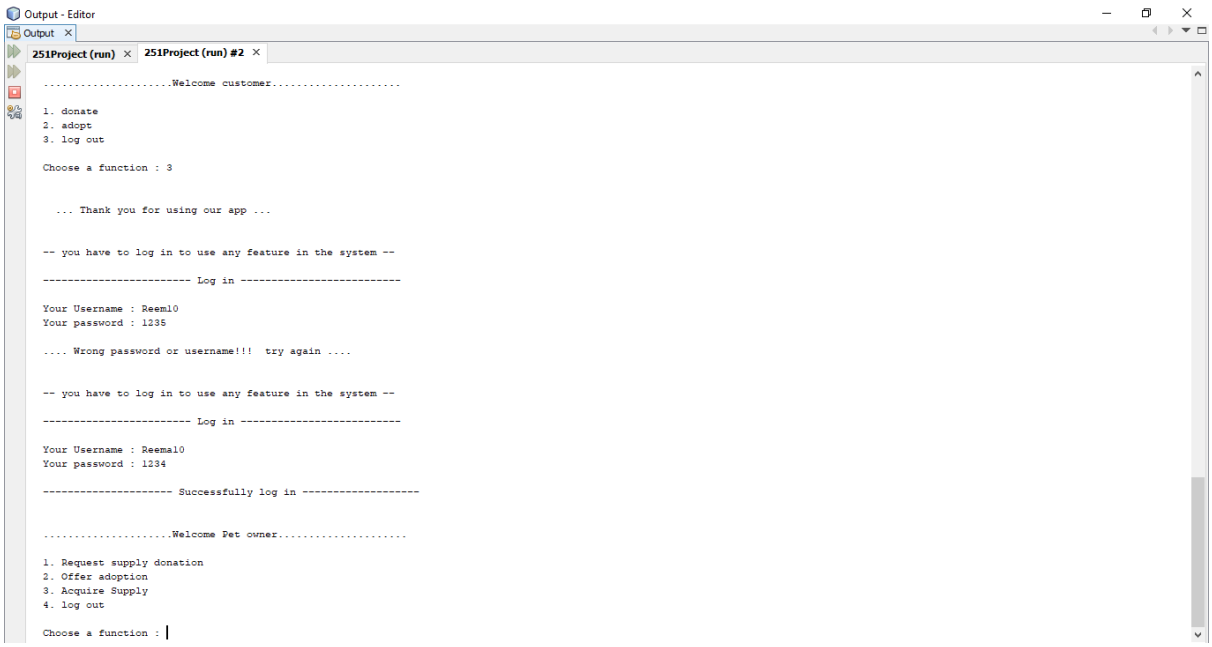

Report
Submit a report to help strays pets
Upload a photo
Share location
Location description:
eg. street
Notes:
optional
Submit


Hi, I am ....
Yes I will....

Implementation

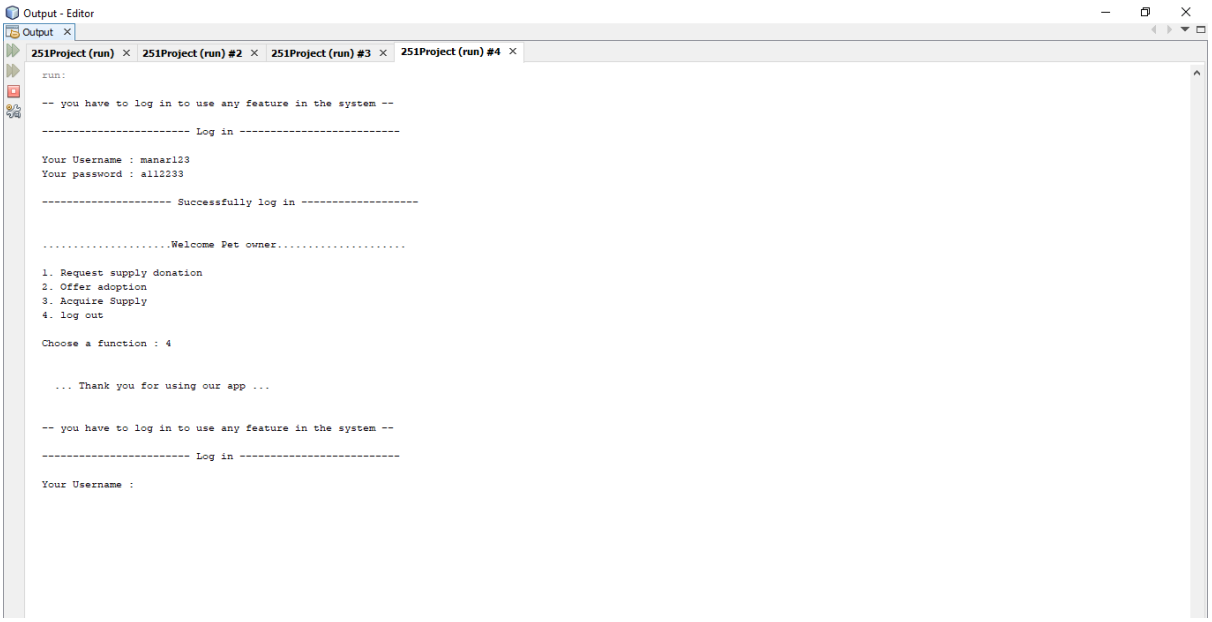
Log in & verify password Function

Reema10 is predefined account that was initialized manually



Log out Function

Any user can log out, so the system returns to the log in page



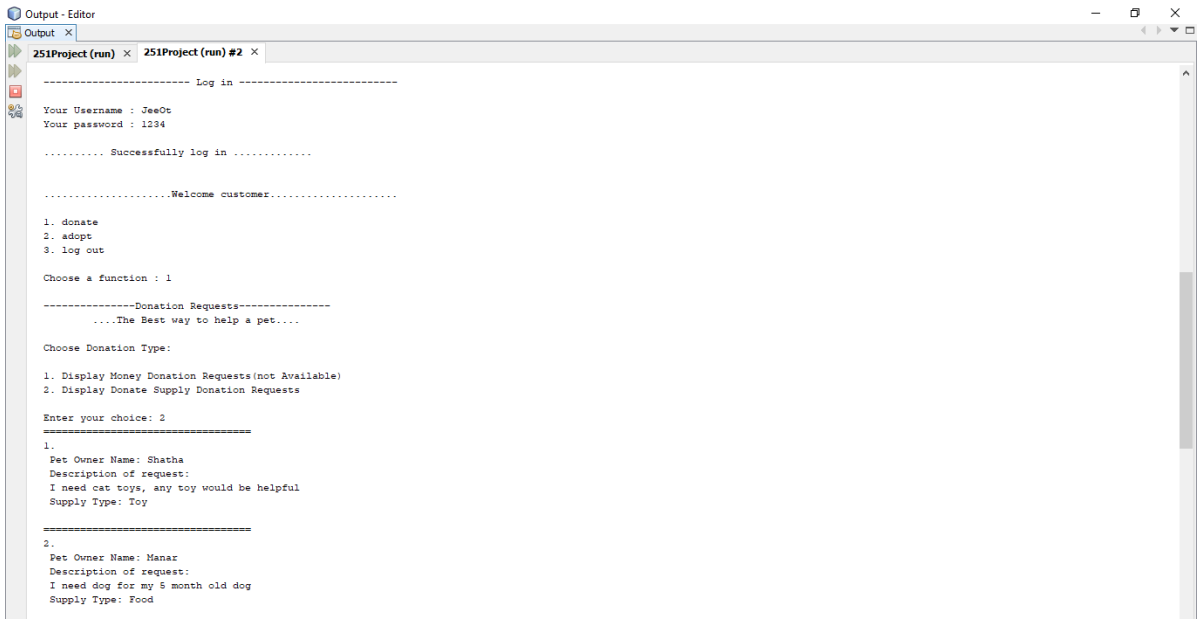
Request supply donation Function

This function is allowed for pet Owners, their requests will be added to the other requests

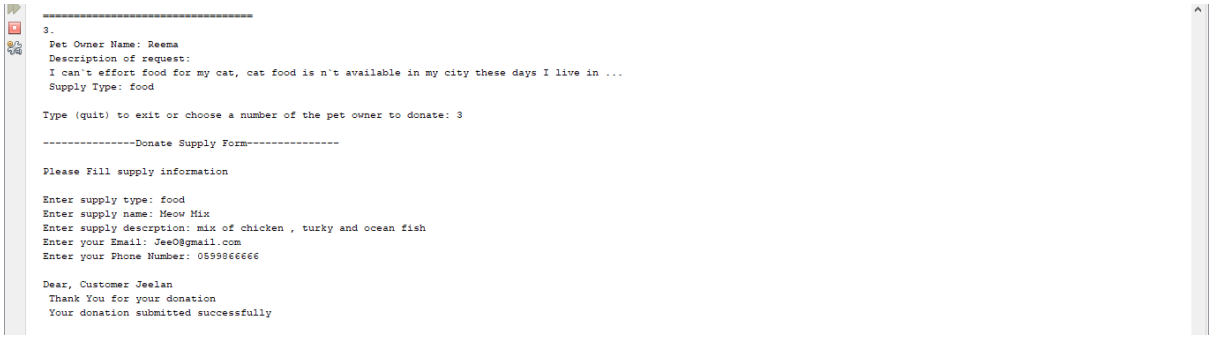
Donate supply Function

All supplies requests will be displayed to the customer to choose from.

(The previous supply request is added automatically in the donation requests)



The function invokes a form for the customer to fill donated supply information, once the use submits the form by hitting enter, a message is displayed for confirmation purposes.



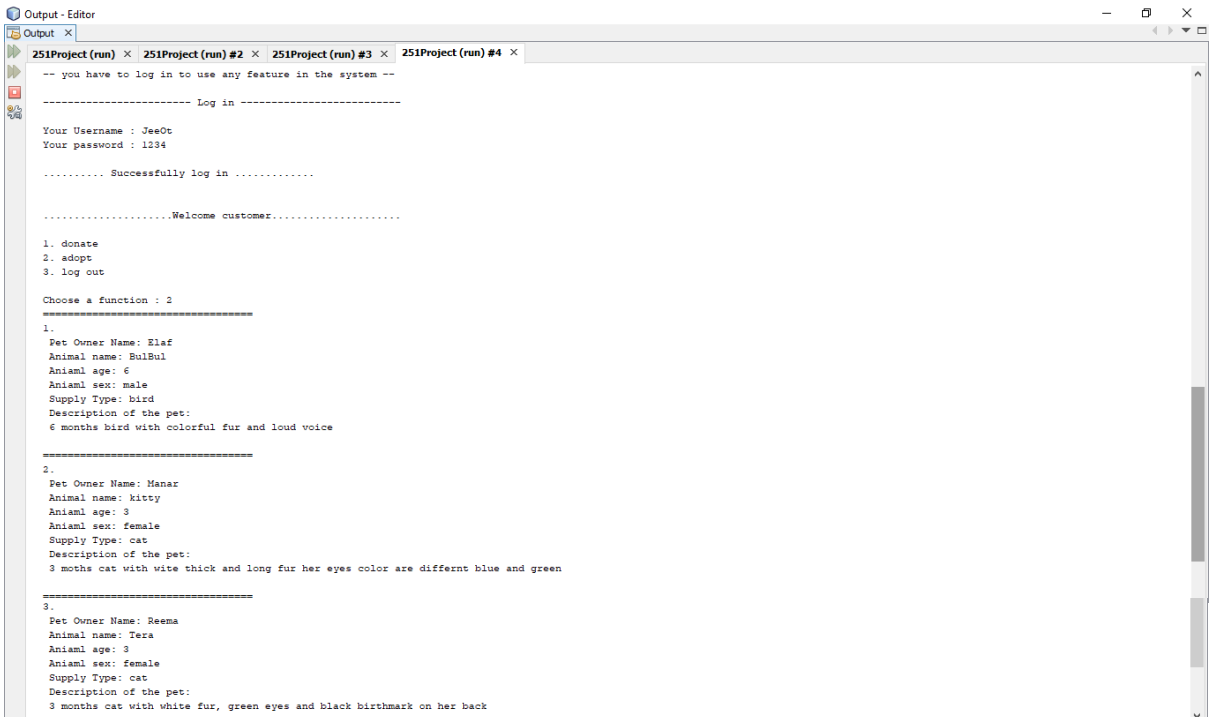
Offer adopt Function

This function is allowed for pet Owners, their offers will be added to the other offers.

Adopt Function

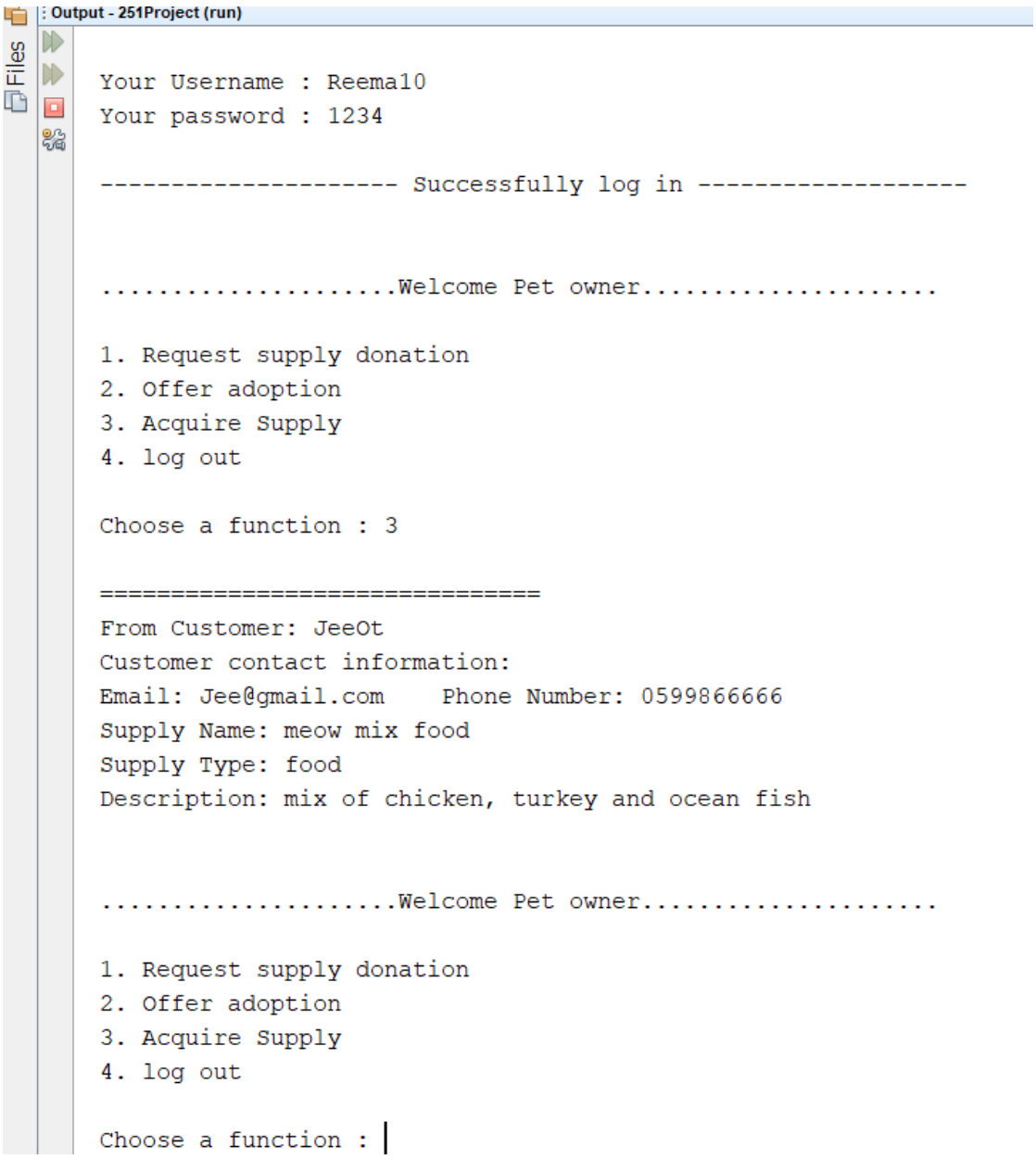
All available pets to adopt will be displayed to the customer to choose from.

(The previous offer is added automatically here you can see it)



Acquire Supply

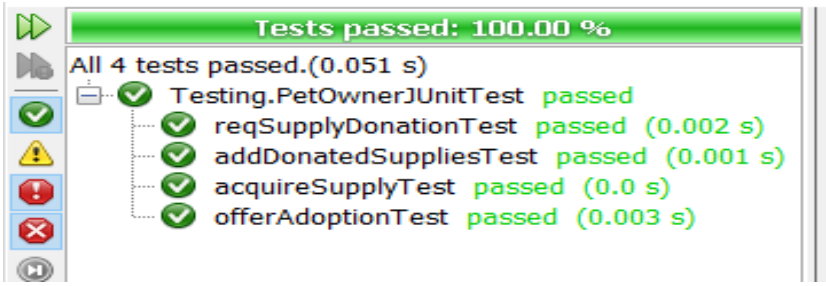
Donated supplies from customers to a pet owner are displayed



Testing Results

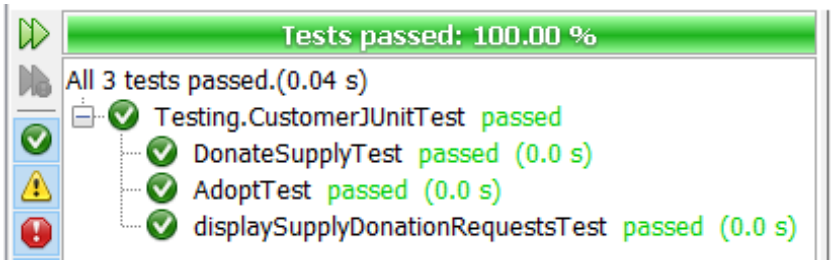
Pet Owner class

- reqSupplyDonationTest():To check if the method add the donation supply request in the donationSupplyRequets array List
- offerAdoptionTest(): Check if method successfully add the pet in the pet Offers array list
- addDonatedSuppliesTest():To ensure that the method add the supply that donate by customer to donatedSupplies array list
- acquireSupplyTest():check if method display all supplies that donated by customer to pet owner



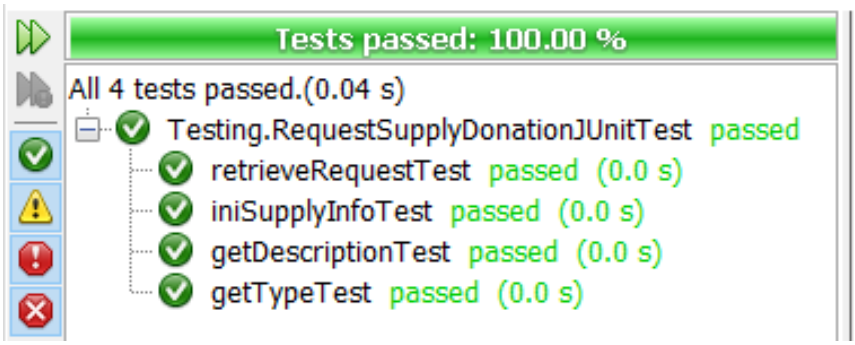
Customer class

- Adopttest(): to ensure that all offers of pet adoption are displayed to the customer
- displaySupplyDonationRequeststest():to ensure that all donation request are displayed to the customer
- DonateSupplyTest(): To check if the donated supply is added into the pet owner request successfully or not



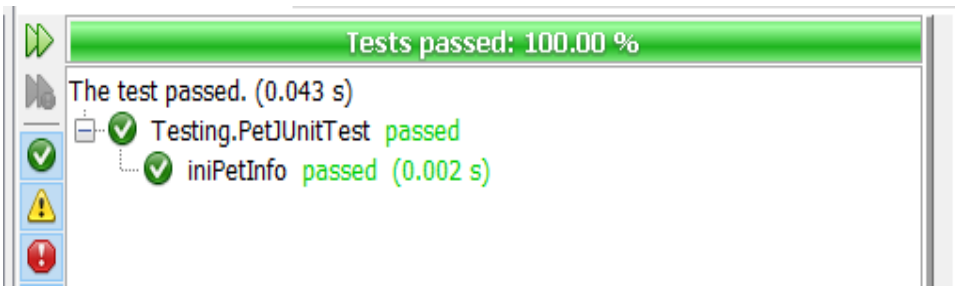
Request supply donation class

- retrieveRequestTest():ensure that the method return the right information of supply request
- getTypeTest():ensure that the method return the type of the supply
- getDescriptionTest():ensure that the method return the supple description of the supply
- iniSupplyInfoTest(): to check if the method added initial supplay information to donationSupplyRequets array List



Pet class

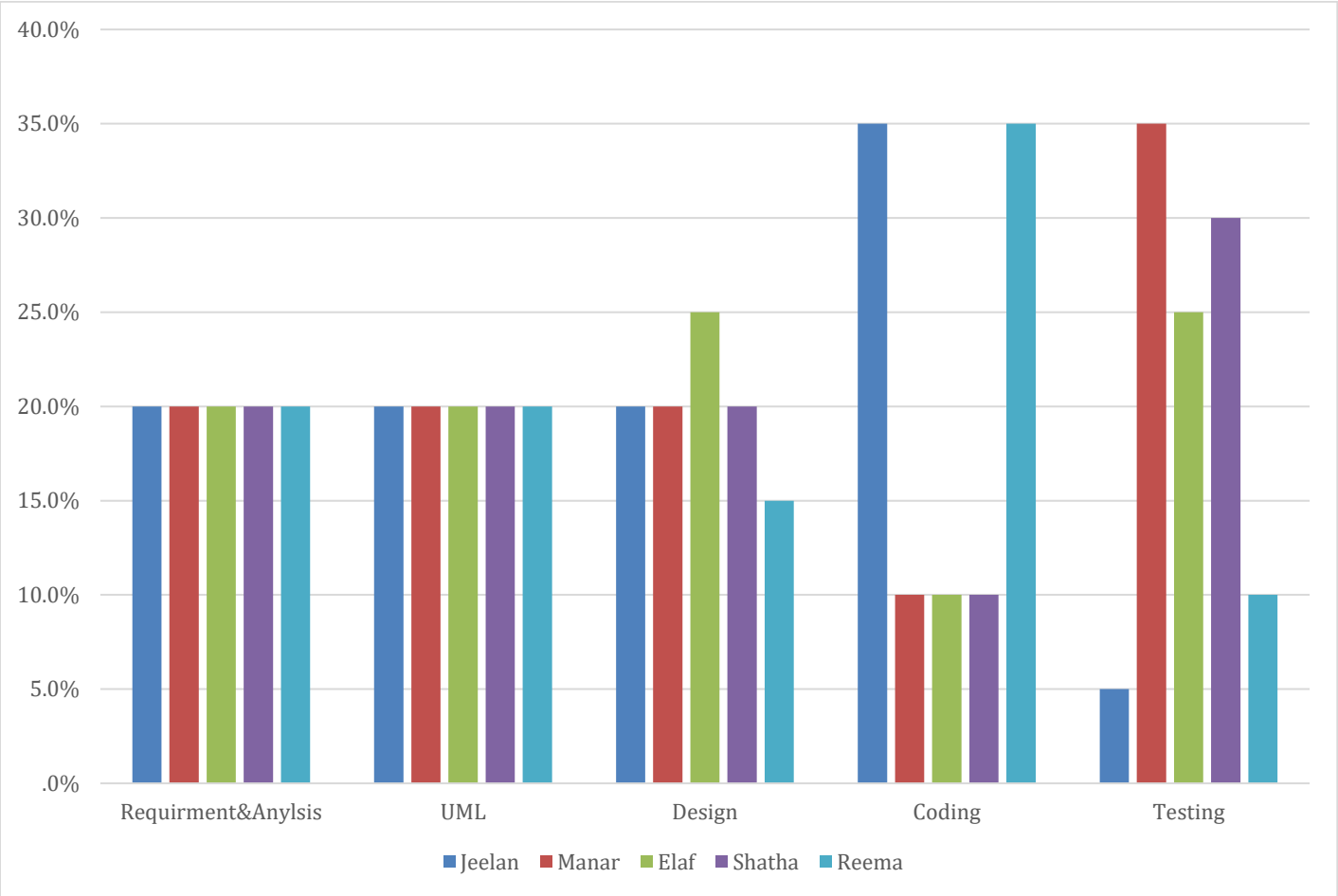
- iniPetInfo():to check if the method added pets information to pet offers array list



Evolution

This phase can`t be started until the project is complete, the code was applied on four main functions, so the project is not completed yet.

Organization of Work



Lessons Learned

- Improved out time management skills
- Learning how to deal and cooperate with team members who have different opinions
- Building a system from scratch
- How to work in waterfall methodology
- Using Junit testing

Challenges

- The differences in experience (Logic, backgrounds, and programming skills) for each member caused delays in some process
- As we are from different levels 6 and 7 our schedules were not the same which caused a difficulty to schedule a long meeting after each phase to discus and understand all our previous works before going to the next phase
- Using waterfall methodology made changes of requirements difficult after completing the requirements and analysis phase

Tools

- Lucidchart
- Google Drive
- Google docs
- GitHub
- MS Project
- Word
- Zoom
- NetBeans

Conclusion

In conclusion, we strongly believe that every animal is deserved to be happy, have good owners and live in suitable environment. Showing mercy to animals can be embodied in adoption, donation or just reporting them to be in the right place.

Future Work

- Removing pet owner and a customer two user experience and merging the functions to one user
- Customers can donate supplies for any pet owners who are interested
- Gallery feature for a pet owner user to add their offered pets in one place
- Merging Request supply and money donations page to one
- Pet Owners can open a shop within the application featuring offered pets and supplies to buy