# Faculty of Computing & Information Technology


---

Title: **PETx** Deliverable Two

| Sr #. | Roll Number | Student Name |
|---|---|---|
| 1 | BSEF19M031 | Ata Ul Mohsin |
| 2 | **BSEF19M033** | **Hassan Ahmad Sarfraz (Lead)** |
| 3 | BSEF19M037 | Muhammad Saad |
| 4 | BSEF19M047 | Muhammad Saleh Butt |

Course Name: Final Year Project

Project Supervisor: Ma’am Natalia Chaudhry

Date of Submission: Dec 26, 2022

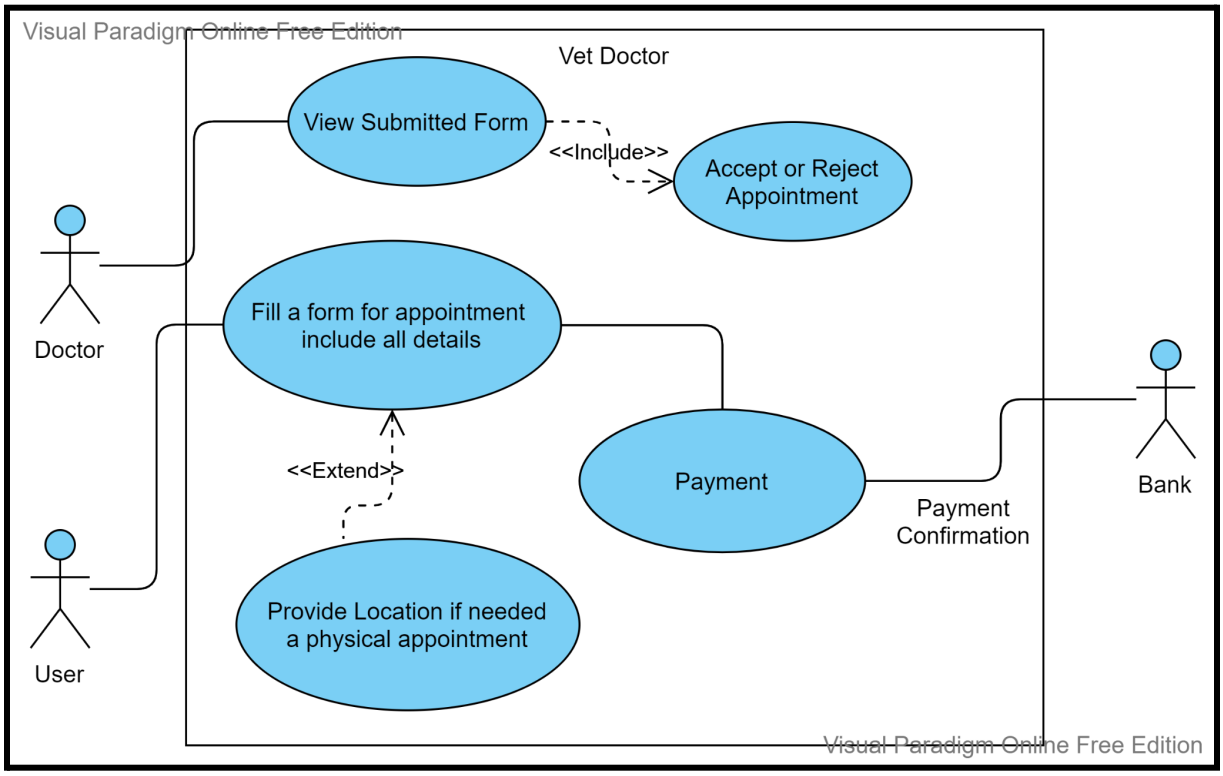
---

# Table of Contents

# Individual Use Cases

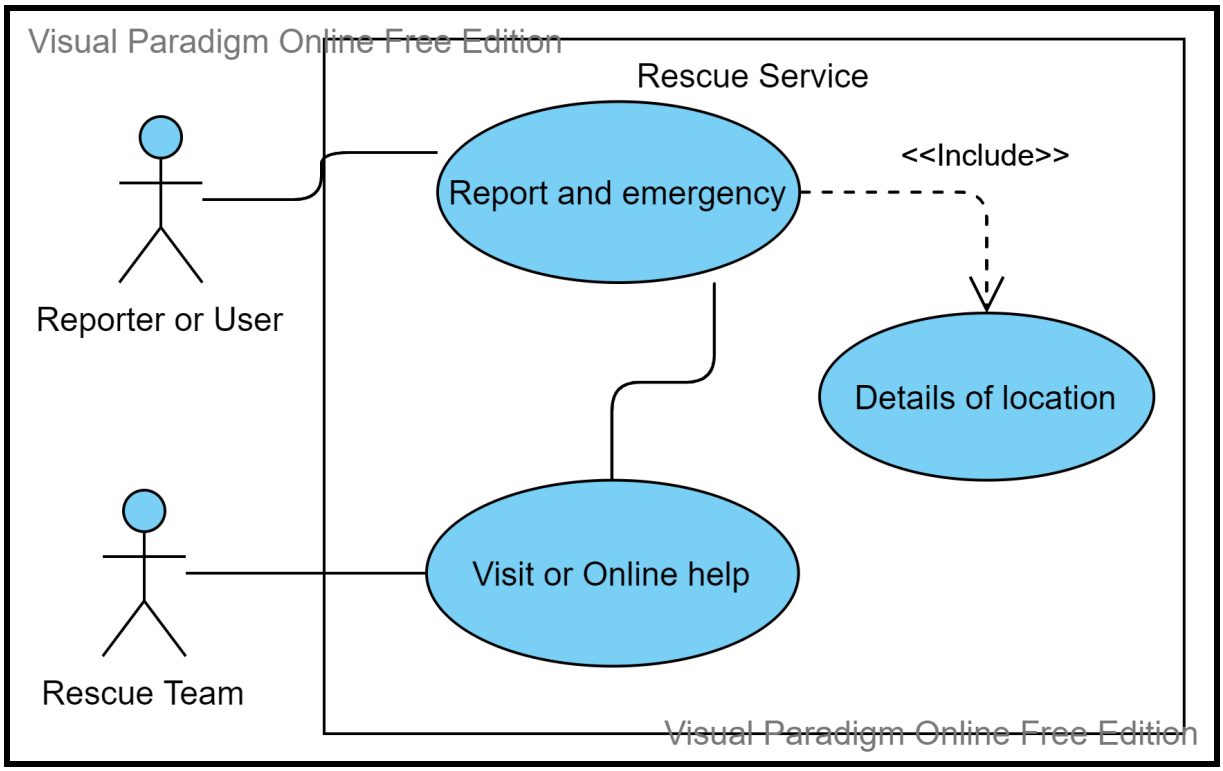
## Vet Doctor

**Goal:** Patient pet owner logs into the system and fills out a form for an appointment and selects a schedule as well. Then he pays for appointments via credit card and checks out or exits the system.
A veterinary doctor on the hand, logs in the system and can see a list of his scheduled appointments. He has the right to accept or reject the appointment or he can update the schedule of the appointment.



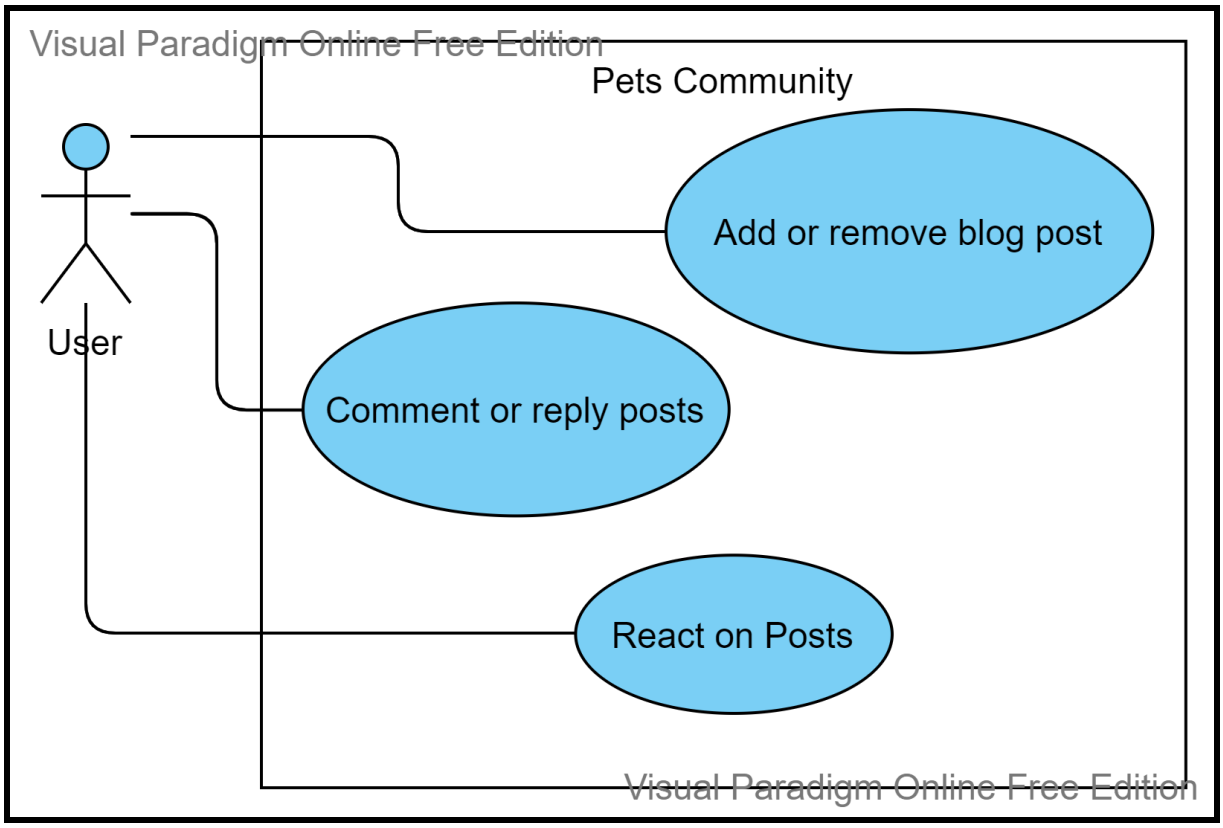

## Rescue Team

**Goals:** Any person who can be a user or reporter, reports an incident of a pet emergency and rescue team can view that in the system. If rescue teams can assist online, they do so. If online help is not possible, he can access the location of the emergency incident and visit the place for help.



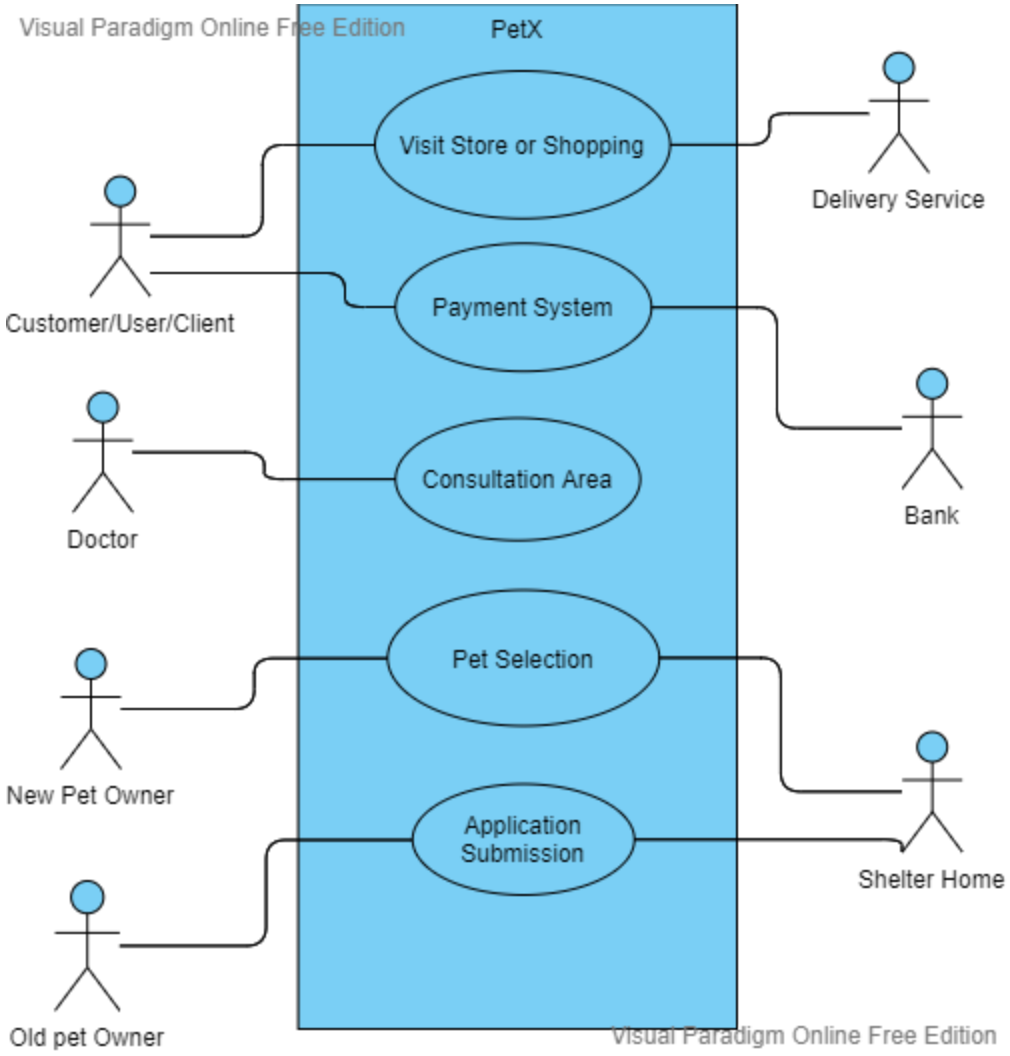

## Pets Community

**Goals:** It is a place for social hang outs online. You may call it a blog for pet owners.
Here the pet owner can browse the feed where he can react or comment on the posts of others. He can also post for others.
There'll be a dedicated area in this blog for common questions for new pet owners where a new pet owner can browse Frequently asked questions for his newly adopted pet.

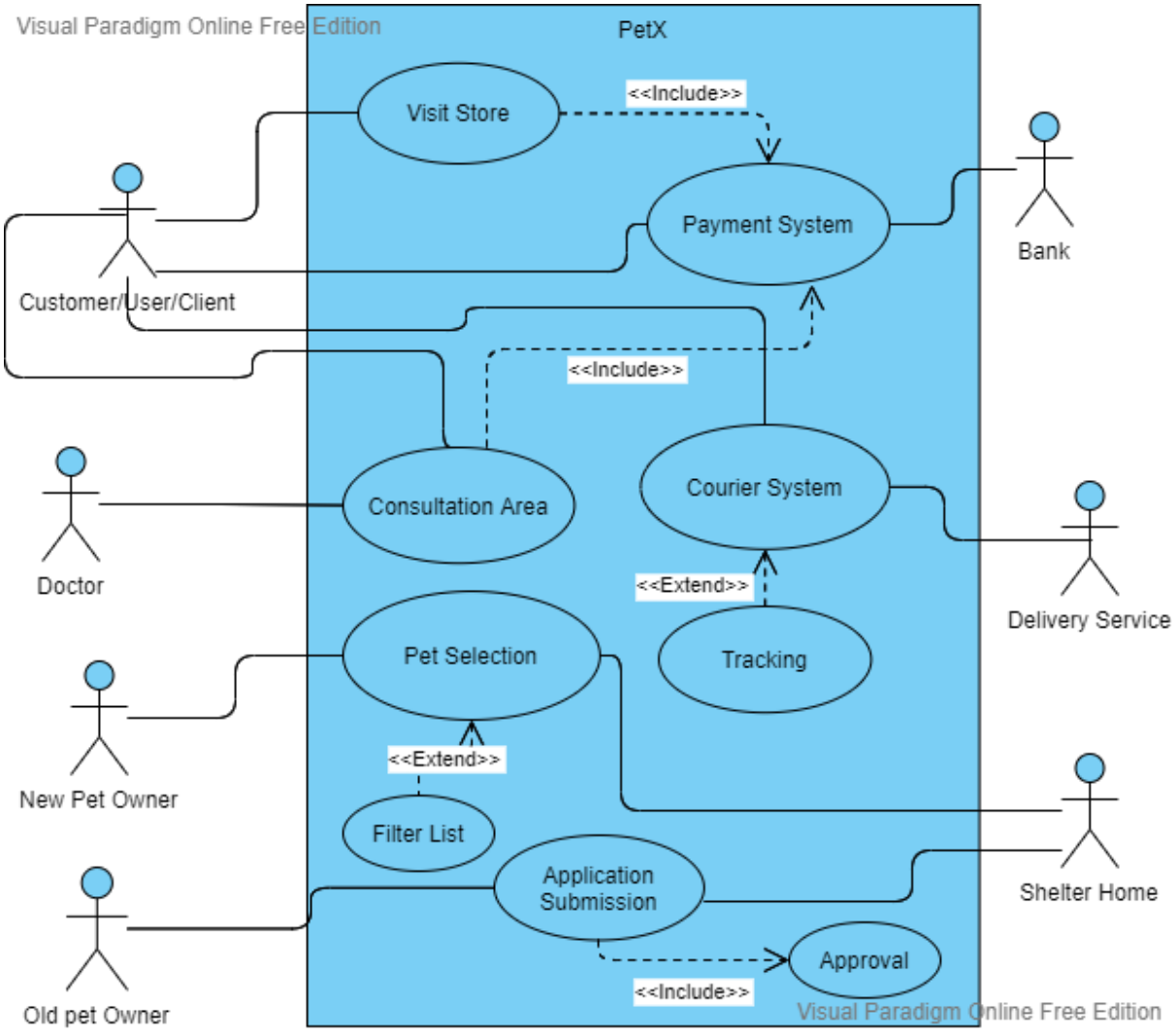



# Use Case Diagrams

## Business Level Use Case Diagram


Visual Paradigm Online Free Edition
PetX
Visit Store or Shopping
Payment System
Consultation Area
Pet Selection
Application Submission
Customer/User/Client
Doctor
New Pet Owner
Old pet Owner
Delivery Service
Bank
Shelter Home
Visual Paradigm Online Free Edition

## Analysis Level Use Case Diagram




# Descriptive Use Cases

| | |
|---|---|
| ***Sign Up*** | |
| ***Description*** | Users provide credentials to create their new account. |
| ***Actors*** | User |
| ***Pre Condition*** | isSignedUp: False |

| | | | | |
|---|---|---|---|---|
| ***Post Condition*** | New user account will be created. isSignedUp: True | | | |
| ***Primary Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | Enters Username and Password | 1 | Check if Username is unique [1(a)] or Not [1(b)]. |
| | 2 | Account accessed | 1(a) | Account created with unique Username |
| | | | | |
| ***Alternate Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | Re-Enters the Username and Password | 1(b) | Username already found, takes input again. |
| | | | | |

| | | | | |
|---|---|---|---|---|
| ***Login*** | | | | |
| ***Description*** | User provides credentials to access their already created account. | | | |
| ***Actors*** | User. | | | |
| ***Pre Condition*** | isSignedUp: Ture, isLoggedIn: False | | | |
| ***Post Condition*** | isLoggedIn: True | | | |
| ***Primary Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | Enters Username and Password | 1 | Check if Record is matched [1(a)] or Not [1(b)]. |
| | 2 | Account accessed | 1(a) | Account accessed |
| | | | | |
| ***Alternate Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | Re-Enters the Username and Password | 1(b) | Match not found, takes input again. |
| | | | | |

| | |
|---|---|
| ***Report Emergency*** | |
| ***Description*** | Users report some emergency situation to get help for some animal or pet. |

| | | | | |
|---|---|---|---|---|
| ***Actors*** | User, Rescue Team | | | |
| ***Pre Condition*** | isSent: False | | | |
| ***Post Condition*** | caseDescription: Not Null, pinLoc: Not Null, isSent: True, isTeamNotified: True, caseID: Not Null. | | | |
| ***Primary Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | User clicks the SOS button. | 1 | Emergency Form Form will be shown to the user. |
| | 2 | Describes the case in a form. | 2 | Highlight the input field for description. |
| | 3 | Adds location. | 3 | Ask for location |
| | 4 | Submit Report. | 4 | Check if input fields are empty [1(a)] or not [5]. |
| | | | 5 | Send the report to the Rescue Team and notify them. |
| | | | | |
| ***Alternate Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | Enter the specified details. | 1(a) | Ask for form resubmission. |
| | | | | |

| | | | | |
|---|---|---|---|---|
| ***Request Appointment*** | | | | |
| ***Description*** | Users will request some doctor for an appointment. | | | |
| ***Actors*** | User, Doctor | | | |
| ***Pre Condition*** | petHist: Not Null, petDetails: Not Null, isDocAvbl: True, isFeeAgreementSigned: True | | | |
| ***Post Condition*** | description: Not Null, isSent: True, approvalStatus: Pending, aptmntDateTime: Null. | | | |
| ***Primary Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | User provides a case description on request. | 1 | System provides a form to the user. |
| | 2 | User provides pet medical history (text / attachments) | 2 | Input areas are highlighted on focus. |
| | 3 | User provides pet details (breed, age, etc) | 3 | Checks if inputs are empty [1(a)] or not [4] |

| | | | | |
|---|---|---|---|---|
| | 4 | Request is submitted | 4 | Notify the doctor and wait for req approval. |
| | 5 | User is notified about the doctor's response on request. | 5 | User is notified with approval status and date time. |
| | | | | |
| ***Alternate Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | Enter the specified details. | 1(a) | Ask for form resubmission. |

| ***Read Blogs*** | | | | |
|---|---|---|---|---|
| ***Description*** | Users can read the blogs published by other users. | | | |
| ***Actors*** | Users. | | | |
| ***Pre Condition*** | isLoggedIn: True, publishedBlogs: >0 | | | |
| ***Post Condition*** | Not Required | | | |
| ***Primary Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | User can scroll through blogs and read | 1 | System will enlist all the published blogs |
| | | | | |
| ***Alternate Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | Error Message. | 1 | Blog not found or is data unreadable |

| ***Write Blogs*** | |
|---|---|
| ***Description*** | Users can write a blog and publish it. |
| ***Actors*** | User. |
| ***Pre Condition*** | isLoggedIn: True, blogStatus: Draft |
| ***Post Condition*** | blogStatus: Published |

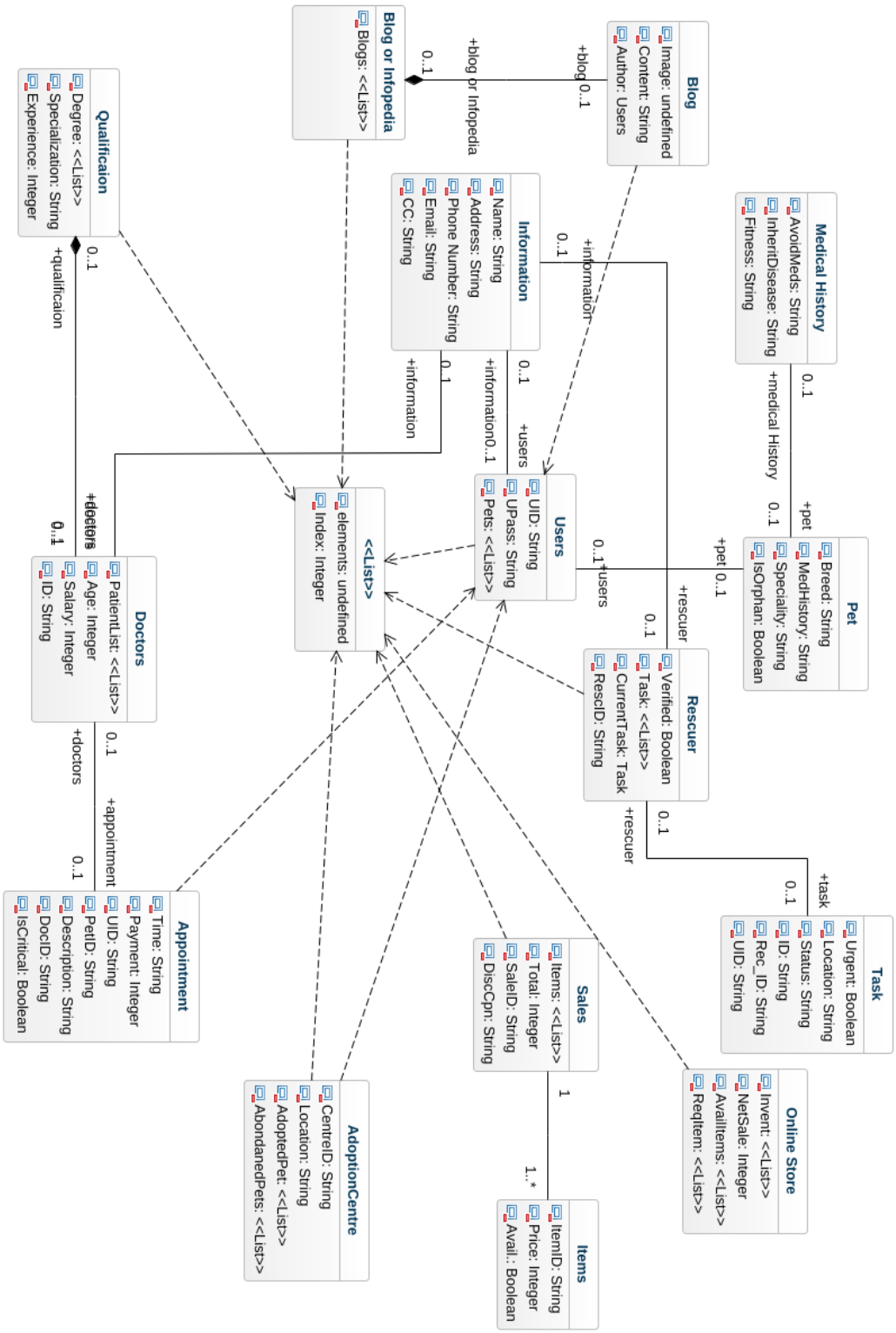
| | # | Actor | # | System |
|---|---|---|---|---|
| ***Primary Flow*** | 1 | User will click the pen icon on the bottom. | 1 | A form will be displayed. |
| | 2 | Users will write blogs and may attach pictures as well. | 2 | System will check if input isvalid [3] or not [1(a)]. |
| | 3 | Redirected to the blog reading page after publishing. | 3 | Blog is published and all users can read it. |
| | | | | |
| ***Alternate Flow*** | # | ***Actor*** | # | ***System*** |
| | 1 | Re-enters the specified details. | 1(a) | Ask for form resubmission. |

| ***Verify Card*** | | | | |
|---|---|---|---|---|
| ***Description*** | User provides the credit / debit card details for payment of the purchases. | | | |
| ***Actors*** | User, Bank | | | |
| ***Pre Condition*** | isPaymentMethodAdded: False, isCardVerified: False. | | | |
| ***Post Condition*** | isPaymentMethodAdded: True, isCardVerified: True. | | | |
| ***Primary Flow*** | # | ***Actor*** | # | ***System*** |
| | 1 | User clicks on the "Add Payment Method" button. | 1 | User is redirected to the verification form. |
| | 2 | User provides card number, expiry and CVV | 2 | Checks if inputs are valid [3] or [1(a)] |
| | 3 | User is notified with verification results. | 3 | Send data to the bank and wait for verification. |
| | | | 4 | Verified: Payment method added, Declined: [1(a)] |
| | | | | |
| ***Alternate Flow*** | # | ***Actor*** | # | ***System*** |
| | 1 | Re-enters the specified details. | 1(a) | Ask for form resubmission. |

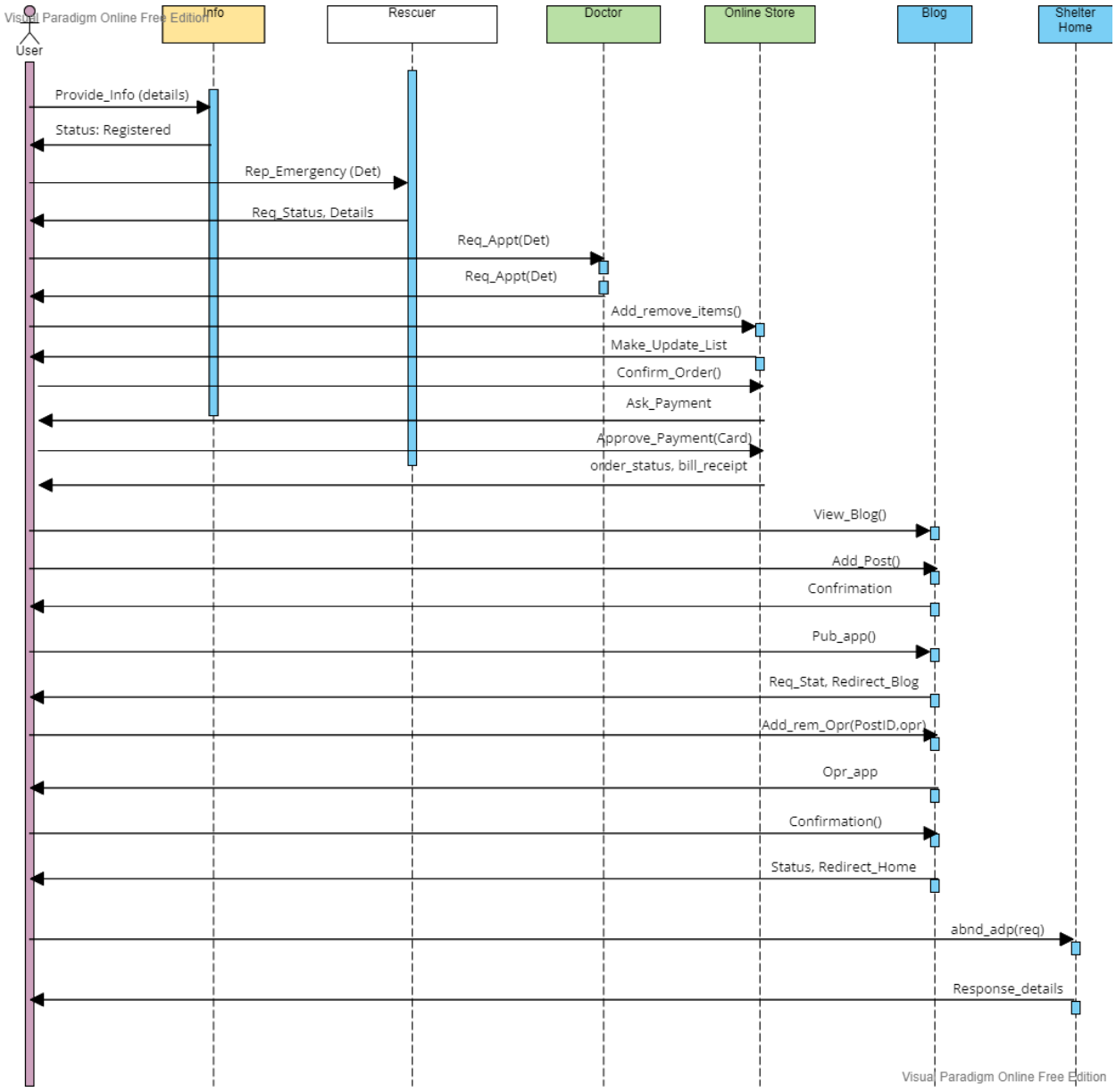
| | | | | |
|---|---|---|---|---|
| | | | | |

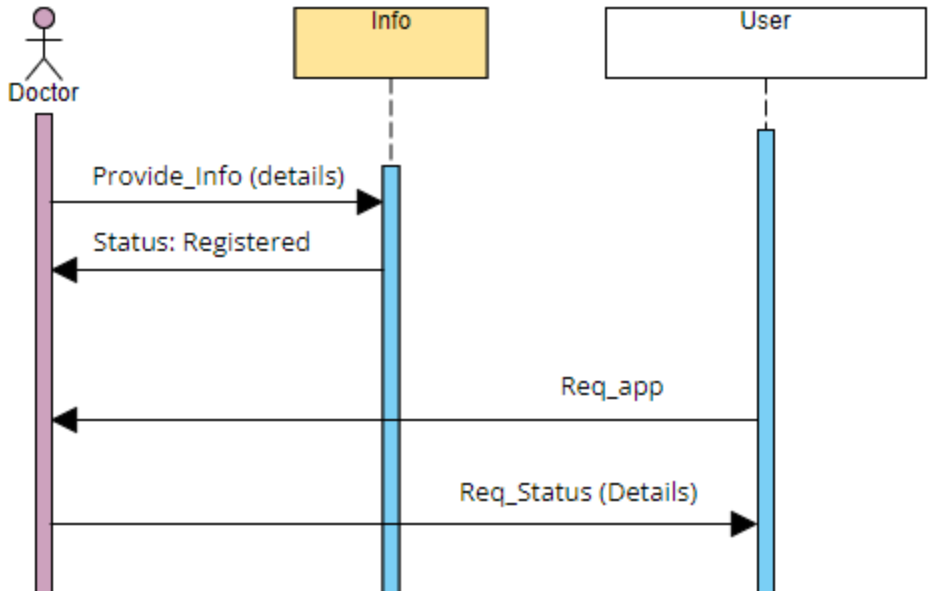
| ***Checkout*** | | | | |
|---|---|---|---|---|
| ***Description*** | User use cards to pay doctor fees and other purchases. | | | |
| ***Actors*** | User. | | | |
| ***Pre Condition*** | isPaymentMethodAdded: True, isCardVerified: True, trasacStatus: inProgress. | | | |
| ***Post Condition*** | transacStatus: Completed / Declined. | | | |
| ***Primary Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | User proceed to pay with the bill | 1 | Card is checked for Balance |
| | 2 | Purchase completed with success message. | 2 | Process transaction if bal is avbl, otherwise [1(a)]. |
| | | | | |
| ***Alternate Flow*** | ***#*** | ***Actor*** | ***#*** | ***System*** |
| | 1 | User notified with failure message. | 1 | Redirects to the checkout page. |
| | | | | |

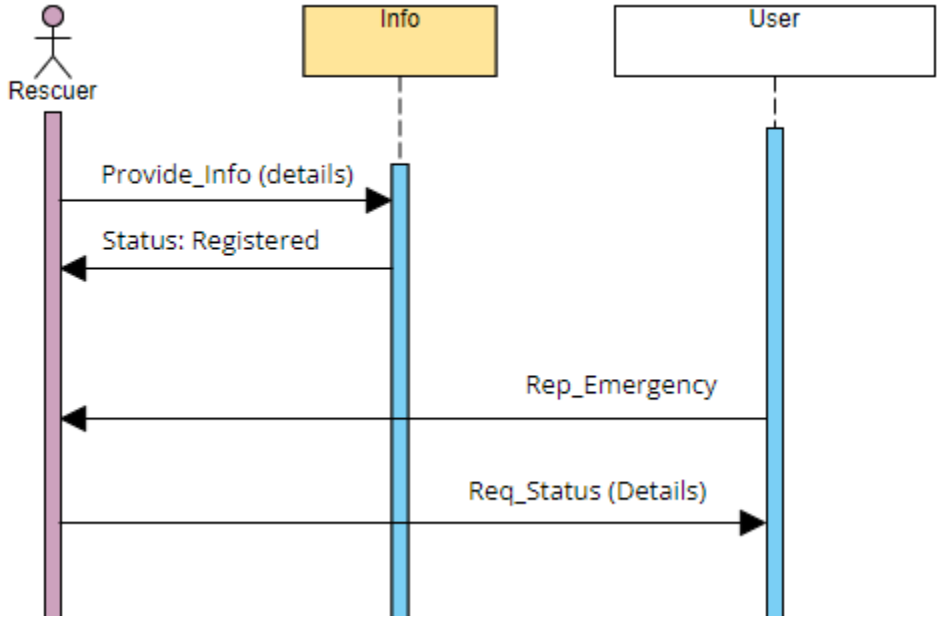
# Domain Model



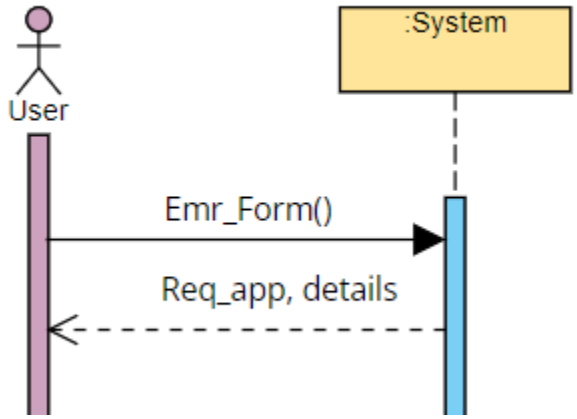
# Sequence Diagram

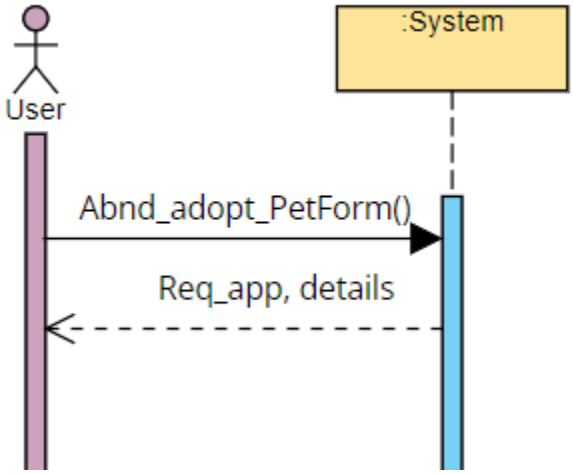


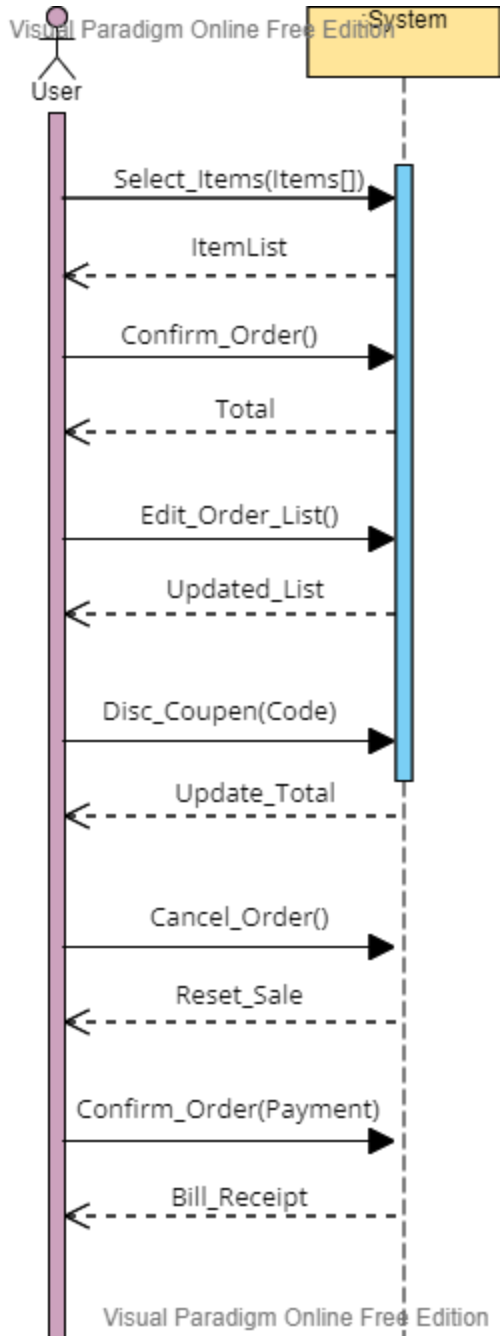

Doctor
Info
User
Provide_Info (details)
Status: Registered
Req_app
Req_Status (Details)


Rescuer
Info
User
Provide_Info (details)
Status: Registered
Rep_Emergency
Req_Status (Details)

# System Sequence Diagrams






# Operation / Use Case Contracts

## Contract 01: Card verification (Credit / Debit)

**Operation:** Card verification
**Cross-reference:** Card Verification API
**Pre-Condition:** Payment method: unverified, Checkout Eligibility: false, Service Access: View

Only.
**Post-Condition:** Payment method: verified, Checkout Eligibility: true, Service Access: View and Select.

## Contract 02: Courier delivery

**Operation:** Courier delivery
**Cross-reference**: Delivery System
**Pre-Condition:** Cart Item List: Not Null, Payment: Successful, Destination: Not Null, Destination Status: Reachable, Order Id: Not Null and Unique, Tracking ID: Null.
**Pos-Condition:** Tracking ID: Not Null, Wait Time: Not Null, Delivery Status: Delivered / On the way, Cart Item List: Null.

## Contract 03: Appointment System

**Operation:** Appointment System
**Cross-reference**: Hospital's Management System
**Pre-Condition:** Doc availability status: uncertain, Request Status: Pending, Issue Overview: Not Null, Appointment Date Time: Null, Fee Agreement: Signed.
**Pos-Condition:** Doc availability status: Occupied / Avbl, Request Status: Rejected / Approved, Appointment Date Time: Not Null and Valid.

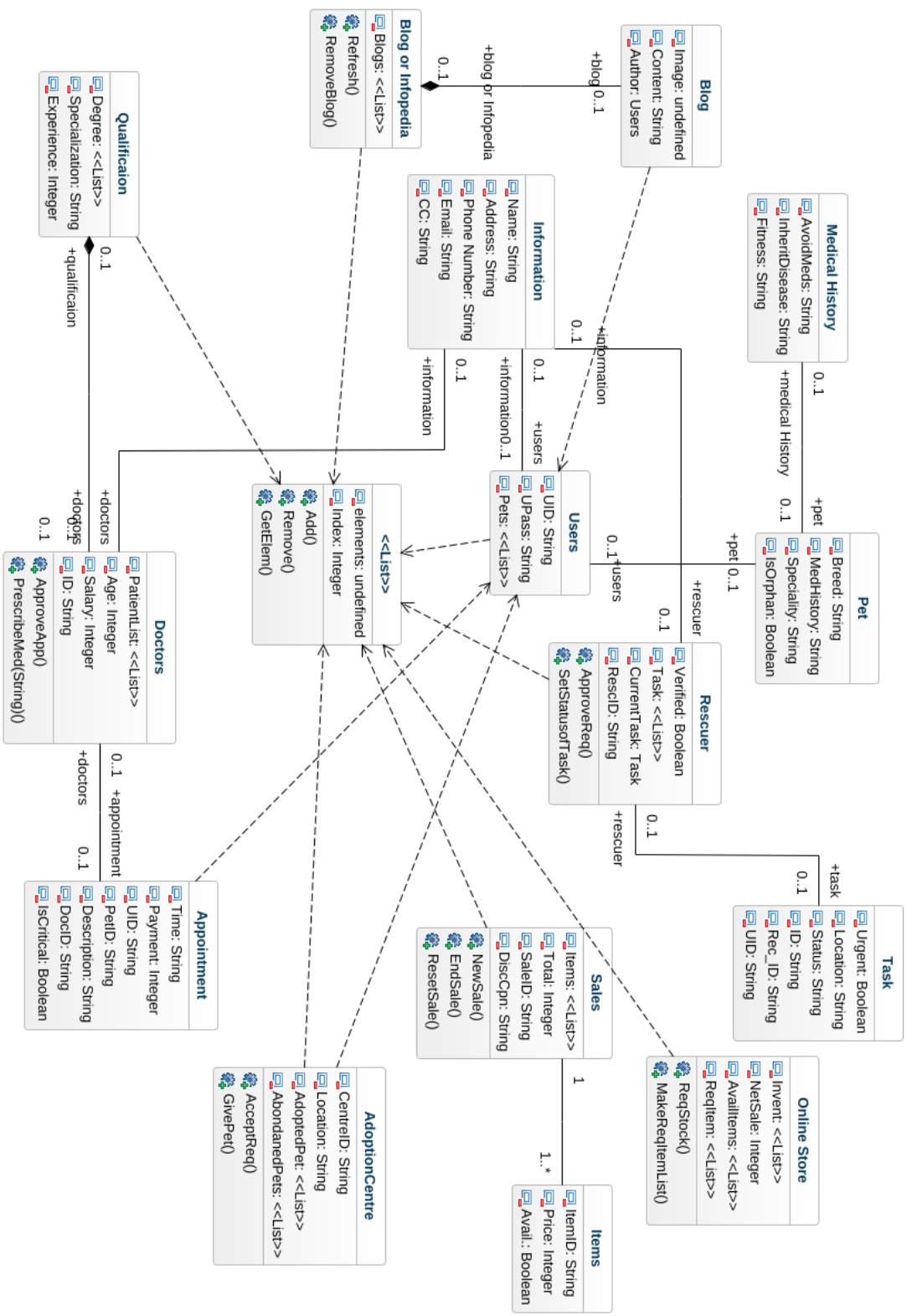
## Contract 04: Blogging System

**Operation:** Blogging System
**Cross-reference**: FBs blogging System
**Pre-Condition:** User added: true, Publishing Access: True, View Access: True, Blog body: Not Null, Blog Picture: Null / Not Null.
**Pos-Condition:** Post Status: Published / Draft, Comments: on / off.

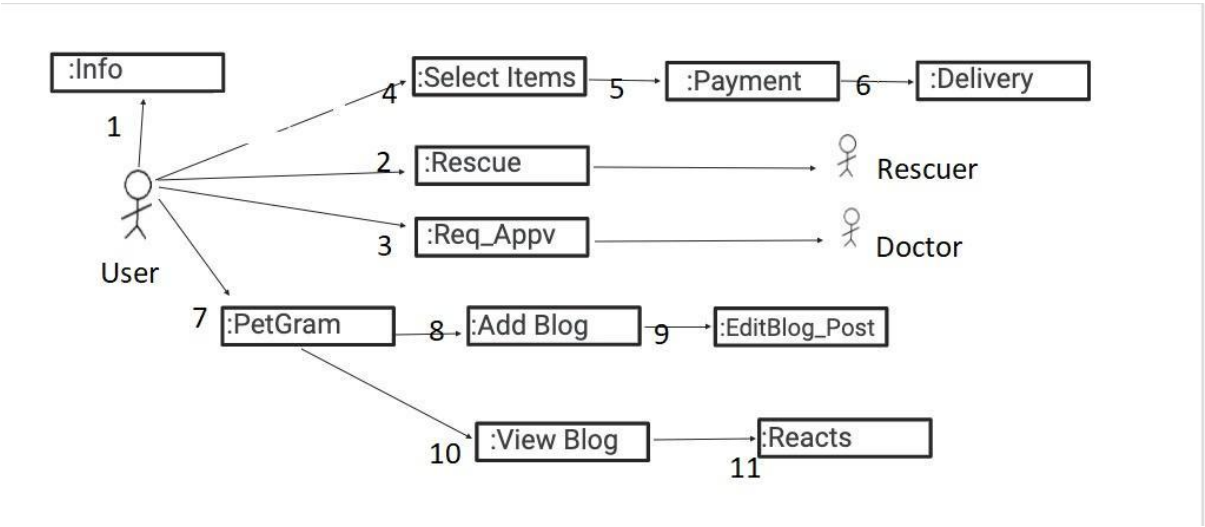
## Contract 05: Crypto Payments

**Operation:** Crypto Payments
**Cross-reference**: Distributed Ledger System.
**Pre-Condition:** Wallet Integration: Successful, Amount: >0, Currency: Selected (from the list), Amount Available: Ture, Receiver Address: Not Null and Valid, Payment Network: Selected (ERC20, BSC, etc), Password Verification: Successful.
**Pos-Condition:** Transaction Status: Successful / Failed, Balance updated: True.

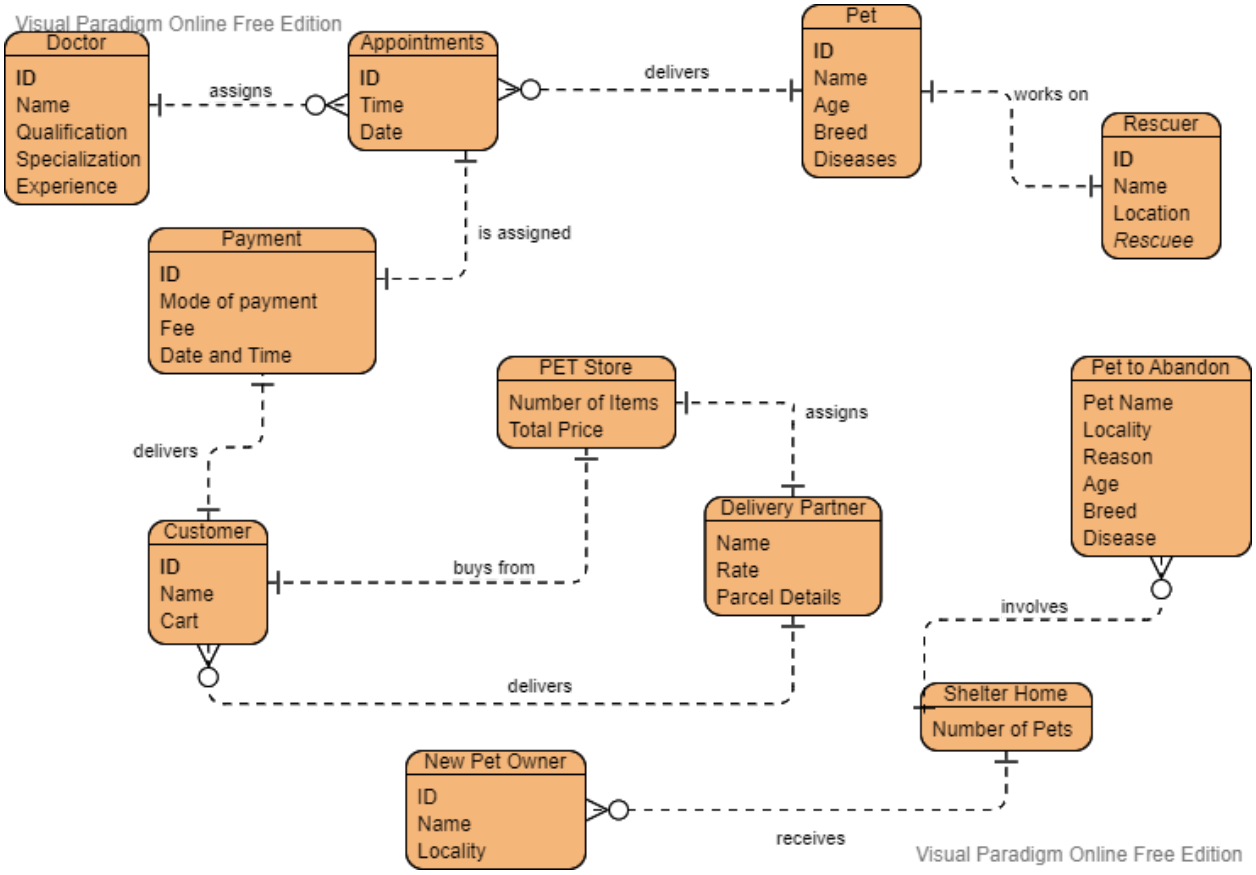
# Design Class Diagram


Blog
Image: undefined
Content: String
Author: Users
Blog or Infopedia
Blogs: <<List>>
Refresh()
RemoveBlog()
Medical History
AvoidMeds: String
InheritDisease: String
Fitness: String
Pet
Breed: String
MedHistory: String
Speciality: String
IsOrphan: Boolean
Information
Name: String
Address: String
Phone Number: String
Email: String
CC: String
Users
UID: String
UPass: String
Pets: <<List>>
Rescuer
Verified: Boolean
Task: <<List>>
CurrentTask: Task
RescID: String
ApproveReq()
SetStatusofTask()
Task
Urgent: Boolean
Location: String
Status: String
ID: String
Rec_ID: String
UID: String
Online Store
Invent: <<List>>
NetSale: Integer
AvailItems: <<List>>
ReqItem: <<List>>
ReqStock()
MakeReqItemList()
Qualification
Degree: <<List>>
Specialization: String
Experience: Integer
<<List>>
elements: undefined
Index: Integer
Add()
Remove()
GetElem()
Doctors
PatientList: <<List>>
Age: Integer
Salary: Integer
ID: String
ApproveApp()
PrescribeMed(String)()
Sales
Items: <<List>>
Total: Integer
SaleID: String
DiscCpnr: String
NewSale()
EndSale()
ResetSale()
Items
ItemID: String
Price: Integer
Avail.: Boolean
Appointment
Time: String
Payment: Integer
UID: String
PetID: String
Description: String
DocID: String
IsCritical: Boolean
AdoptionCentre
CentreID: String
Location: String
AdoptedPet: <<List>>
AbondanedPets: <<List>>
AcceptReq()
GivePet()
+blog
0..1
+blog or Infopedia
0..1
+information
0..1
+information
0..1
+information0..1
0..1
+users
0..1
+users
0..1
+medical History
0..1
+pet
0..1
+pet
0..1
+rescuer
0..1
+rescuer
0..1
+task
0..1
+qualification
0..1
+doctors
0..1
+doctors
0..1
+doctors
0..1
+appointment
0..1
1
1..*

# Collaboration Diagram




# Data Model


Visual Paradigm Online Free Edition
Doctor
ID
Name
Qualification
Specialization
Experience
assigns
Appointments
ID
Time
Date
delivers
Pet
ID
Name
Age
Breed
Diseases
works on
Rescuer
ID
Name
Location
Rescuee
is assigned
Payment
ID
Mode of payment
Fee
Date and Time
PET Store
Number of Items
Total Price
assigns
Pet to Abandon
Pet Name
Locality
Reason
Age
Breed
Disease
delivers
Customer
ID
Name
Cart
buys from
Delivery Partner
Name
Rate
Parcel Details
involves
delivers
Shelter Home
Number of Pets
New Pet Owner
ID
Name
Locality
receives
Visual Paradigm Online Free Edition