

Faculty of Computing & Information Technology



Title: **“PETx” FYP Documentation**

Group ID: **BSEF19-17**

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

Project Supervisor: Ma'am **Natalia Chaudhary**

Date of Submission: Jun 15, 2023

Table of Contents

Serial Number	Document	Page Number
1	FYP Registration Form	3
2	PETx Proposal	4
3	Deliverable One	14
4	Deliverable Two	38

PUNJAB UNIVERSITY COLLEGE OF INFORMATION TECHNOLOGY

FINAL PROJECT REGISTRATION FORM

Degree SE Session F19-M Type Research
Date: 7th Nov, 22

No: _____

Project Title: PETx

Proposed Primary Advisor's Information:

Name: NATALIA CHAUDHARY





Address PUCIT, OLD CAMPUS, KATCHERY ROAD, UNIVERSITY OF THE PUNJAB, LAHORE.

Ph. No. Res. - Office - Mobile: 03224118071

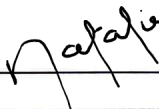
Group Leader:

Roll No. BSEF19M033 Mobile no: 03039636478

Name. HASSAN AHMAD SARFRAZ

	Roll Number (Group member)	Name of Group member	Email	Signature
1.	BSEF19M031	ATA UL MOHSIN	BSEF19M031@PUCIT.EDU.PK	
2.	BSEF19M037	Muhammad Saad	BSEF19M037@PUCIT.EDU.PK	
3.	BSEF19M047	M SALEH BUTT	BSEF19M047@PUCIT.EDU.PK	
4.	BSEF19M033	HASSAN AHMAD	BSEF19M033@PUCIT.EDU.PK	

Supervisor Signature:



(For Official Use Only)

☐ **Recommended**

☐ Meeting Required: Date: _____ Time: _____ Place: _____

☐ **Rejected**

Remarks: _____

Project Title (Revised): PETx

Project Office Supervisor



PUCIT
Punjab University College of Information Technology

Final Project Proposal

Version 1.0

TABLE OF CONTENTS

Final Project Proposal	3
1. Introduction	3
1.1 Project Title	3
1.2 Project Overview Statement	3
1.3 Project Overview Statement	4
1.4 Project Goals & Objectives	5
1.5 High-level system components	5
1.6 List of optional functional units	5
1.7 Exclusions	5
1.8 Application Architecture	6
1.9 Gantt chart	7
1.10 Hardware and Software Specification	9
1.11 Tools and technologies used with reasoning	9

1. Introduction

With the fast-paced and everyday developing world, we have excelled in almost every field related to humans. Everyone is pushing the limits to facilitate humanity. In this race of serving humanity, we have, somehow, overlooked animals and their facilities. Being human poses some responsibilities to us that also includes a duty to protect the living creatures around us. We don't even have sufficient time for our pets, from our busy and social life. This project is specially chosen to provide a simple and easy approach to avail pet-related services to pet owners so that their pets can get what they need at the right time.

1.1 Project Title

PETx

1.2 Project Overview Statement

As the name suggests, the project revolves around animal care and services related to them. Since everyone is so engaged in their lives, no one finds enough time to research their pet, which doctor is better, and where to find the best quality pet products. In case, some animal is ever struck in a do-or-die situation, no one knows how to help the poor soul. Thus keeping all this in mind, PETx will provide all these services with ease so that pet owners can serve their pets with peace of mind.

1.3 Project Overview Statement

Project Title: PETx			
Group Leader: Hassan Ahmad Sarfraz			
Project Members:			
Name	Registration #	Email Address	Signature
ATA UL MOHSIN	BSEF19M031	bsef19m031@pucit.edu.pk	<i>AUMohsin</i>
HASSAN AHMAD SARFRAZ	BSEF19M033	bsef19m033@pucit.edu.pk	<i>HassanAhmad</i>
MUHAMMAD SAAD	BSEF19M037	bsef19m037@pucit.edu.pk	<i>SaadAmjad</i>
MUHAMMAD SALEH BUTT	BSEF19M047	bsef19m047@pucit.edu.pk	<i>MSaleh</i>
Project Goal: To create a safe space for pets in the community by providing their owners with multiple pet-related services on a single platform. Also encouraging users to use his crypto for payments.			
Objectives:			
Sr.#			
1	Develop a rescue system for pets		
2	Community building through a social interaction corner for pet owners.		
3	Add online veterinary appointment system		
4	Add blockchain-based payment systems		
Project Success criteria: Develop an all-in-one place for pet-related services so that pet owners can find the best for their pets with minimum effort on a single platform.			
Assumptions, Risks, and Obstacles: <ul style="list-style-type: none">It is assumed that there are some service providers working already in the market. We will be developing a platform to onboard them and unite their services on a single web app.There is a risk that providers do not provide the promised service quality which can result in a dent in our reputation and loss of customers as well.Limited tech knowledge in hand.			
Organization Address (if any):			
Type of project: Research			
Target End users: Pet Owners, Pet Rescuers, Doctors			
Development Technology: Object Oriented			
Platform: Web-based			
Suggested Project Supervisor: Natalia Chaudhary <i>Natalia</i>			
Approved By:			
Date: Nov 4, 2022			

1.4 Project Goals & Objectives

Goals:

- 1) To help pet owners take the best care of their pets.
- 2) To make it easy for people to adopt pets by providing ease in services.
- 3) To reduce the number of animal deaths by providing immediate first aid services.
- 4) To help people show affection to injured animals by using our rescue service.
- 5) To make revenue by making it easy for humans to treat animals with kindness.

Objectives:

To achieve the above goals we are determined to:

- 1) Develop a rescue system for pets
- 2) Add online veterinary appointment system
- 3) Community building through a social interaction corner for pet owners.
- 4) Add blockchain-based payment systems

1.5 High-level system components

- 1) Login / SignUp: Leads the user to Landing Page
- 2) Rescue Service: No login required, Immediate contact to rescue team
- 3) Appointment Section: Browse through doctors list, see reviews and choose from available appointment times.
- 4) Blog page: Users can share their insights, findings, research, experiences, or other information with each other.
- 5) Payment: Blockchain-based secure payment system

1.6 List of optional functional units

- 1) Online store: Where users can buy all kinds of pet accessories, food, and medicines.
- 2) Shelter Home: Where you can submit some stray / orphan animals, adopt a pet, or might abandon your pet as well.

1.7 Exclusions

Functionalities: Due to limited time, we might not be able to practically implement optional functionalities.

Performance: Due to a very limited budget, we would not be able to buy subscriptions for an excellent host thus performance can be compromised.

Traffic: We will also face traffic limitations that might not be able to handle a large number of online users at once.

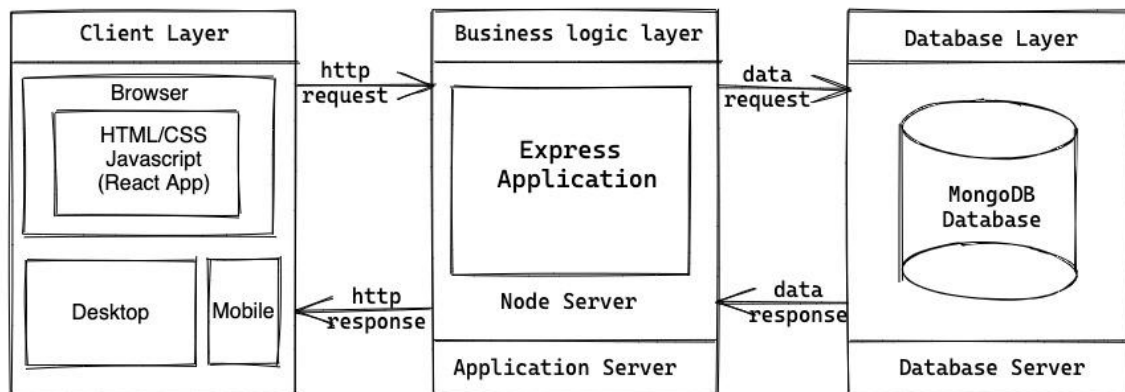
Domain name: Domain name has a key role in attracting an audience. Due the limited budget, we would be buying any popular domain like .com etc, instead, we will be any less attractive one that would fit in our budget.

1.8 Application Architecture

We will be using Three-Tier architecture for this project. These 3 layers will be as

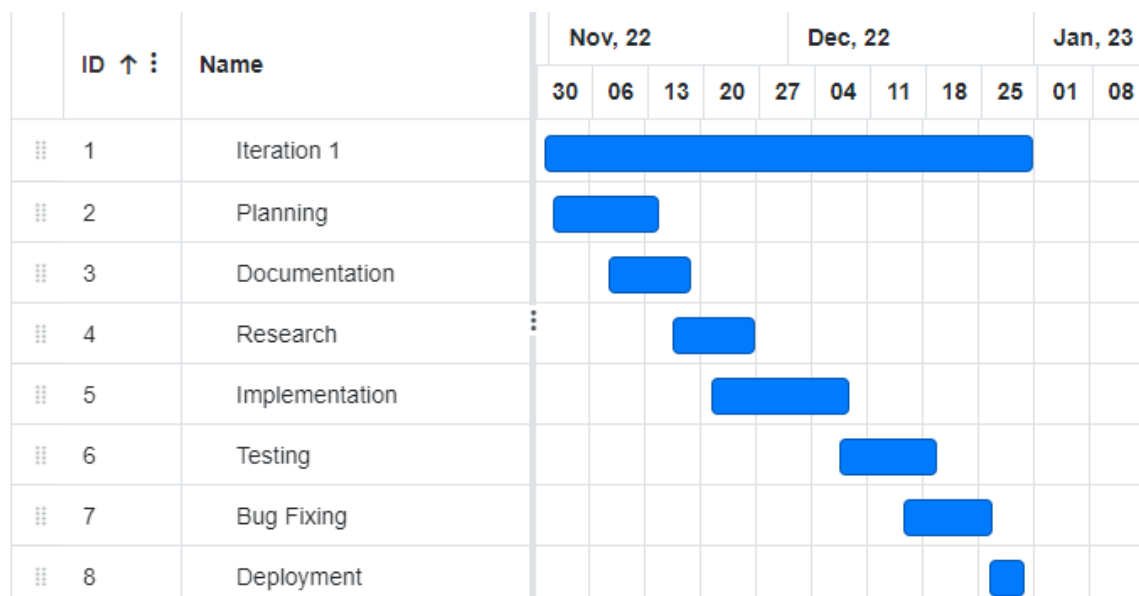
- 1) Frontend Tier
- 2) Application Tier
- 3) Database Tier

Frontend will be handled using HTML5, CSS3, ES6, and ReactJS. The application tier will be implemented using NodeJS and ExpressJS. While the last layer of the database will be implemented using MongoDB.



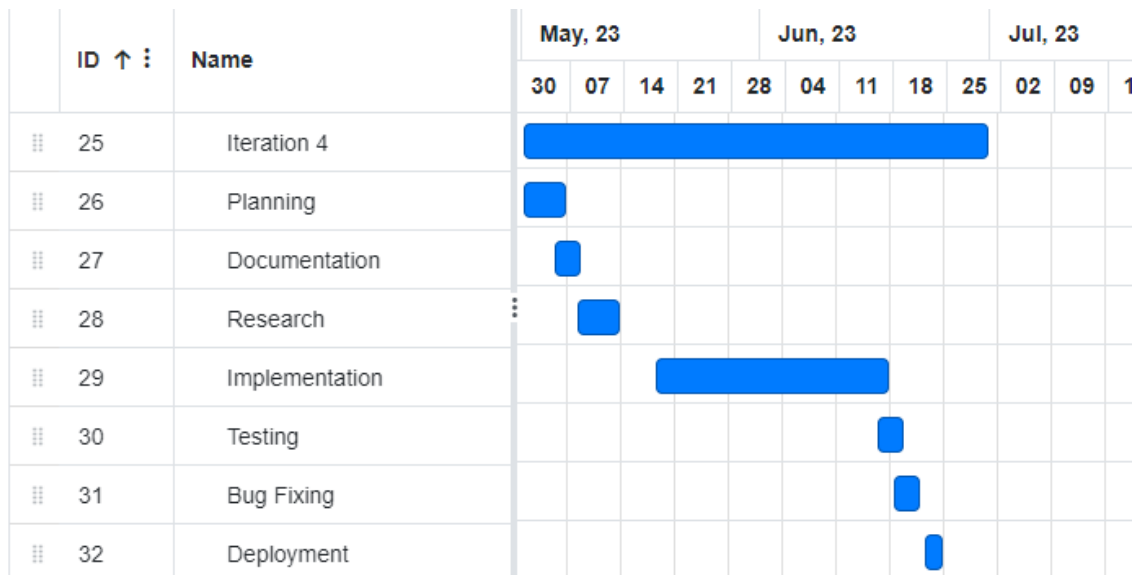
1.9 Gantt chart

ID	Name	Start Date	End Date	Duration
1	Iteration 1	Oct 31, 2022	Dec 31, 2022	62 days
2	Planning	Nov 01, 2022	Nov 14, 2022	14 days
3	Documentation	Nov 08, 2022	Nov 18, 2022	11 days
4	Research	Nov 16, 2022	Nov 26, 2022	11 days
5	Implementation	Nov 21, 2022	Dec 08, 2022	18 days
6	Testing	Dec 07, 2022	Dec 19, 2022	13 days
7	Bug Fixing	Dec 15, 2022	Dec 26, 2022	12 days
8	Deployment	Dec 26, 2022	Dec 30, 2022	5 days
9	Iteration 2	Jan 01, 2023	Feb 28, 2023	59 days
10	Planning	Jan 01, 2023	Jan 07, 2023	7 days
11	Documentation	Jan 08, 2023	Jan 14, 2023	7 days
12	Research	Jan 15, 2023	Jan 17, 2023	3 days
13	Implementation	Jan 16, 2023	Feb 10, 2023	26 days
14	Testing	Feb 12, 2023	Feb 15, 2023	4 days
15	Bug Fixing	Feb 16, 2023	Feb 20, 2023	5 days
16	Deployment	Feb 19, 2023	Feb 22, 2023	4 days
17	Iteration 3	Mar 01, 2023	Apr 30, 2023	61 days
18	Planning	Mar 01, 2023	Mar 08, 2023	8 days
19	Documentation	Mar 08, 2023	Mar 15, 2023	8 days
20	Research	Mar 15, 2023	Mar 18, 2023	4 days
21	Implementation	Mar 19, 2023	Apr 15, 2023	28 days
22	Testing	Apr 16, 2023	Apr 20, 2023	5 days
23	Bug Fixing	Apr 20, 2023	Apr 22, 2023	3 days
24	Deployment	Apr 21, 2023	Apr 27, 2023	7 days
25	Iteration 4	May 01, 2023	Jun 30, 2023	61 days
26	Planning	May 01, 2023	May 06, 2023	6 days
27	Documentation	May 05, 2023	May 08, 2023	4 days
28	Research	May 08, 2023	May 13, 2023	6 days
29	Implementation	May 18, 2023	Jun 17, 2023	31 days
30	Testing	Jun 16, 2023	Jun 19, 2023	4 days
31	Bug Fixing	Jun 18, 2023	Jun 21, 2023	4 days
32	Deployment	Jun 22, 2023	Jun 24, 2023	3 days



	ID ↑ :	Name	Jan, 23				Feb, 23				Mar, 23		
			01	08	15	22	29	05	12	19	26	05	12
⋮	9	Iteration 2											
⋮	10	Planning											
⋮	11	Documentation											
⋮	12	Research											
⋮	13	Implementation											
⋮	14	Testing											
⋮	15	Bug Fixing											
⋮	16	Deployment											

	ID ↑ :	Name	Mar, 23				Apr, 23					May, 23		
			26	05	12	19	26	02	09	16	23	30	07	14
⋮	17	Iteration 3												
⋮	18	Planning												
⋮	19	Documentation												
⋮	20	Research												
⋮	21	Implementation												
⋮	22	Testing												
⋮	23	Bug Fixing												
⋮	24	Deployment												



1.10 Hardware and Software Specification

Client-Side Hardware Requirements:

Laptop or Desktop or Mobile Device

Client-Side Software Requirements:

Browsers that can support HTML 5

1.11 Tools and technologies used with reasoning

Tools	Reason
VS Code	Used as IDE for front-end development. It is lightweight, extensible, free, open source and cross-platform
GitHub and Git	Used for team collaboration. Git is a version control system that lets you manage and keep track of the source history. GitHub is a cloud-based hosting service that lets you manage Git repositories.
Google Docs	Used for documentation. It helps to make professional-quality documents.

Technology	Reason
ReactJS	It's used for building interactive user interfaces and web applications quickly and efficiently with significantly less code than you would with vanilla JavaScript.
NodeJS	Node. js is commonly used to develop real-time applications, also known as RTAs. Its asynchronous, event-driven nature, allows it to handle heavy input-output operations, which makes it much easier for Node. js developers to achieve the level of performance users have come to expect from modern real-time applications.
ExpressJS	Express is a node js web application framework that provides broad features for building web and mobile applications. It is used to build a single page, multipage, and hybrid web application. It's a layer built on top of the Node js that helps manage servers and routes.
MongoDB	is an open-source NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, and store or retrieve information.

Faculty of Computing & Information Technology



Title: **PETx** Deliverable One

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: Nov 21, 2022

TABLE OF CONTENTS

Introduction	4
Project/Product Feasibility Report	4
Technical Feasibility	4
Operational Feasibility	4
Economic Feasibility	5
Schedule Feasibility	5
Specification Feasibility	5
Information Feasibility	5
Motivational Feasibility	5
Legal & Ethical Feasibility	6
Project/Product Scope	6
Project/Product Costing	6
Project Cost Estimation By Function Point Analysis	6
Project Cost Estimation by using COCOMO'81 (Constructive Cost Model)	7
CPM - Critical Path Method	7
Gantt chart	9
Iteration One	10
Iteration Two	11
Iteration Three	11
Iteration Four	12
Introduction to Team members and their skill set	12
Tools and Technology with reasoning	13
Vision	14
-- In the loving memory of Laika	14
Elaborated View	14
Risk List	15
Systems Specifications	15
Introduction	15
Existing System	15
Phases of the System Development	16
Phase I	16
Phase II	16

Phase III	16
Organizational Chart	16
Summary of Requirements (Initial Requirements)	17
Rescue System	17
Veterinary Doctor Appointment System	17
Social Feed	18
Identifying External Entities and Actors	18
Context Level Data Flow Diagram	18
Capture "shall" Statements	19
Allocate Requirements	19
Prioritize Requirements	20
Requirements Trace-ability Matrix	21
High Level Use Case Diagram	22
Business Level	22
Analysis Level	24

Introduction

PETx will be all in one platform for services related to pets. We aim to facilitate pet owners by creating an online veterinary appointment system, rescue system, and social community specifically designed for pet owners. Apart from that, it will serve as a social awareness platform about pets for the general public and application visitors.

Project/Product Feasibility Report

To make PETx, we have commands on MongoDB, ReactJS, NodeJS, and ExpressJS to design and function websites while also having a database connected to them. For this project we will use a desktop having either Windows or Linux, CPU 2.0 GHz or more, RAM 8GB or more, having tools such as VS Code, Git, GitHub, etc. Our team members are capable of doing frontend, and backend as well as connecting databases, thus the “PETx” can be developed within a given time.

There are many types of feasibilities:

- Technical
- Operational
- Economic
- Schedule
- Specification
- Information
- Motivational
- Legal and Ethical

Technical Feasibility

This project requires command on development of frontend, backend, and database. For this purpose, we will use MongoDB, ReactJS, NodeJS, and ExpressJS. Our team does possess the ability to create this project, thus the project can be developed. For development purposes, Windows or LINUX, CPU 2.0 GHz or more, and RAM 8GB will be needed which will already have, so the development of this project is possible within the given time.

Operational Feasibility

With respect to the operational feasibility of the project, we aim to use the system for the following purposes. It is believed that they are important for the user base.

- Rapid Emergency Services for Pets
- Easy approach to veterinary doctors for pet owners
- Social interaction for Pet owners
- More societal awareness about Pets

→ Feel of satisfaction upon using the services of PETx

Development of this project will benefit many people who have pets yet are unable to find a one-stop shop for their pets. Upon sharing this idea with people around us having pets we get a positive response thus we think this project has a high probability of success.

Economic Feasibility

The project cost is not very high. The required softwares is either open source or we are using student subscriptions. For the hardware costs, we already have the required and recommended machines for development. Irrespective of the financial benefits we gain from the project, input resources will be low and hence it is a win-win situation in either case.

Schedule Feasibility

For schedule feasibility, we have a dedicated team of four members ready to work 15 hours per week for this project. So, we believe that the project milestones will be achieved within the given time frame.

Specification Feasibility

Requirements for this project have been analyzed, refined, and specified with mutual understanding of group members (stakeholders of the project). So, requirements are all set for the next phase which is implementation. The project scope has also been discussed and finalized. Some of the requirements have been labeled as options for the first year of the project.

Information Feasibility

All the information regarding the pet will be uploaded by the pet owner which will have a verified account. Furthermore, if an admin thinks the information uploaded by the user is false, he will be able to delete it. In the case of the newsfeed, the same policy of admin controls will be applicable. The non-admin users will be able to report content.

Motivational Feasibility

The team responsible for the project is ready and energized to start working on the project. We as a team are on the same and with the same mindset to create an impact in the community and that is why we have started this project.

Legal & Ethical Feasibility

Information and data hiding of stakeholders (patients, doctors, and volunteers) is solely the responsibility of the development team and we are taking care of using various techniques. That is the only legal responsibility and we are bound to secure the system from illegal usage.

Project/Product Scope

The main aim of the project revolves around the rescue of pets in times of distress. This involves communication between doctors and pet owners. The product will also facilitate veterinary doctors to work from home instead of being bound to specific locations. The project will also cover online payments to doctors and storing the information of pets in blockchain-based system

Project scope has been defined, validated, and specified. With the passage of time, we aim to extend the scope or descope in the coming iterations and sprints.

Project/Product Costing

Project Cost Estimation By Function Point Analysis

Function Point Metric Calculated using:

<https://w3.cs.jmu.edu/bernstdh/web/common/webapps/oop/fpcalculator/FunctionPointCalculator.html>

Function Point Calculator					Total	Factor	FP		
The Madison Utilities, Department of Computer Science, James Madison University					311	0.96	299		
Direct Measure	Simple	Count	Average	Complex	Weighted Measure				
External Inputs (EIs)	4	4	2		40				
External Outputs (EOs)	2	5	3		54				
External Inquiries (EQs)	2	1	2		22				
Internal Logical Files (ILFs)	7	3	4		139				
External Interface Files (EIFs)	3	3	2		56				
Clear									
Value Adjustment Factor					3	5	2	4	2
The system requires reliable backup and recovery.					<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
Specialized data communications are required.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
There are distributed processing functions.					<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
Performance is critical.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
The system runs in an existing, heavily utilized operational environment.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
The system requires on-line data entry.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
The on-line data entry requires transactions over multiple screens/operations.					<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
ILFs are updated on-line.					<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
The inputs, outputs, files or inquiries are complex.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
The internal processing is complex.					<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>
The code is designed to be reusable.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
Conversions/installation are included in the design.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
The system is designed for multiple installations in different organizations.					<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
The system is designed to facilitate change and ease of use.					<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>

Calculate Function Point:

FP est. = 299 p-m

For our project:**Assuming:**

1 month = 22 working days, per person

Labour Rate = 30000 Rs/pm (per day: 1363.63 Rs)

Average productivity = 15 FP/month (per day: 0.58 FP)

Cost per FP = Cost/FP = Labor Rate/Productivity

Cost/FP = Labour rate / productivity parameter

Cost/FP = 1363.63/0.58

Cost/FP = 2350 Rs/FP

Total Project Cost = FP estimation * (Cost/FP)

Total Project Cost = 299 * 2350

Total Project Cost = 702,650 Rs

Effort = Total cost/Labor rate per p-m**Total estimation effort = FP estimation / productivity parameter**

Total estimation effort = 299/15

Total estimation effort = 19.93 pm. (19.93 * 22 = 430 days)

Project Cost Estimation by using COCOMO'81 (Constructive Cost Model)

<https://strs.grc.nasa.gov/repository/forms/cocomo-calculation/>

<http://softwarecost.org/tools/COCOMO/>

COCOMO RESULTS for PETx								
MODE	"A" variable	"B" variable	"C" variable	"D" variable	KLOC	EFFORT, (in person-months)	DURATION, (in months)	STAFFING, (recommended)
semi-detached	9.692949569986077	1.12	2.5	0.35	5.000	58.790	10.404	5.651
<p>Explanation: The coefficients are set according to the project mode selected on the previous page, (as per Boehm). Note: the decimal separator is a period.</p> <p>The final estimates are determined in the following manner:</p> <p>effort = $a \cdot KLOC^b$, in person-months, with KLOC = lines of code, (in thousands), and:</p> <p>staffing = effort/duration</p> <p>where a has been adjusted by the factors:</p>								

CPM - Critical Path Method

[Access the calculator used for this calculation](#)

PERT Calculator

Online calculator to find the critical path method of project plans. Program Evaluation and Review Technique (PERT) and Critical Path Method (CPM) are the two useful techniques for planning a project.

Find Project Critical Path Method (CPM)

Number of Activities

Activity	Succes - sors	Predeces - sors	Duration	EST	EFT	LST	LFT	Slack Time
A	B		0	0	0	0	0	0
B	C	A	1	0	1	0	1	0
C	D	B	2	1	3	1	3	0
D	E	C	3	3	6	3	6	0
E	F	D	4	6	10	6	10	0
F	G	E	5	10	15	10	15	0
G	H	F	6	15	21	15	21	0
H		G	0	21	21	21	21	0

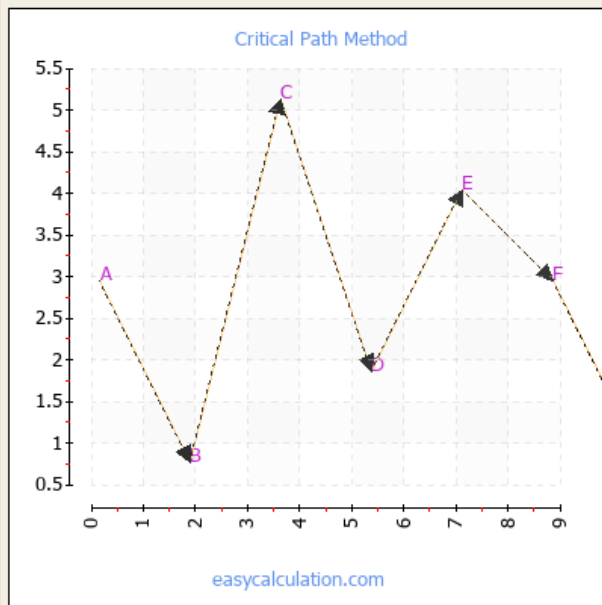
Calculate

Reset

Enter Single Character of the 8th activity field

Critical Path

A --> B --> C --> D --> E --> F --> G --> H



Gantt chart

ID	Name	Start Date	End Date	Duration
1	Iteration 1	Oct 31, 2022	Dec 31, 2022	62 days
2	Planning	Nov 01, 2022	Nov 14, 2022	14 days
3	Documentation	Nov 08, 2022	Nov 18, 2022	11 days
4	Research	Nov 16, 2022	Nov 26, 2022	11 days
5	Implementation	Nov 21, 2022	Dec 08, 2022	18 days
6	Testing	Dec 07, 2022	Dec 19, 2022	13 days
7	Bug Fixing	Dec 15, 2022	Dec 26, 2022	12 days
8	Deployment	Dec 26, 2022	Dec 30, 2022	5 days
9	Iteration 2	Jan 01, 2023	Feb 28, 2023	59 days
10	Planning	Jan 01, 2023	Jan 07, 2023	7 days
11	Documentation	Jan 08, 2023	Jan 14, 2023	7 days
12	Research	Jan 15, 2023	Jan 17, 2023	3 days
13	Implementation	Jan 16, 2023	Feb 10, 2023	26 days
14	Testing	Feb 12, 2023	Feb 15, 2023	4 days
15	Bug Fixing	Feb 16, 2023	Feb 20, 2023	5 days
16	Deployment	Feb 19, 2023	Feb 22, 2023	4 days
17	Iteration 3	Mar 01, 2023	Apr 30, 2023	61 days
18	Planning	Mar 01, 2023	Mar 08, 2023	8 days
19	Documentation	Mar 08, 2023	Mar 15, 2023	8 days
20	Research	Mar 15, 2023	Mar 18, 2023	4 days
21	Implementation	Mar 19, 2023	Apr 15, 2023	28 days
22	Testing	Apr 16, 2023	Apr 20, 2023	5 days
23	Bug Fixing	Apr 20, 2023	Apr 22, 2023	3 days
24	Deployment	Apr 21, 2023	Apr 27, 2023	7 days
25	Iteration 4	May 01, 2023	Jun 30, 2023	61 days
26	Planning	May 01, 2023	May 06, 2023	6 days
27	Documentation	May 05, 2023	May 08, 2023	4 days
28	Research	May 08, 2023	May 13, 2023	6 days
29	Implementation	May 18, 2023	Jun 17, 2023	31 days
30	Testing	Jun 16, 2023	Jun 19, 2023	4 days
31	Bug Fixing	Jun 18, 2023	Jun 21, 2023	4 days
32	Deployment	Jun 22, 2023	Jun 24, 2023	3 days

Iteration One

	ID ↑ :	Name	Nov, 22				Dec, 22				Jan, 23	
			30	06	13	20	27	04	11	18	25	01
⋮	1	Iteration 1										
⋮	2	Planning										
⋮	3	Documentation										
⋮	4	Research										
⋮	5	Implementation										
⋮	6	Testing										
⋮	7	Bug Fixing										
⋮	8	Deployment										

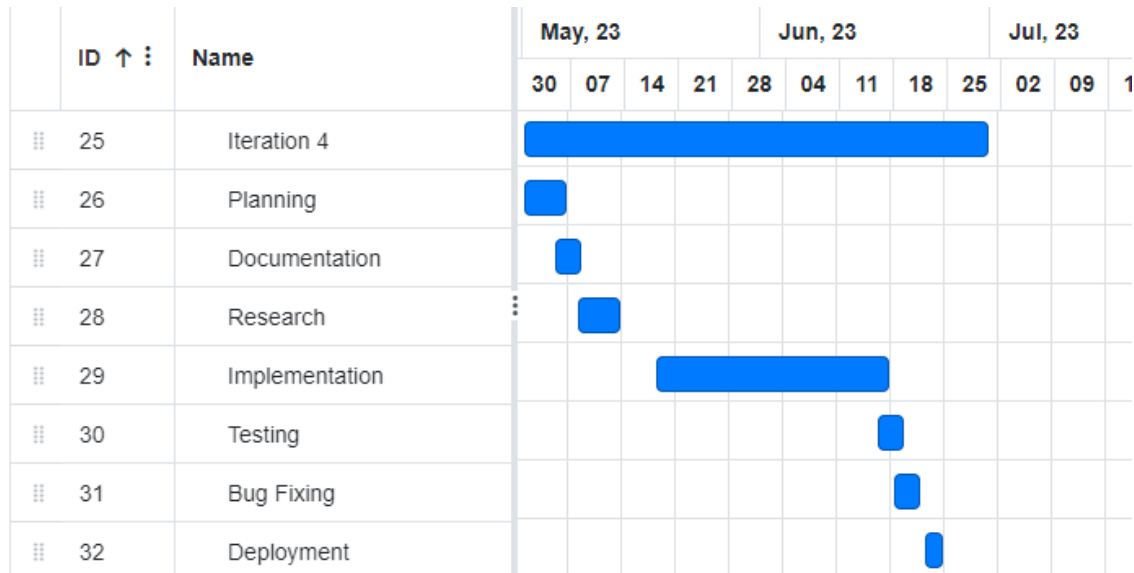
Iteration Two

	ID ↑ :	Name	Jan, 23				Feb, 23				Mar, 23		
			01	08	15	22	29	05	12	19	26	05	12
⋮	9	Iteration 2											
⋮	10	Planning											
⋮	11	Documentation											
⋮	12	Research											
⋮	13	Implementation											
⋮	14	Testing											
⋮	15	Bug Fixing											
⋮	16	Deployment											

Iteration Three

	ID ↑ :	Name	Mar, 23				Apr, 23					May, 23		
			26	05	12	19	26	02	09	16	23	30	07	14
⋮	17	Iteration 3												
⋮	18	Planning												
⋮	19	Documentation												
⋮	20	Research												
⋮	21	Implementation												
⋮	22	Testing												
⋮	23	Bug Fixing												
⋮	24	Deployment												

Iteration Four



Introduction to Team members and their skill set

Serial No.	Name	Expertise	Responsibilities
1	ATA UL MOHSIN	ReactJS, Quality Assurance	→ Frontend Development of PETx → Quality Assurance of the product
2	HASSAN AHMAD SARFRAZ	MongoDB, Express	→ Backend development of PETx → APIs creation and usage
3	MUHAMMAD SAAD	ReactJS, Blockchain	→ Research on blockchain for PETx → Frontend development
4	MUHAMMAD SALEH BUTT	NodeJS, MongoDB	→ Backend development → Database handling

Tools and Technology with reasoning

Tools	Reason
VS Code	Used as IDE for front-end development. It is lightweight, extensible, free, open source and cross-platform
GitHub and Git	Used for team collaboration. Git is a version control system that lets you manage and keep track of the source history. GitHub is a cloud-based hosting service that lets you manage Git repositories.
Google Docs	Used for documentation. It helps to make professional-quality documents.

Technology	Reason
ReactJS	It's used for building interactive user interfaces and web applications quickly and efficiently with significantly less code than you would with vanilla JavaScript.
NodeJS	Node. js is commonly used to develop real-time applications, also known as RTAs. Its asynchronous, event-driven nature, allows it to handle heavy input-output operations, which makes it much easier for Node. js developers to achieve the level of performance users have come to expect from modern real-time applications.
ExpressJS	Express is a node js web application framework that provides broad features for building web and mobile applications. It is used to build a single page, multipage, and hybrid web application. It's a layer built on top of the Node js that helps manage servers and routes.
MongoDB	is an open-source NoSQL database management program. NoSQL is used as an alternative to traditional relational databases. NoSQL databases are quite useful for working with large sets of distributed data. MongoDB is a tool that can manage document-oriented information, and store or retrieve information.

Vision

-- In the loving memory of Laika

An online portal where users can call rescue teams for pets or birds in times of distress and those team's manager/administrator can send the nearest team to spot for help. This portal will also serve as a base for online appointments with pets and veterinary doctors.

Inspiration for this project is from the first animal taken in earth orbit, a female dog named Laika. While returning back to earth, the spacecraft malfunctioned and she was dead in a few moments. Would it have been much better if some kind of rescue or ejection system was placed there? Let's aim to help them at least in our capacity (nearby areas).

Elaborated View

→ Introduction

We want to design a system or marketplace to be more precise in which we can gather all the important and major services that a pet owner usually needs. We want to develop a platform where the pet owner need not worry about most things associated with pets. This marketplace could make the life of a pet owner easier.

→ Positioning

Individual services related to pets may be available out there, but we lack a system where one can get everything almost at arm's length. Such a system does not exist with such a unique idea that makes pet keeping a lot easier and more convenient. So we are almost certain that we are going to make our own benchmark in the market and will serve the community as well.

→ Stakeholder description

Our stakeholders include our sponsors that are from other domains but looking forward to working with us for their advertisement and branding. The chief investor hit upon the idea and decided to make his dream come true. Also the companies from our domain that want to cooperate with us and provide us with their best services to entertain our end users.

→ High-level goals

A marketplace where we can earn both good reputé and profit by serving the pets community. Our goal is to hit the highest rank in the market by developing a system in which a pet owner can get easy access to all the things he needs for the best growth of his pet.

→ User-level goals

The idea is to provide an emergency rescue team for animals in general that are trapped in a dangerous situation so that their lives may get saved.

We also want to make pet keeping hassle-free by providing our vet care services that include a general appointment with doc for consultation or an appointment with doc for some specific medical condition.

And last but not least, we will be providing our users a common place where similar pet owners could interact with each other to have guidelines from experienced ones or just share their thoughts with others.

→ Product overview

A full-fledged product that will take away all the tension from the mind of pet owners by providing assistance at every step he wants. We will make pet keeping more fun and peace of mind than ever.

→ Summary of features

Users can call the rescue team for emergencies. Users can consult a doctor on general concerns or can have an appointment with a specialist for having a specific issue / disease-oriented discourse. A section of the social community will also be added for the interaction of pet owners with other owners.

Risk List

Listed from highest to lowest risk

1 means the highest risk and 6 means the lowest risk.

1. Customers/Users may change their functional/ elemental requirements.
2. Performance may not be according to the needs of the client.
3. Security and privacy may not be enough for clients.
4. Project costs may increase later.
5. We may have to modify throughout the project resulting in development side loss.
6. Development team may not meet the deadline for the project.

Systems Specifications

Introduction

PETx is an emergency medical service for pets which deals with basic medical prescriptions from veterinary doctors to providing volunteer service to pets in emergency situations. In Pakistan from urban to rural areas, pets and cattle are found in every town. However, the number of which is diminishing day by day due to unavailability of doctors to treat pets and lack of awareness about them.

Existing System

- PETx system mainly deals with the following aspects:
 - ◆ Basic Medical Prescription for pets
 - ◆ Emergency Rescue service
 - ◆ NewsFeed for pet owners
- *Basic Medical Prescription for pets*
 - ◆ Deals with prescribing basic medicine to pets
- Emergency Rescue service
 - ◆ Deals with providing emergency rescue services to pets on voluntary basis
- NewsFeed for pet owners
 - ◆ To share stories about pets Facebook-like news feed will be maintained in system

Phases of the System Development

The PETx System is divided into three phases.

Phase I

Phase I includes the following basic architecture areas:

- Landing pages
- Simplified UI
- Database Setup

Phase II

Phase II involves improvement of user experience. Phase II includes the following areas:

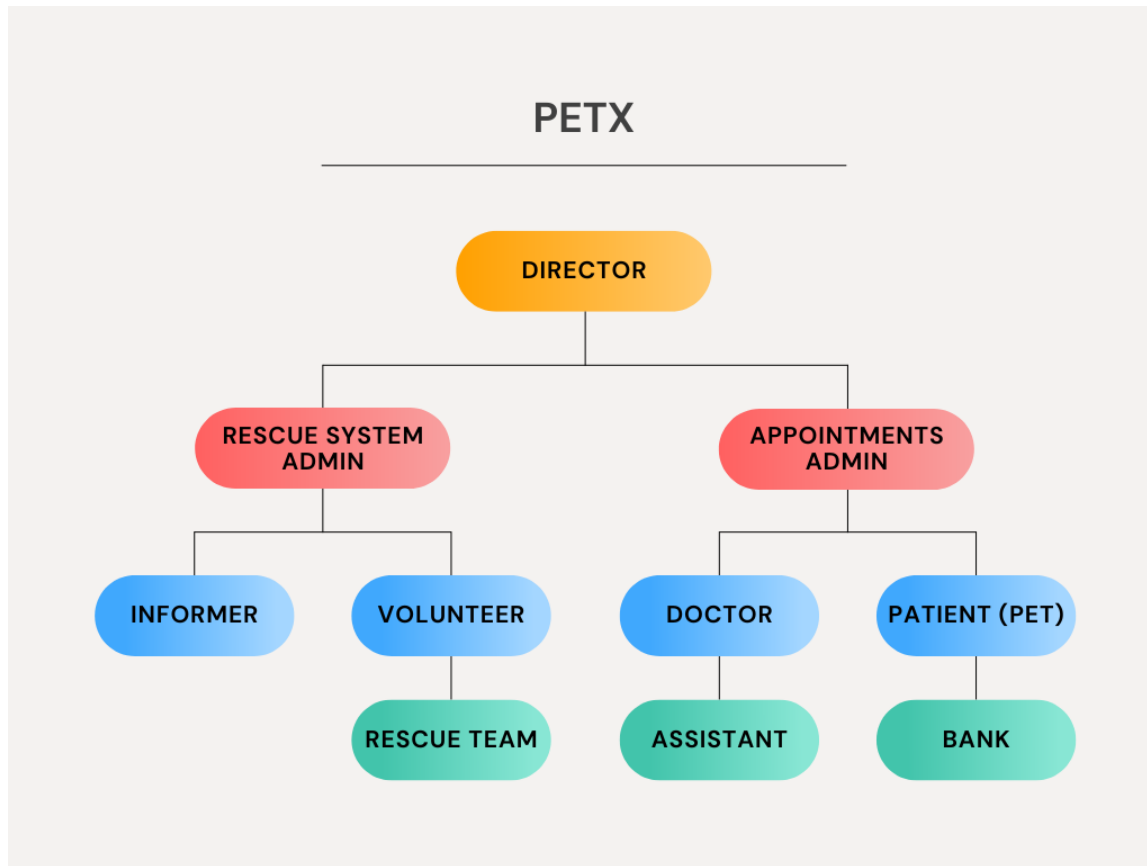
- Customized Roles
- Improved UX and UI

Phase III

Phase III covers a complete solution for PETx. Phase III includes remaining business areas which are not developed in previous phases.

- Blockchain-based payment system
- Blockchain to be used for storing pet information

Organizational Chart



Summary of Requirements (Initial Requirements)

An abstract is necessary at this stage to give an understanding of the initial requirements of the system. This will show what high-level requirements the proposed system must address. This abstract will act as a foundation for the future analysis of the system.

Rescue System

Rescue system is the prime feature of our project. This module will require emergency situation information from the informer or the person who told the system about distress. This information should be in a form having location, contact number, and related information if any about the situation. Once an emergency event is initiated, it'll alert the volunteers and nearby rescue teams to move toward the location. Here only one team will be assigned the task of rescue and that team or volunteer will complete the job. If the task is not completed by a volunteer, a team will be assigned the job then.

Veterinary Doctor Appointment System

The logged-in customer or pet owner will visit the page of available doctors and will select a doctor according to his requirements. Then, he'll be asked to schedule an appointment from the

available time slots of a doctor. If he selects the time, the doctor will be notified about the scheduled appointment and then the person will move to the payment page which will be the banking site for the transaction.

From the doctor's side, he'll be able to see scheduled appointments from his side and will be able to change availability hours and fees.

Social Feed

This will serve as a community for pet owners. They'll be able to see posts of other users in their feed and like, and comment on them. These posts will be focused on pets only just like a Facebook group. Our first priority in this regard will be to integrate this module with Facebook APIs. Otherwise, we shall develop our own social media system.

Identifying External Entities and Actors

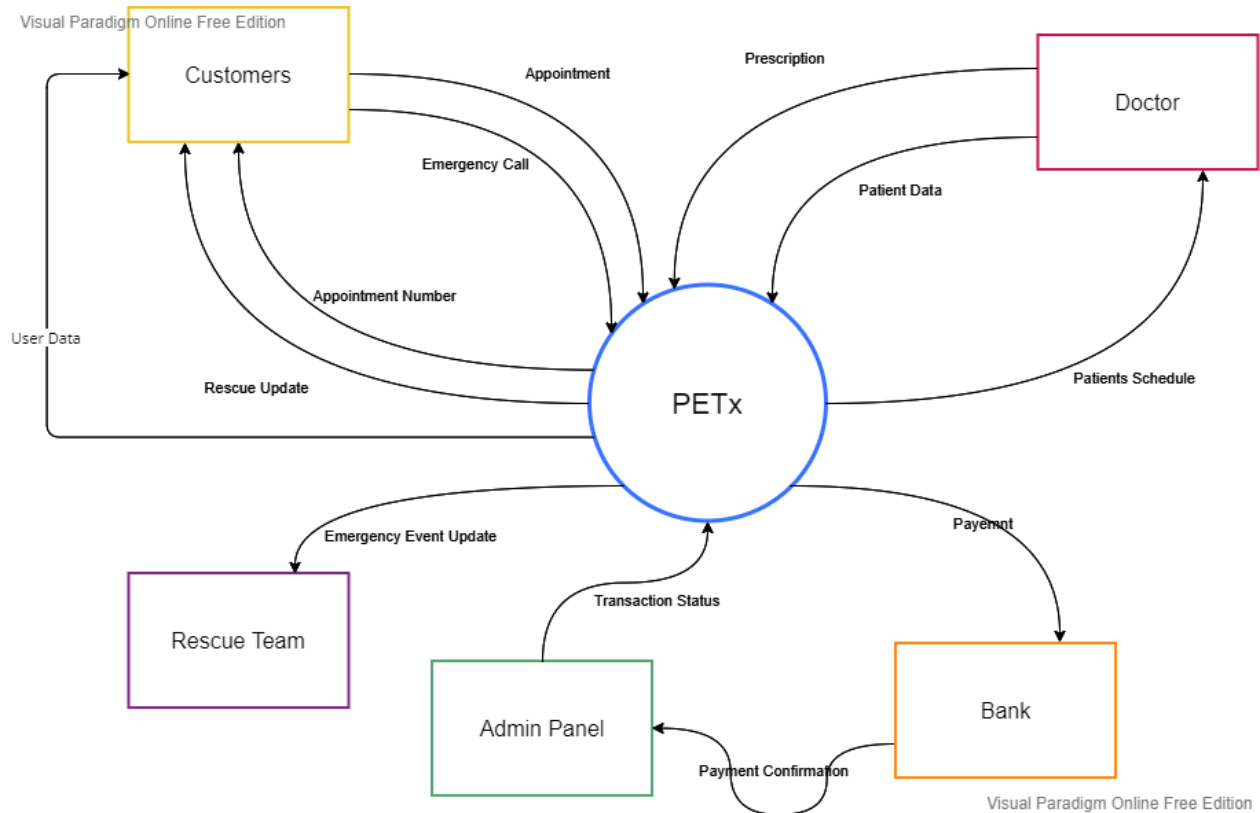
→ Primary Actors

- ◆ Veterinary Doctors
- ◆ Informer
- ◆ Customer

→ Secondary Actors

- ◆ Rescue Team
- ◆ Bank

Context Level Data Flow Diagram



Capture "shall" Statements

- Call for emergency pet rescue
 - ◆ The system shall be able to contact animal rescuers for help in times of emergency.
- Consultancy
 - ◆ The system shall be capable of online appointments with veterinary doctors.
 - ◆ The system shall be able to remind registered users of regular checkups of their pets via email or SMS.
- Interaction with similar pet owners
 - ◆ There shall be able to handle a community where registered users can interact with other users via posts or messages.
- FAQs/ Infopedia
 - ◆ The system will consist of an FAQ list for common questions.
- Membership or Special Discounts
 - ◆ The system shall be able to offer special discounts on different occasions.

Allocate Requirements

Serial No	Initial Requirements	Use Case Name
1	User will create account for Application	UC_Register
2	User will need to login to use Application	UC_Login
3	Users will be allowed to contact the system in case a pet faces an emergency.	UC_Help
4	User will know if there is any volunteer to help	UC_Volunteer
5	User will be able to upload the posts in NewsFeed	UC_Newsfeed
6	Users will be allowed to schedule a meeting with the Doctor.	UC_Meeting
7	Users will be able to make online payments.	UC_Billing
8	Users will also be able to get online prescriptions from Doctors.	UC_Medicine
9	Users will be able to report a NewsFeed Post	UC_ReportPost
10	Admin will be able to add a Doctor in a system.	UC_AddDoctor
11	Admin will be able to delete a post of User from NewsFeed.	UC_DeletePost
12	Admin will remove the rescue post if he thinks it is a fake	UC_FakeReport
13	Volunteers will be added to the system just like normal users.	UC_OnBoarding

Prioritize Requirements

Key: High: H, Medium: M, Low: L

Serial No.	Requirement	Priority
1	Rescue System	H
2	Onboarding of volunteers for rescue module	M
3	Onboarding of rescue organizations for rescue module	L

4	Accessibility of emergency information system	M
5	Online Appointment System	H
6	User shall create Account to use the App	H
7	User must be logged in to use the app	H
8	Appointment Scheduling and time management	M
9	User shall be able to post in Newsfeed	M
10	User will be able to report a post in NewsFeed	M
11	Admin will be able to delete a post	L
12	Admin will be able to register a doctor	H
13	User will be able to get online prescription from doctor	M

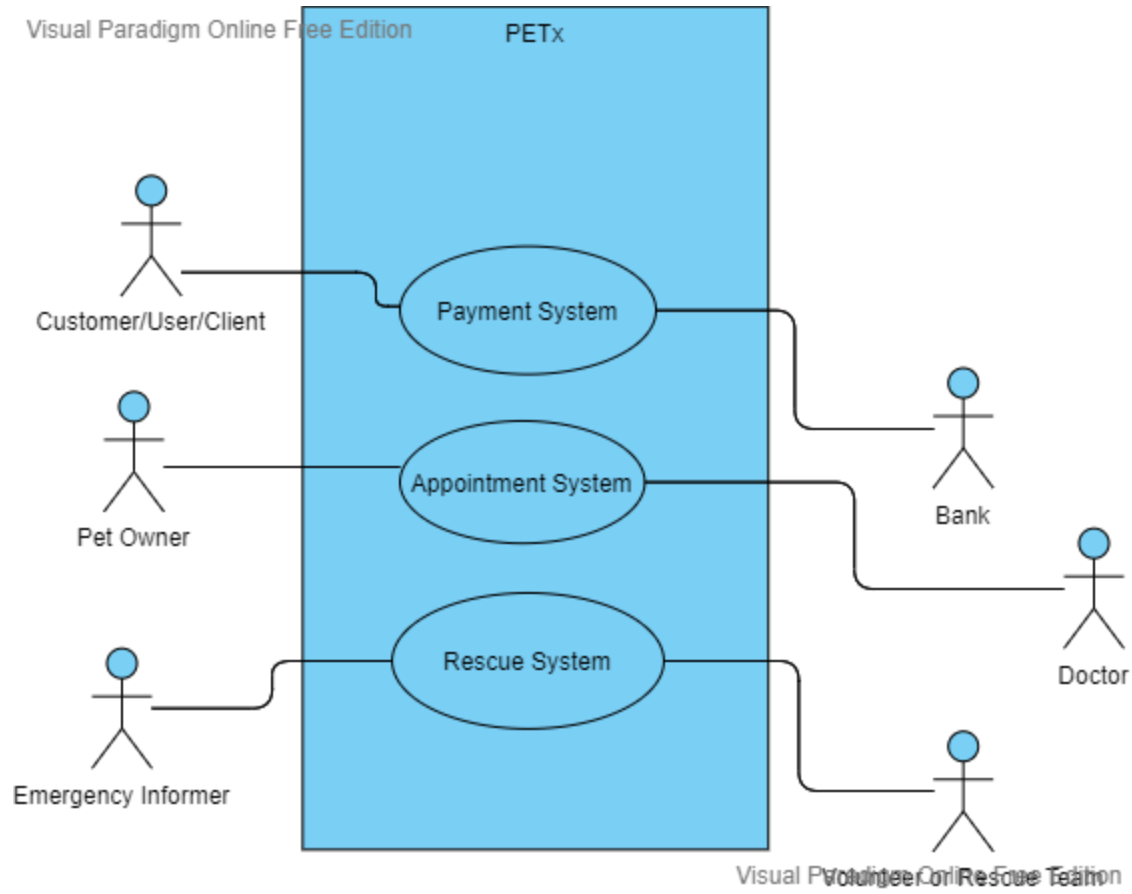
Requirements Trace-ability Matrix

Serial No.	System Specification Text	Use Case Name	Build	Category
1	User will create account for Application	UC_Register	1.0	Functional
2	User will need to login to use Application	UC_Login	1.0	Functional
3	Users will be allowed to contact the system in case a pet faces an emergency.	UC_Help	1.0	Business
4	User will know if there is any volunteer to help	UC_Volunteer	1.0	Functional
5	User will be able to upload the posts in NewsFeed	UC_Newsfeed	1.0	Functional
6	Users will be allowed to schedule a meeting with the Doctor.	UC_Meeting	2.0	Business

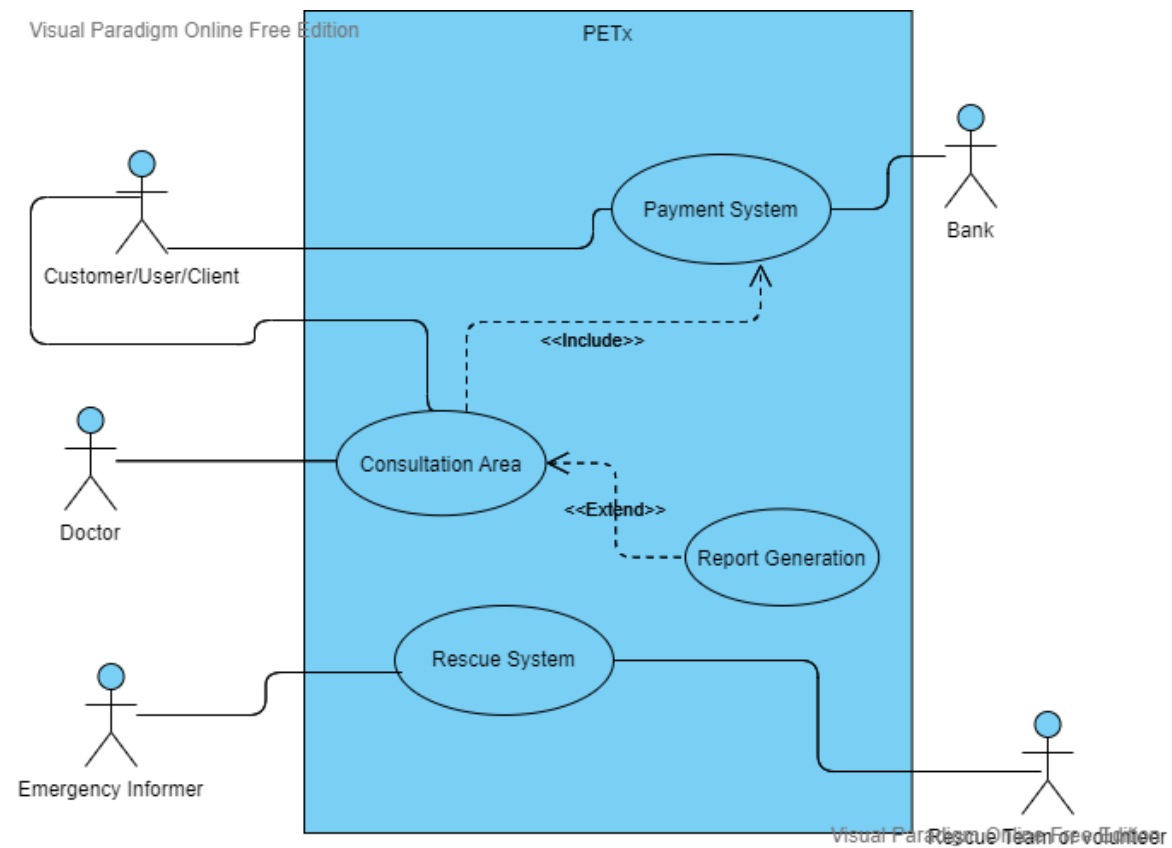
7	Users will be able to make online payments.	UC_Billing	1.0	Functional
8	Users will also be able to get online prescriptions from Doctors.	UC_Medicine	1.0	Business
9	Users will be able to report a NewsFeed Post	UC_ReportPost	2.0	Functional
10	Admin will be able to add a Doctor in a system.	UC_AddDoctor	1.0	Functional
11	Admin will be able to delete a post of User from NewsFeed.	UC_DeletePost	1.0	Functional
12	Admin will remove the rescue post if he thinks it is a fake	UC_FakeReport	2.0	Functional
13	Volunteers will be added to the system just like normal users.	UC_OnBoarding	1.0	Business

High Level Use Case Diagram

Business Level



Analysis Level



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

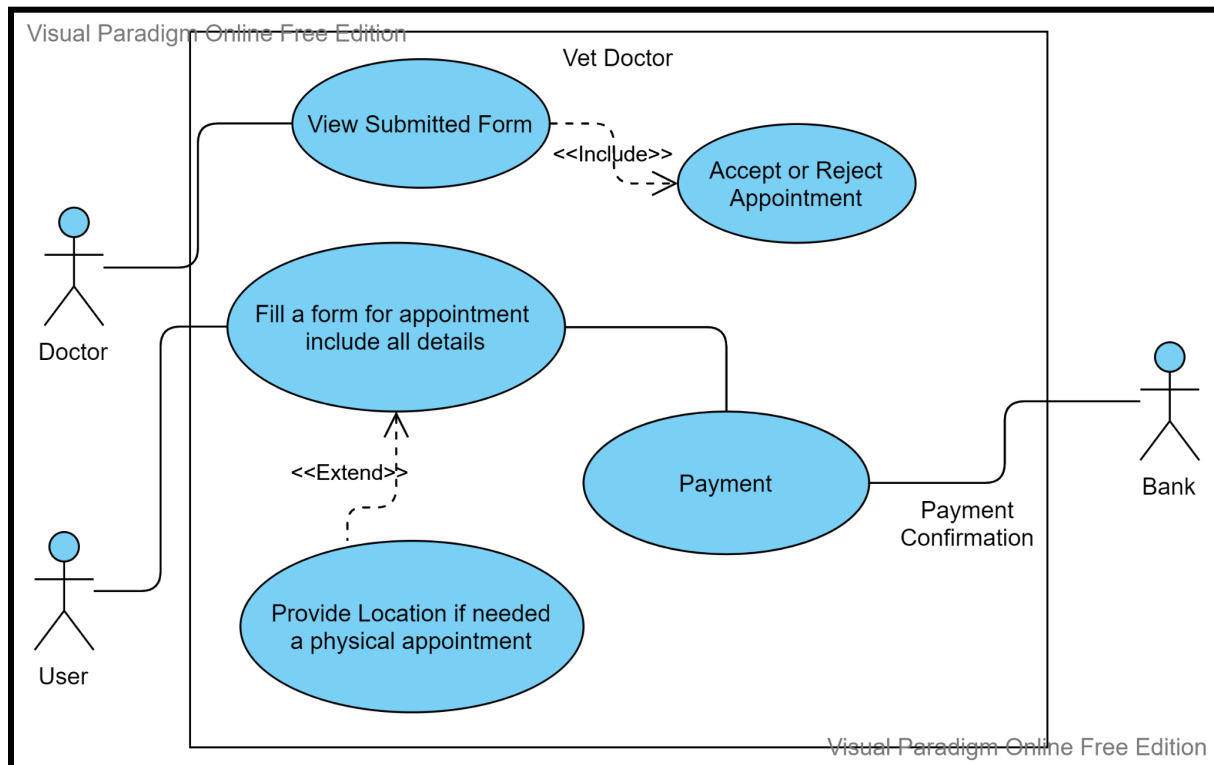
Table of Contents	2
Individual Use Cases	3
Vet Doctor	3
Rescue Team	3
Pets Community	4
Use Case Diagrams	5
Business Level Use Case Diagram	5
Analysis Level Use Case Diagram	7
Descriptive Use Cases	7
Domain Model	13
Sequence Diagram	14
System Sequence Diagrams	16
Operation / Use Case Contracts	17
Contract 01: Card verification (Credit / Debit)	17
Contract 02: Courier delivery	18
Contract 03: Appointment System	18
Contract 04: Blogging System	18
Contract 05: Crypto Payments	18
Design Class Diagram	19
Collaboration Diagram	21

Individual Use Cases

Vet Doctor

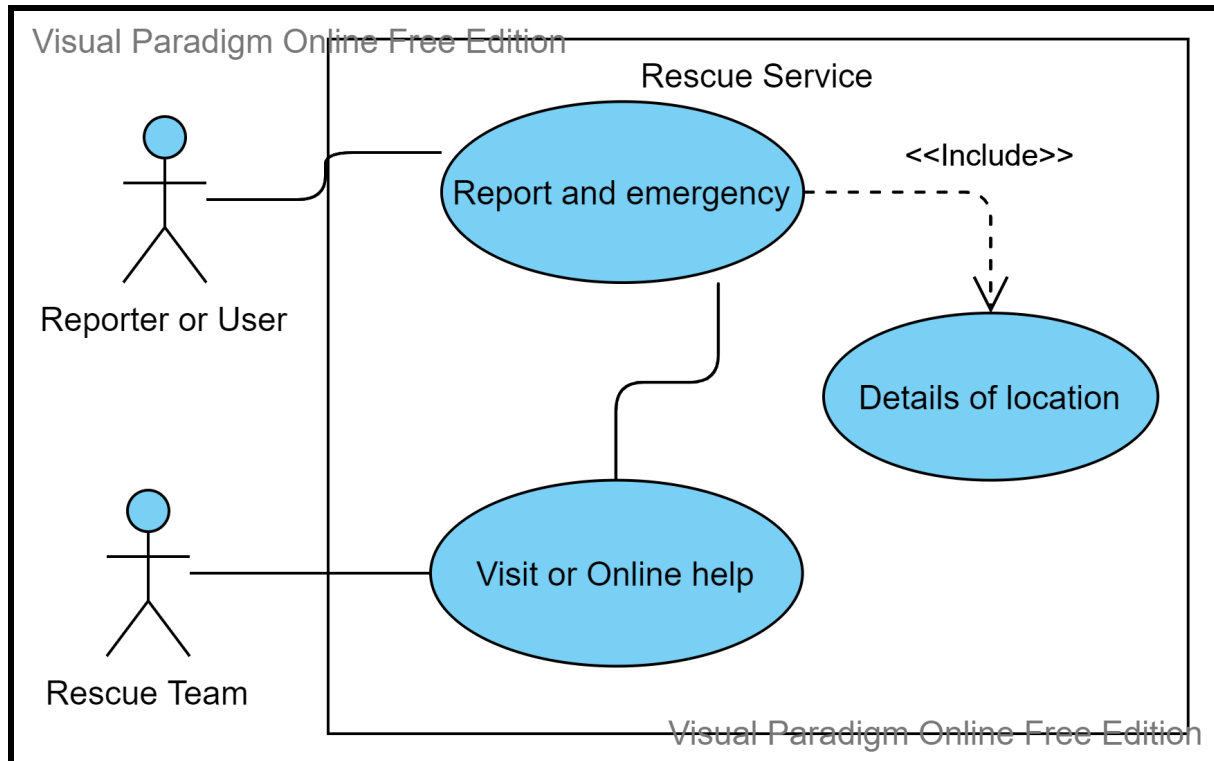
Goal: Patient pet owner logs into the system and fills out a form for an appointment and selects a schedule as well. Then he pays for appointments via credit card and checks out or exits the system.

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.

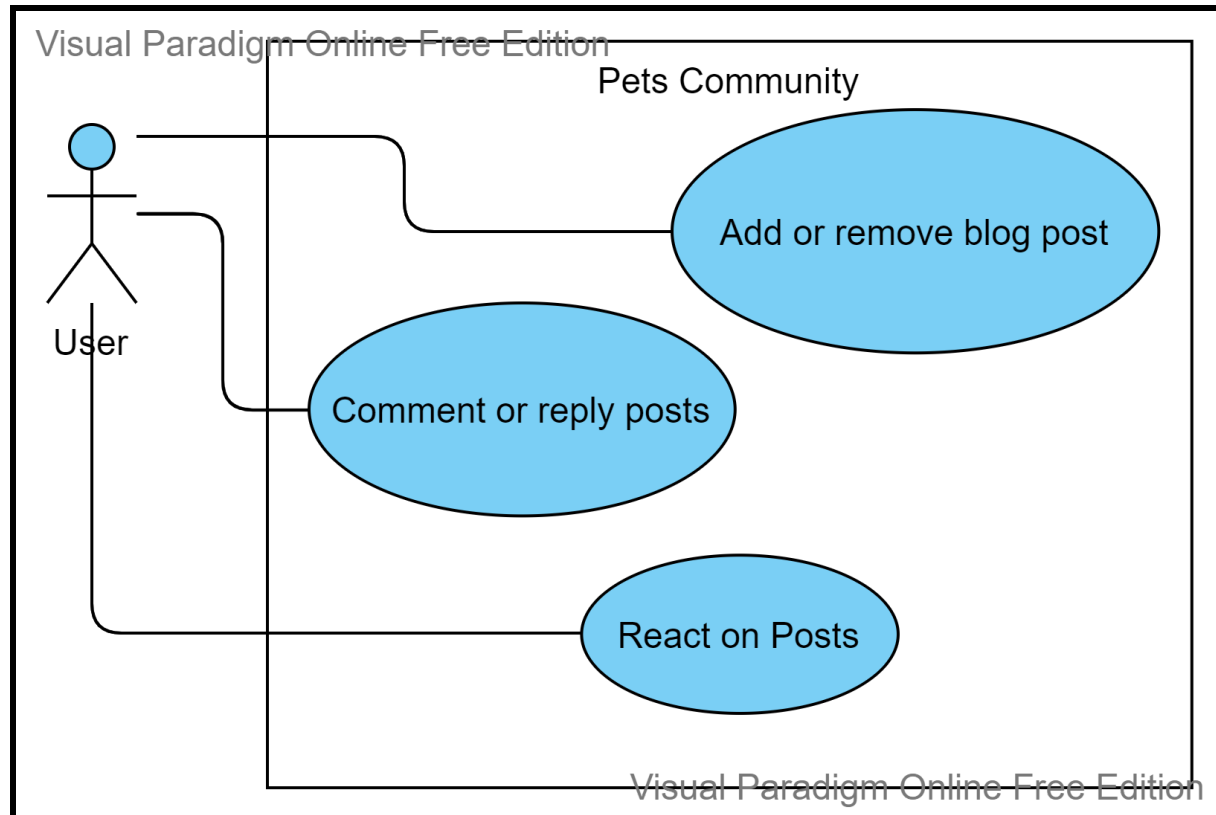


Pets Community

Goals: It is a place for social hang outs online. You may call it a blog for pet owners.

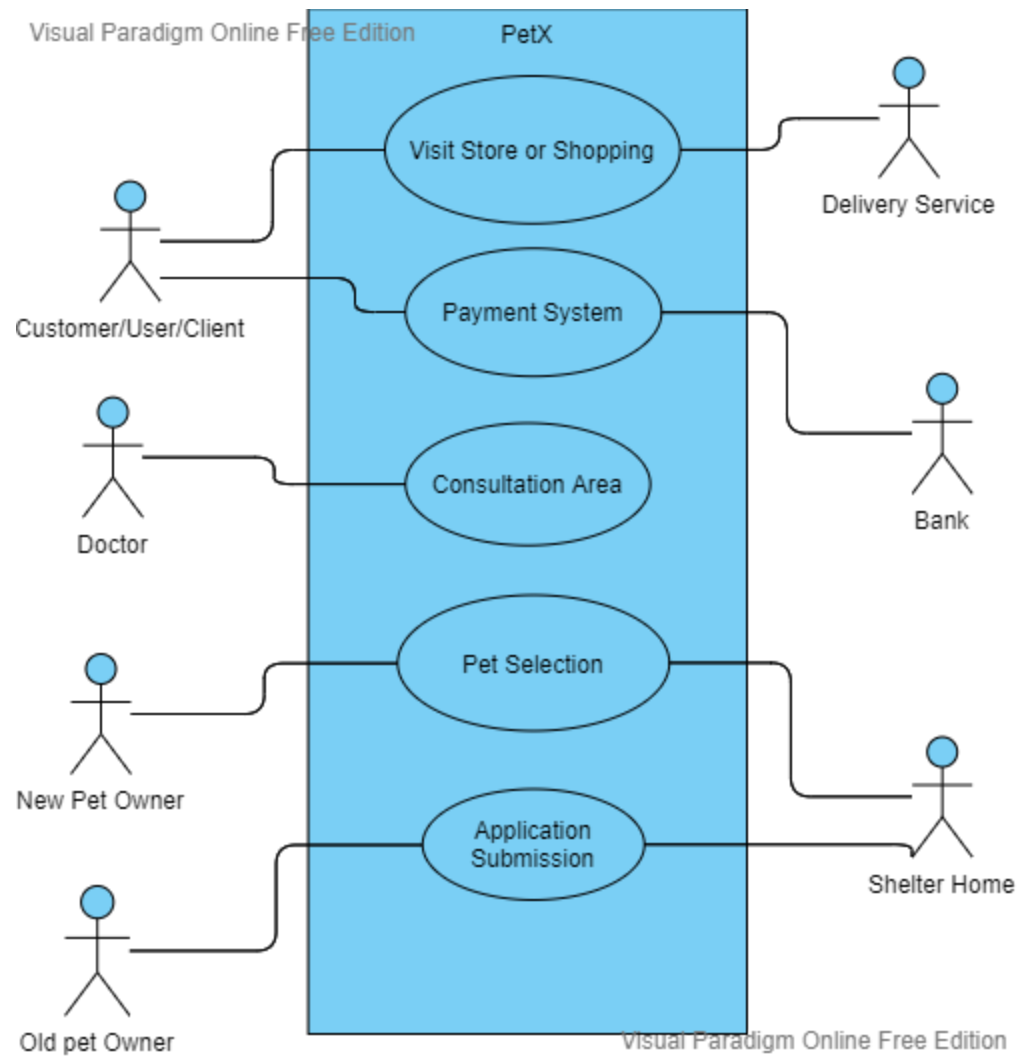
Here the pet owner can browse the feed where he can react or comment on the posts of others. He can also post for others.

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.

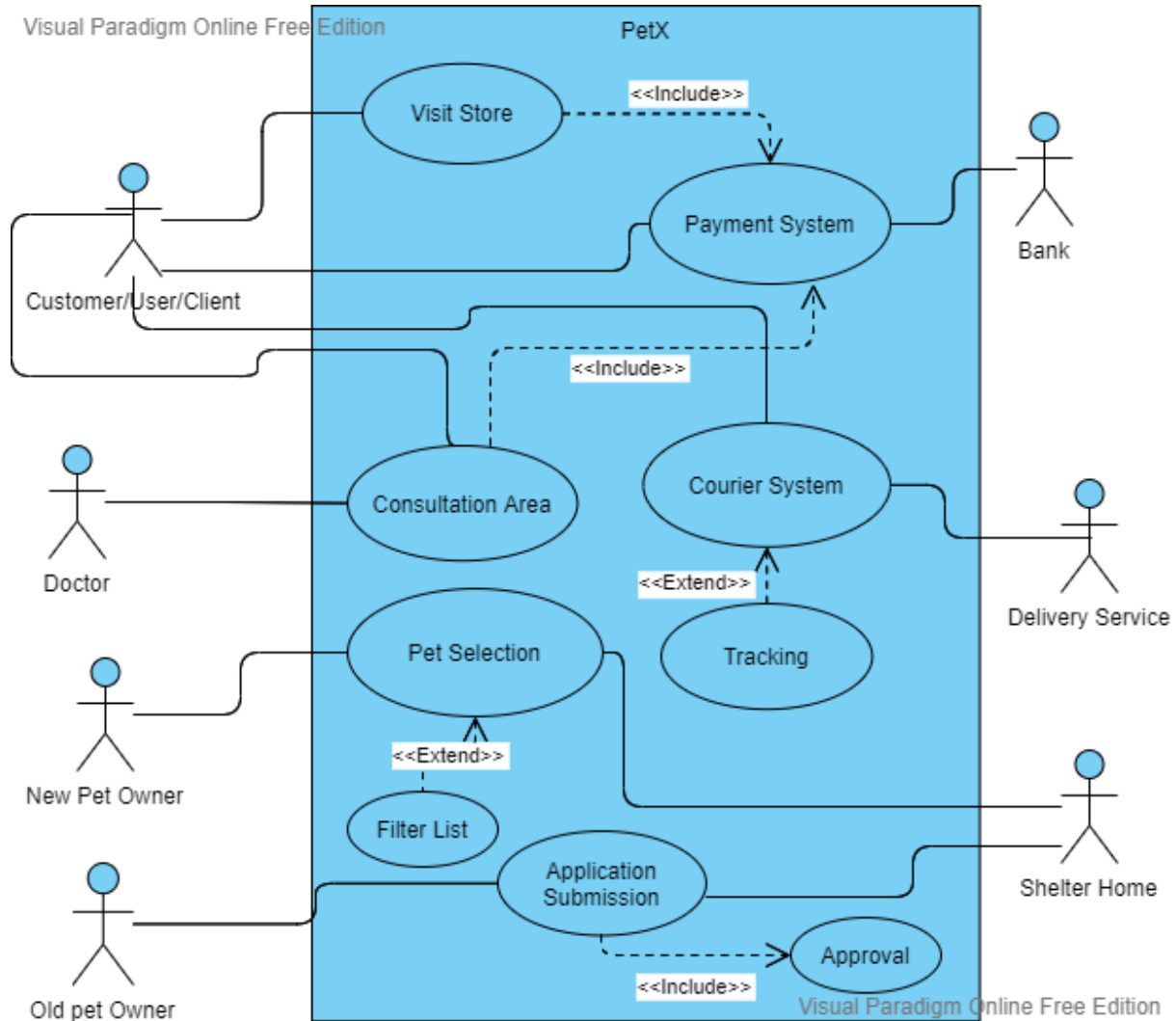


Use Case Diagrams

Business Level Use Case Diagram



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: True, 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.			

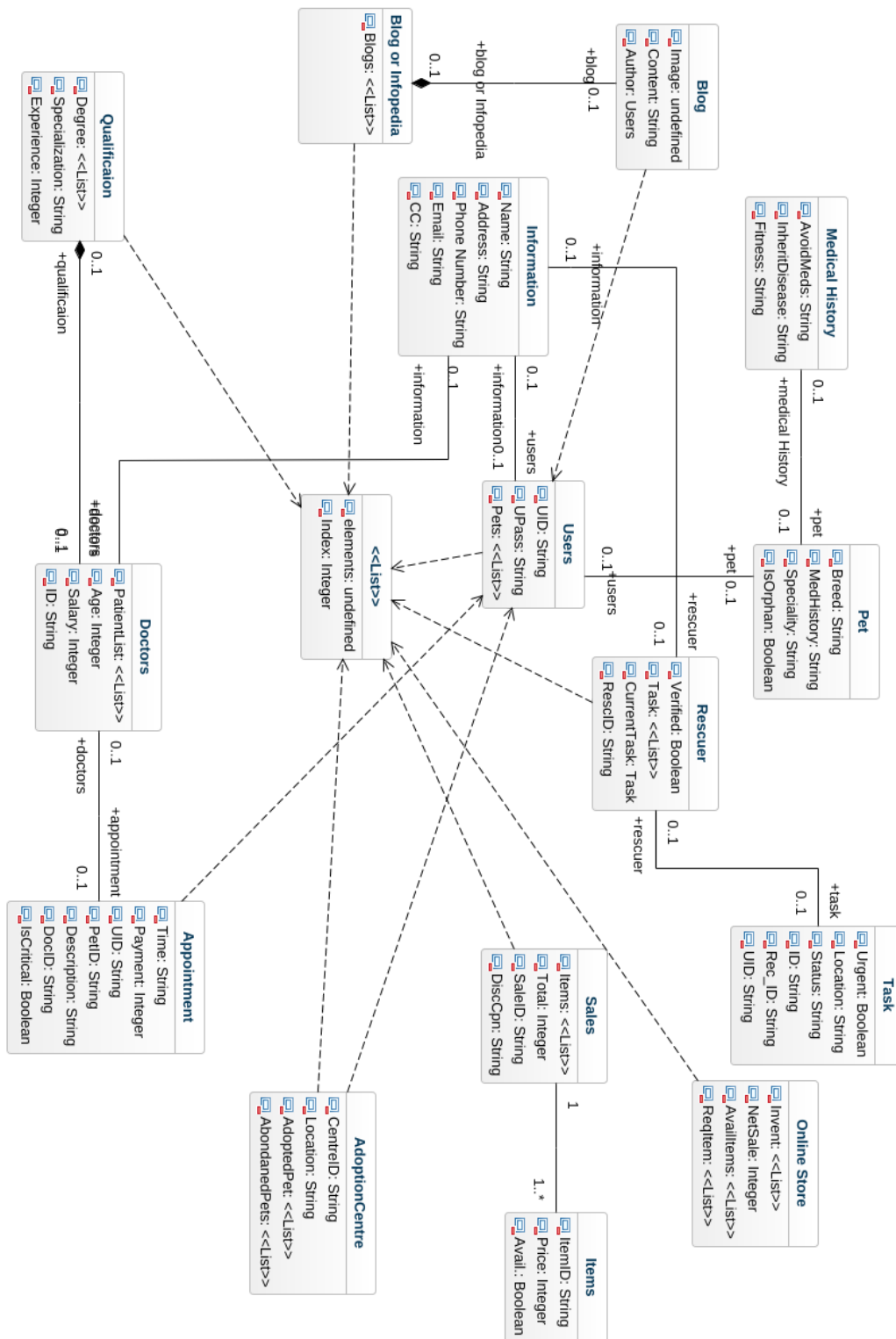
Actors	User, Rescue Team			
Pre Condition	isSent: False			
Post Condition	caseDescription: Not Null, pinLoc: Not Null, isSent: True, isTeamNotified: True, caseID: Not Null.			
Primary Flow	#	Actor	#	System
	1	User clicks the SOS button.	1	Emergency Form Form will be shown to the user.
	2	Describes the 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			
	#	Actor	#	System
Primary Flow	1	User can scroll through blogs and read	1	System will enlist all the published blogs
	#	Actor	#	System
Alternate Flow	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			

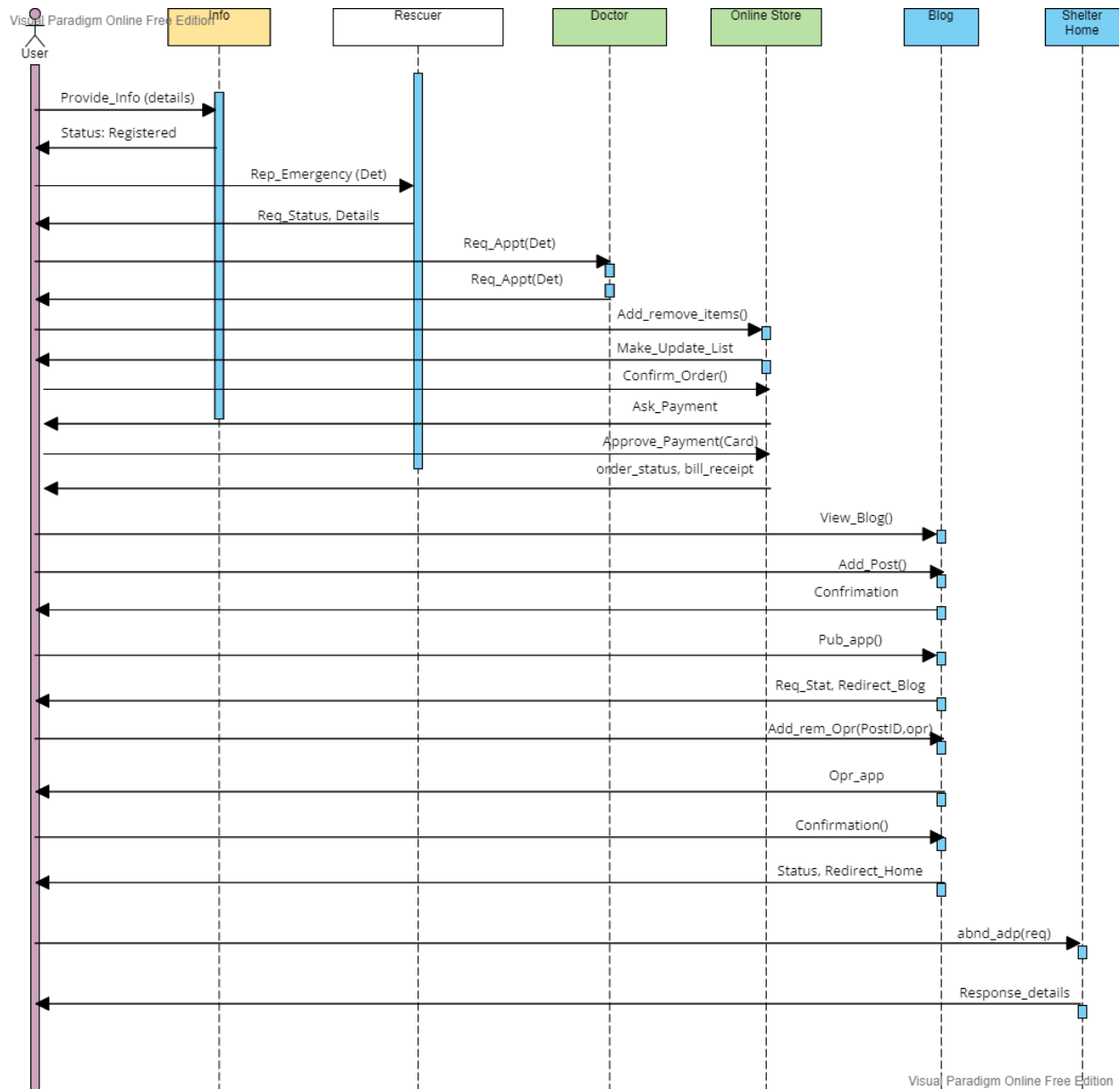
	#	Actor	#	System
Primary Flow	1	User will click the pen icon on the bottom.	1	A form will be displayed.
	2	Users will write blogs and may attach pictures as well.	2	System will check if input is invalid [3] or not [1(a)].
	3	Redirected to the blog reading page after publishing.	3	Blog is published and 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.

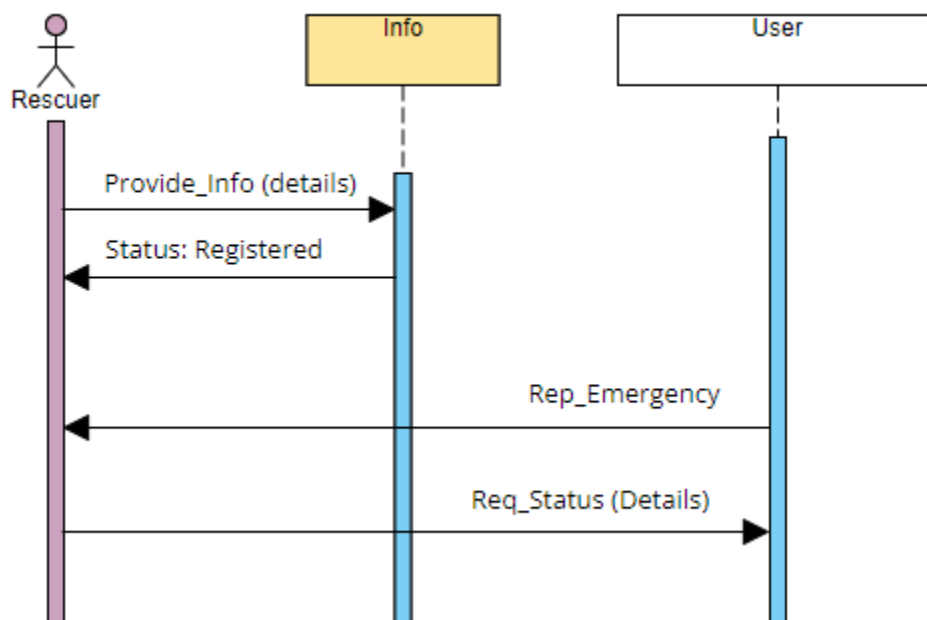
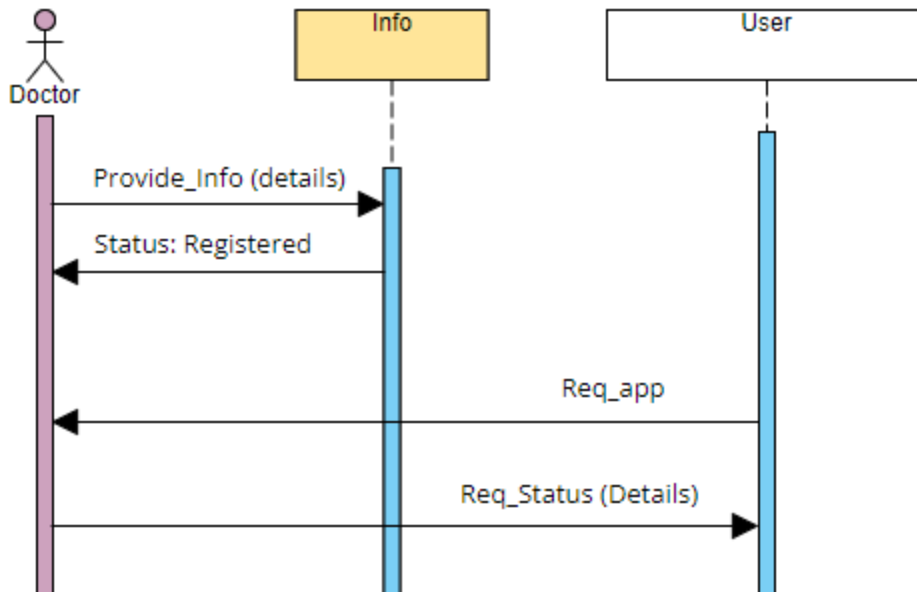
Checkout				
Description	User use cards to pay doctor fees and other purchases.			
Actors	User.			
Pre Condition	isPaymentMethodAdded: True, isCardVerified: True, trasacStatus: inProgress.			
Post Condition	transacStatus: Completed / Declined.			
Primary Flow	#	Actor	#	System
	1	User proceed to pay with 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

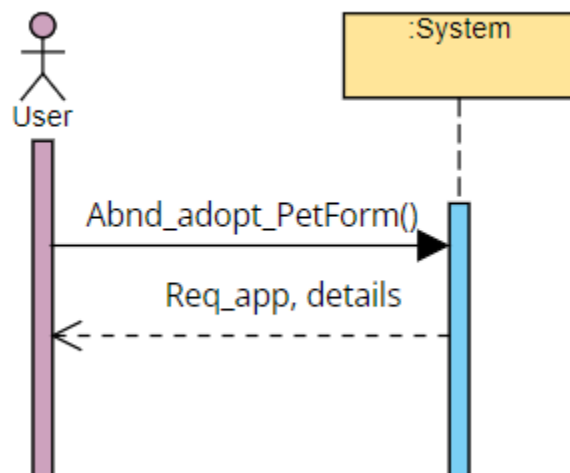
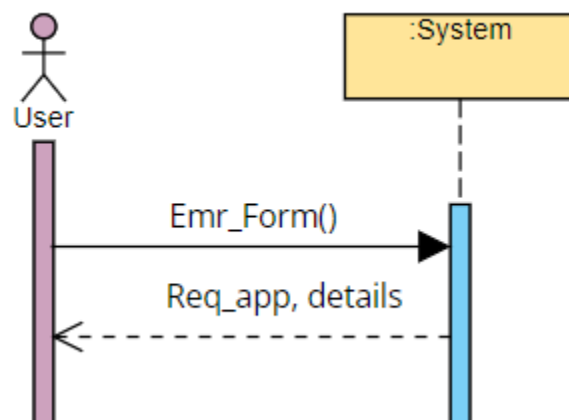


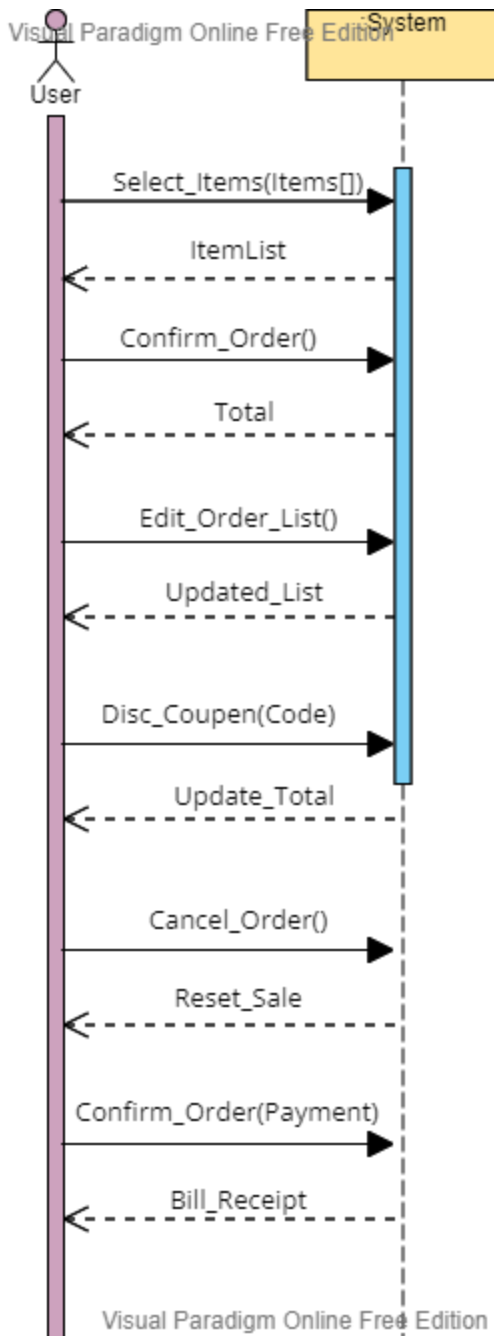
Sequence Diagram





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

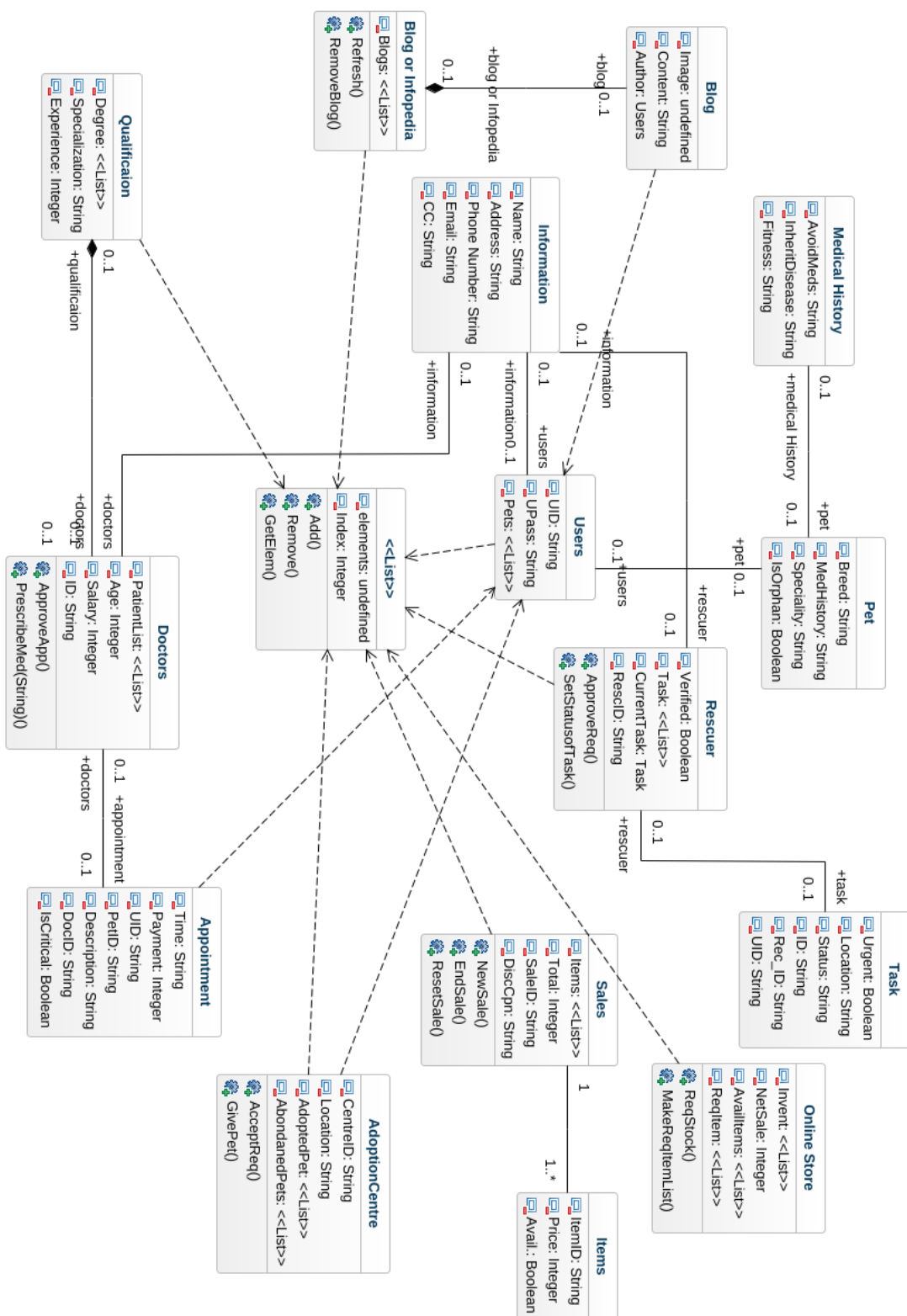
Operation: Crypto Payments

Cross-reference: Distributed Ledger System.

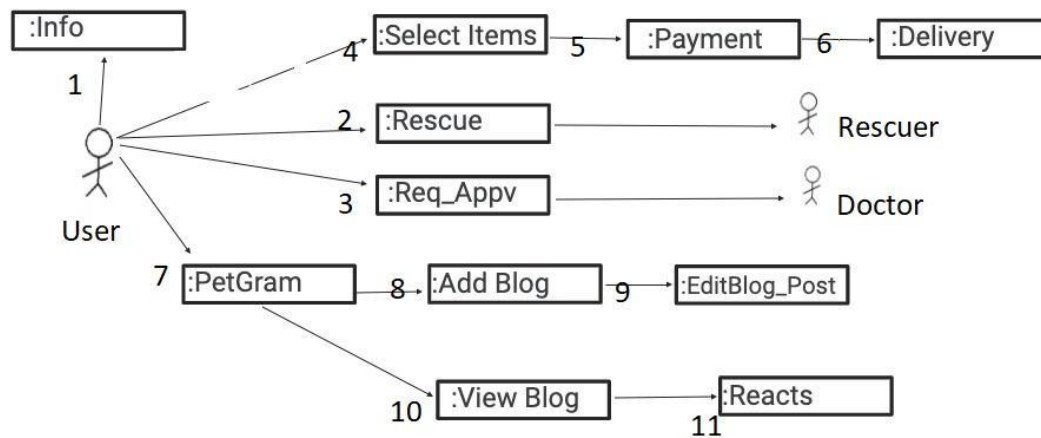
Pre-Condition: Wallet Integration: Successful, Amount: >0, Currency: Selected (from the list), Amount Available: True, Receiver Address: Not Null and Valid, Payment Network: Selected (ERC20, BSC, etc), Password Verification: Successful.

Pos-Condition: Transaction Status: Successful / Failed, Balance updated: True.

Design Class Diagram



Collaboration Diagram



Data Model

