[CHAPTER 1 INTRODUCTION 3](#_Toc29171)

[1. INTRODUCTION 4](#_Toc27226)

[1.1 System Introduction 4](#_Toc30388)

[1.2 Background of the System 4](#_Toc15850)

[1.3 Objectives of the System 5](#_Toc31915)

[1.4 Significance of the System 5](#_Toc14255)

[CHAPTER 2 6](#_Toc17272)

[REQUIREMENT SPECIFICATIONS 6](#_Toc16783)

[2 REQUIREMENT SPECIFICATIONS 7](#_Toc6951)

[2.1 Product Scope 7](#_Toc10803)

[2.2 Product Description 7](#_Toc9031)

[2.2.1 Product Perspective 7](#_Toc6226)

[2.2.2 Product Functionality 8](#_Toc24458)

[2.2.3 Users and Characteristics 9](#_Toc330)

[2.2.4 Operating Environment 9](#_Toc13952)

[2.3 Specific Requirements 10](#_Toc1387)

[2.3.1 Functional Requirements 10](#_Toc21303)

[2.3.2 Behavioral Requirements 11](#_Toc26523)

[2.3.3 External Interface Requirements 11](#_Toc26817)

[2.4 Non-functional Requirements 13](#_Toc17228)

[2.4.1 Performance Requirements 13](#_Toc3815)

[2.4.2 Safety and Security Requirements 14](#_Toc31439)

[2.4.3 Software Quality Attributes 15](#_Toc27039)

[CHAPTER 3 17](#_Toc27015)

[DESIGN SPECIFICATIONS 17](#_Toc6393)

[3 DESIGN SPECIFICATIONS 18](#_Toc149)

[3.1 Introduction 18](#_Toc7578)

[3.2 Composite Viewpoint 18](#_Toc27116)

[3.3 Logical Viewpoint 19](#_Toc32265)

[3.4 Information Viewpoint 20](#_Toc27882)

[3.5 Interaction Viewpoint 21](#_Toc6991)

[3.6 State Dynamics Viewpoint 22](#_Toc26865)

[3.7 Algorithmic Viewpoint 23](#_Toc22467)

[Explanation of Decision Table 24](#_Toc7575)

[CHAPTER 4 25](#_Toc2288)

[DEVELOPMENT AND TOOLS 25](#_Toc30040)

[4 DEVELOPMENT AND TOOLS 26](#_Toc13817)

[4.1 Introduction 26](#_Toc28411)

[4.2 Development Plan 26](#_Toc28146)

[4.3 Development Tools 27](#_Toc25355)

[4.4 Conclusion and Future Work/Extensions 27](#_Toc7668)

[CHAPTER 5 28](#_Toc3594)

[QUALITY ASSURANCE 28](#_Toc19717)

[4.5 28](#_Toc19787)

[5 QUALITY ASSURANCE 29](#_Toc5128)

[5.1 Introduction 29](#_Toc11158)

[5.2 Traceability Matrix 29](#_Toc6729)

[5.3 Test Plan 29](#_Toc1033)

[CHAPTER 6 32](#_Toc25451)

[USER MANUAL 32](#_Toc29902)

[6 USER MANUAL 33](#_Toc25023)

[6.1 Introduction 33](#_Toc26772)

[6.2 Hardware/Software Requirements for the System 33](#_Toc6216)

[6.3 Installation guide for Application 33](#_Toc29956)

[6.4 Operating Manual 33](#_Toc12798)

[Figure 1 system diagram illustrating the major components and their interactions 10](#_Toc241)

[Figure 2 Composite Viewpoint 20](#_Toc30490)

[Figure 3 Logical Viewpoint 21](#_Toc25761)

[Figure 4 Information Viewpoint 22](#_Toc3022)

[Figure 5 Interaction Viewpoint 23](#_Toc12192)

[Figure 6 State Dynamic Viewpoint 24](#_Toc11711)

[Table 1 State Viewpoint table 25](#_Toc8011)

[Table 2 Algorithm Viewpoint Table 26](#_Toc23140)

[Table 3 Development Plan 29](#_Toc25460)

[Table 4 Traceability Matrix 31](#_Toc6715)

[Table 5 Test Case for Application Startup 32](#_Toc4202)

[Table 6 Test Case for Adoption Form Submission 32](#_Toc22900)

[Table 7 Test Case for Chatbot Response 33](#_Toc22932)

# CHAPTER 1 INTRODUCTION

# INTRODUCTION

Pawtect is a holistic web platform aiming at protecting and enhancing the lives of pets and animals in need. The platform provides an integrated solution for pet adoption, surrender, stray pickup, donations, and volunteers. Pawtect ensures that interactions are seamless with users, whether adopters, pet owners, those requesting rescue services or volunteers, have the perfect experiences according to their needs. Furthermore, the platform has automated email sending system, which aids in improving communication between the pet owner and veterinarians, thus boosting transparency and care during the adoption or surrendering process.

With Pawtect, we have tried to bridge the gap between animal welfare organizations, potential adopters, and pet lovers, promoting animal well-being and humanity. The intuitive interface and robust functionalities are designed to make an impact in animal rescue and pet care.

## System Introduction

Pawtect is a comprehensive web platform designed to revolutionize animal welfare by providing an integrated suite of features for protecting and improving the lives of pets and animals in need. This system serves as a central hub connecting animal welfare organizations, prospective adopters, pet owners, and volunteers. Pawtect streamlines critical processes such as pet adoption and surrender, facilitates the coordination of stray animal pickups, manages donations effectively, and supports volunteer efforts. By offering a user-friendly interface and robust functionalities, Pawtect aims to create seamless interactions and ensure optimal experiences for all users, tailored to their specific needs within the animal welfare ecosystem.

Furthermore, Pawtect incorporates an automated email communication system to enhance transparency and care, particularly during the adoption and surrender processes, by fostering better communication between pet owners and veterinarians. Ultimately, Pawtect seeks to bridge the existing gaps within the animal welfare landscape by providing a holistic solution that promotes animal well-being and strengthens the connection between humans and animals.

## Background of the System

An old project, "Four Paws," is available on the RMS website. This application caters to people looking to adopt pets, buy pet supplies, and volunteer. Despite serving the core needs, "Four Paws" cannot fully support the breadth and depth of features of an extensive animal welfare system. Pawtect enhances this foundation through a more versatile and feature-rich platform. Unlike "Four Paws," Pawtect features donation, and volunteering options, sharing capabilities of missing pet posters, and innovative features that connect the users to both human and AI-powered veterinarians.

Globally, Petfinder and Adopt-a-Pet, to name a few, primarily focus on pet adoption or specific animal welfare aspects. These systems normally operate in isolated domains, leaving gaps in functionalities such as pet surrender, disaster response, and AI-driven support tools for first aid and pet suitablity assessment. Pawtect differs by addressing all aspects of animal welfare holistically. Since it has a focus on integrating pet adoption, veterinary consultations, and AI-powered recommendations into one cohesive system, it caters to both the needs of the individual and the organization. Its end-to-end solutions concerning pet surrender, disaster or stray animal pickup, and volunteer management in one platform make it unique.

## Objectives of the System

* To provide a centralized platform for pet adoption, surrender, and disaster or stray animal
* pickup.
* To utilize AI-powered algorithms for matching pets with adopters based on preferences and lifestyle compatibility.
* To enable seamless communication between users and veterinarians via video calls.
* To support animal welfare organizations with volunteer recruitment and donation
* management.
* To ensure transparency and accountability in the adoption and surrender process.
* To enhance the efficiency of animal rescue efforts through integrated mapping and tracking tools.

## Significance of the System

Pawtect’s significance lies in its ability to revolutionize how pet welfare services are managed and

delivered. Its impact spans various application areas:

● **Animal Welfare:** Streamlining adoption, rescue, and surrender processes ensures better

care for animals in need.

● **Community Engagement:** Encourages community involvement through volunteer

opportunities and donation drives.

● **Veterinary Support:** Provides a trusted channel for users to consult veterinarians,

improving pet health outcomes.

● **Disaster Management:** Facilitates quick response and recovery for stray and disaster-

affected animals.

# CHAPTER 2

# REQUIREMENT SPECIFICATIONS

# REQUIREMENT SPECIFICATIONS

This chapter outlines the software requirements for Pawtect, including the scope, intended users, functionalities, and operating environment. It provides detailed functional and behavioral requirements along with external interface designs to guide system implementation.

## Product Scope

The scope of Pawtect centers on providing functionalities for the care of animals while defining its boundaries clearly to maintain focus and efficiency. Pet adoption and surrender processes are supported, with an AI-powered pet matching system and a chatbot to answer any inquiries from users. The recruitment and management of volunteers are supported, and disaster or stray animal pickups are coordinated. Veterinary consultations can be carried out via video calls, and monetary and in-kind donations are managed alike.

Other things are outside Pawtect's scope, so it remains focused on its intended role. These include real-time animal tracking devices or IoT integration, veterinary services beyond consultations, and wider-scale wildlife rescue services beyond domesticated pets. By keeping such distinctions, Pawtect is keen to stay clear of its envisioned role for the mission of using innovative technological solutions to enhance the lives of pets effectively.

## Product Description

### Product Perspective

A single, self-sufficient product whose capacities can greatly revolutionize pet welfare services, integrating several functionalities on a single platform, unlike most existing systems that focus more on specific aspects of animal care adoption or managing donations, for example. The scope

of features that Pawtect will have includes pet surrender, adoption, disaster rescue, management of volunteers, and AI-powered tools for matching pets and support from a chat-bot.

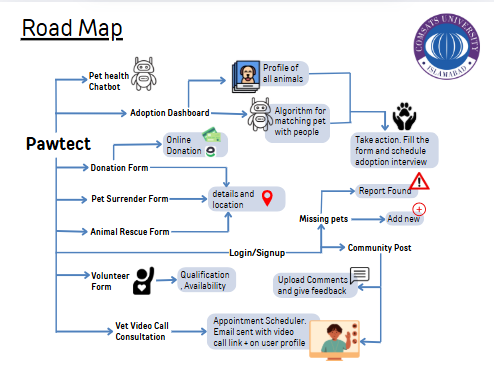


Figure 1 system diagram illustrating the major components and their interactions

The centralized platform will take the form of a system that will enable easy interactions between users, veterinarians, animal welfare organizations, and volunteers. With cutting-edge AI and cloud-based technologies, Pawtect offers a scalable, reliable, and accessible solution to better the lives of pets.

### Product Functionality

Pet Adoption Management

Users can browse available pets using filters such as species, breed, and age.

Adoption Interest Form allows users to express intent to adopt.

Automated emails and interview scheduling streamline the adoption process.

Pet Surrender Management

Pet owners can surrender their pets by submitting detailed information including pet history, health status, and images.

The form is accessible both from the homepage and the dedicated surrender section.

Disaster/Stray Animal Reporting

Users can report injured, stray, or disaster-affected animals through a dedicated form.

Requests are routed to nearby volunteers or admins.

Volunteer CV Submission

Interested individuals can submit their CVs and availability to assist with animal welfare activities.

Form is embedded on the homepage for quick access.

Donations

Users can contribute through a simple donation form with secure payment options.

Donations support Pawtect’s rescue, shelter, and medical activities.

AI Chatbot

A floating chatbot is available site-wide, answering questions about pet health and care for cats, dogs, and birds.

Powered by FAISS and FLAN-T5 for semantic search and natural language understanding.

Veterinary Consultation

Users can book a vet appointment by submitting a form with pet and issue details.

Upon submission, an automated email with a scheduled video call link is sent.

The link activates only on the selected appointment date.

Vets can access the schedule and related information through their dashboard.

Admin Dashboard

Admins manage pet listings, rescue cases, adoption applications, volunteer CVs, and vet appointments.

Includes approval actions, appointment control, and record monitoring.

Vet Dashboard

Veterinarians log in securely to view upcoming appointments.

Each entry includes pet details and direct access to the consultation link.

### Users and Characteristics

The anticipated users of Pawtect can be categorized as follows:

**Adopters:**

● **Frequency of Use:** Moderate to high during the adoption process.

● **Characteristics:** Typically, pet enthusiasts have varying levels of technical expertise.

**Pet Owners:**

● **Frequency of Use:** Moderate for surrender or veterinary consultation purposes.

● **Characteristics:** May have an emotional attachment to pets, requiring a user-friendly

interface.

**Missing Pet Owners:**

● **Frequency of Use:** Moderate to high after uploading the missing pet poster.

● **Characteristics:** May have an emotional attachment to pets, requiring a user-friendly

interface.

**Volunteers:**

● **Frequency of Use:** High for applying as a Volunteer.

● **Characteristics:** Technologically proficient, motivated to support animal welfare.

**Animal Welfare Organizations:**

● **Frequency of Use:** High for managing adoption, surrender, and rescue requests.

● **Characteristics:** Professional users requiring robust functionality and reporting.

**Priority Users:** Adopters and animal welfare organizations are the primary focus due to their critical role in achieving the system's objectives.

### Operating Environment

Pawtect will operate in the following environments:

● **Hardware Requirements:**

 Web: Minimum 4 GB RAM, dual-core processor, and a stable internet connection.

● **Operating System:**

 Web: Compatible with Chrome, Firefox, Safari, and Edge browsers.

● **Software Dependencies:**

 Backend: Node.js, MongoDB.

 Frontend: React.js.

 External APIs: Google Maps API for location services, Stripe/PayPal for payment

processing, and Twilio for video communication.

This environment ensures compatibility across devices and platforms, supporting seamless

functionality

## Specific Requirements

### Functional Requirements

This section provides a detailed breakdown of Pawtect's functionalities, divided into major

functional areas.

**1. Pet Adoption Management**

● Users can browse available pets with filters (species, breed, age, size, etc.).

● Adopters can create accounts, save favorites, and apply for adoptions.

● System matches adopters with suitable pets using an AI-powered algorithm based on user

preferences and pet profiles.

● Notifications and updates about application status.

**2. Pet Surrender Management**

● Pet owners can submit surrender requests, including details like pet history, medical

records, and reasons for surrender.

● Option for scheduling a pickup for surrendered pets.

● System ensures humane processing and coordination with animal shelters.

**3. Disaster/Stray Animal Pickup**

● Users can report stray or disaster-affected animals with location details via Google Maps

integration.

● The system routes requests to the nearest volunteer or organization. ● Status tracking of

pickup requests.

**4. Volunteer Management**

● Volunteers can apply and specify availability.

**5. Donation and Payment Processing**

● Secure payment options for donors (Stripe, PayPal).

● Transparent donation tracking for adopters and organizations.

● Automatic receipt generation and monthly donation summaries.

**6. AI Chatbot**

● Provide instant responses to FAQs about adoption, surrender, and donations.

● Capable of guiding users through processes like filling out forms or applying for adoption.

**7. Veterinary Consultation**

● Users request a vet appointment by filling out a form with pet details, date, and reason.

● Upon form submission, the system sends an automated email to the user with a video consultation link.

● The video link is configured to be valid on the appointed date/time only.

● Veterinarians view the scheduled appointments via their dashboard.

### Behavioral Requirements

**Use Case Descriptions:**

1. **Adopt a Pet:**

 **Actors:** Adopter, System.

 **Description:** Adopters search for pets, apply for adoption, and receive updates.

2. **Surrender a Pet:**

 **Actors:** Pet Owner, System.

 **Description:** Pet owners submit surrender requests and schedule pickups.

3. **Volunteer Application:**

 **Actors:** Volunteer, Organization.

 **Description:** apply as a volunteer.

4. **Consult a Vet:**

 **Actors:** Pet Owner, System.

 **Description:** Users schedule consultations.

### External Interface Requirements

#### User Interface

**Home Page:**

Serves as the central hub of the platform with direct access to all major sections via the top navigation bar.

Features quick-access embedded versions of the Surrender Form, Rescue Form, and Volunteer CV Submission Form to ensure immediate action for high-priority use cases.

Also includes links/buttons for Adoption Dashboard, Donate, AI Chatbot, and Vet Appointment.

**Adoption Dashboard**

Displays a gallery of adoptable pets with filters and search options.

Each pet card links to a detailed profile with an "Express Interest" button leading to the Adoption Interest Form.

**Surrender Form**

A detailed input form collecting pet type, age, medical history, image, and owner details.

Also embedded on the home page for faster access in urgent cases.

**Rescue Form**

Lets users report injured, stray, or disaster-affected animals with fields for location, description, and contact details.

Accessible both from the navbar and directly on the home page.

**Vet Appointment Form**

Users schedule online consultations by selecting date, time, pet type, and reason.

Confirmation message shown on submission, with automated email including the video call link.

**Volunteer CV Submission**

A simple form for uploading a CV, adding contact information, and availability.

Also available on the home page for easy access.

**Donation Form**

A lightweight form to enter donor info and payment details.

AI Chatbot Interface

A fixed chat icon accessible on every page, offering instant pet care advice for cats, dogs, and birds

**Admin Interfaces**

**Admin Dashboard**

Allows management of pet listings, volunteer entries, adoption and surrender requests, rescue cases, and appointment scheduling.

Includes action buttons like Approve, Reject, Edit, and Schedule.

**Vet Interfaces**

Vet Login Page, secure login portal for veterinarians.

**Vet Dashboard**

Displays all scheduled appointments, details about pets and owners, and video consultation links.

**Standard UI Components**

Universal buttons such as Submit, Cancel, and Back on every form.

Real-time input validation and inline error messages.

Modal popups for confirmations, and toast notifications for successful or failed actions.

Consistent header/footer design with quick navigation and contact/help information.

These interfaces are designed to maximize usability while minimizing confusion, ensuring users can access help or take action within just a few clicks.

#### Other Interfaces (if any)

#### ****a. Hardware Interfaces****

Pawtect is a web-based application and does not require specialized hardware components for general use. It is designed to run on standard consumer devices such as desktops, laptops, tablets, and smartphones.

**b. Software Interfaces**

Pawtect interfaces with several essential software components:

Operating System Compatibility:

Designed to be platform-independent via web browsers, compatible with Windows, Linux, macOS, Android, and iOS.

**Backend Stack:**

Node.js for server-side logic and API routing.

FastAPI (Python) for handling AI chatbot and ML-based scheduling/logics.

MongoDB (Compass) as the database for storing user data, pet listings, appointments, and form submissions.

**Frontend Stack:**

React.js for building user interfaces.

CSS/Bootstrap for styling.

Axios/Fetch for client-server communication.

**Libraries & APIs:**

Nodemailer for sending automated emails (e.g., appointment confirmations).

Express.js for defining API routes.

**c. Communication Interfaces**

**Pawtect primarily relies on standard web communication protocols:**

HTTP/HTTPS is used for all web communications between the frontend and backend. HTTPS ensures encrypted and secure data transmission.

**Email Communication:**

Auto-generated emails (using SMTP via Nodemailer) are sent for vet appointments and adoption interview confirmations.

**Video Conferencing Links:**

Admin schedules consultations using any standard platform, and the system sends the meeting links via email.

**Security:**

Sensitive data such as email, contact info, and appointment details are transmitted securely. Future plans include adding JWT-based authentication and SSL encryption on server endpoints.

## Non-functional Requirements

### Performance Requirements

The system should meet the performance requirements under varying conditions. Some of the

performance requirements for the Pawtect system are:

**1. Response Time:**

● The system will respond to the actions of the users on the system, for example, button click,

form submission, in 2 seconds.

● Search results for pets, shelters, or volunteers should be returned within 5 seconds of

entering it.

**2. System Throughput:**

**●** The system should meet the requirements of handling 100 concurrent users without

degrading performance.

**●** The platform should support up to 1,000 adoption applications per day.

**3. Data Processing:**

● When adopter preferences are submitted, the pet matching algorithm using artificial

intelligence should take no more than 3 seconds to complete the match operation.

● Video calls should have a latency of less than 200ms to ensure effective communication

between the users and the veterinarians.

**4. Load and Stress Handling:**

● The system should be stress-tested for up to 200 concurrent requests during peak times

without failure.

● Pawtect should work flawlessly under stress test conditions with a 20% escalation in

average daily traffic during disaster or special event.

**5. Availability:**

• The system must be available for accessible usage by users worldwide, with a minimum of 99.9% uptime.

**6. Transaction Time:**

● Processing of donations and payments should be accomplished in 7 seconds from

submission to confirmation.

### Safety and Security Requirements

**Safety Requirements**

**1. Backup and Recovery of Data:**

● Daily backups of all critical data to enable recovery in case of a failure.

**2. Disaster Recovery Plan:**

● An effective disaster recovery plan must be implemented for system restoration within 60 minutes in case of a critical failure.

● Conducting regular drills and testing of disaster recovery plans to be adequately prepared.

**3. System Monitoring and Alerts:**

● System Health and Performance Monitoring.

● Time-based alerts to administrators on any suspicious activities or critical system failures

**Security Requirements**

**1. User Identity Authentication:**

● Implementation of multifactor authentication (MFA) for all user accounts, especially those with administrative rights.

● Passwords must adhere to a strong complexity rule, (min. 8 characters; includes uppercase, lowercase, numbers, and special characters.)

**2. Access Control:**

● Role-based access control, to avoid resource and function access by users except those

relevant to their roles.

● Administrative and sensitive operations should be accessible only to authorized personnel.

**3. Privacy Compliance:**

● The platform must follow compliance with data protection regulations such as GDPR and CCPA to ensure proper privacy of its users.

● Ability to request deletion or export of personal data based on privacy laws

**4. Audit Logging and Monitoring:**

● All critical activities (such as login attempts, data access, and configuration changes) must be logged.

● Secure storing of logs and review monthly in terms of security threats or unusual activity

**5. Input Validation and Security Testing:**

● Input validation for all incoming user inputs to prevent security threats of SQL injection and Cross Site Scripting (XSS)

● Normal penetration testing and vulnerability assessments to detect and mitigate any identified risks.

**6. Session Management**

● User sessions to terminate after 15 minutes of inactivity.

● Sessions should be handled securely using HTTP Only and Secure cookies.

### Software Quality Attributes

The following quality attributes ensure that the Pawtect system meets the expectations of both

users and developers. These attributes are specified to be measurable, verifiable, and actionable.

**Reliability**

● **Requirement**: The system must have an uptime of **99.9%**, ensuring high availability for users globally.

● **Implementation Plan**:

 Use redundant servers and cloud-based infrastructure to handle failover.

 Regular system health checks and automatic error recovery mechanisms.

**Portability**

● **Requirement**: The platform must be compatible with various environments, including

Windows, macOS, iOS, and Android.

● **Implementation Plan**:

 Develop the front end using **React.js** for cross-platform compatibility.

 Ensure mobile compatibility with responsive web design and testing across multiple devices.

**Maintainability**

● **Requirement**: System updates, bug fixes, and feature enhancements should be deployable

without affecting system availability.

● **Implementation Plan**:

 Use modular architecture to isolate changes.

 Maintain comprehensive documentation and version control via **Git**.

 Automate testing and deployment pipelines for seamless updates.

**Adaptability**

● **Requirement**: The system should easily integrate new features (e.g., future IoT capabilities

for pet tracking) without major redesigns.

● **Implementation Plan**:

 Design flexible APIs to allow seamless integration with third-party systems

 Ensure system modules are loosely coupled to accommodate future extensions.

**Interoperability**

● **Requirement**: The system should interact efficiently with external services, such as

**Google Maps**, **Stripe**, and **Twilio**.

● **Implementation Plan**:

 Use standardized communication protocols like **RESTful APIs**.

 Conduct interoperability testing to ensure seamless integration.

**Usability**

● **Requirement**: The platform should be easy to navigate, with minimal learning curves for users of all technical skill levels.

● **Implementation Plan**:

 Conduct usability testing with diverse user groups.

 Implement clear visual cues, intuitive navigation, and comprehensive onboarding guides.

**Robustness**

● **Requirement**: The system should handle unexpected input and usage scenarios without crashing.

● **Implementation Plan**:

 Implement thorough error handling and logging.

 Use defensive programming techniques to ensure system stability.

**Reusability**

● **Requirement**: Core components such as the AI pet matching algorithm should be reusable

in different modules.

● **Implementation Plan**:

 Develop reusable libraries for AI and user management modules.

 Document reusable components to facilitate easy integration.

**Testability Requirement**: All system components must be testable individually and as part of the entire system.

● **Implementation Plan**:

 Use unit testing for individual components and integration testing for interconnected modules.

 Automate regression testing to ensure system stability after updates.

**Availability**

● **Requirement**: The system must be accessible 24/7 with minimal downtime.

● **Implementation Plan**:

 Implement load balancing and server redundancy to prevent service disruptions.

 Regularly monitor system performance using cloud-based monitoring tools.

This section ensures that Pawtect delivers high-quality, reliable, and adaptable software that meets both user and developer expectations.

# CHAPTER 3

# DESIGN SPECIFICATIONS

# DESIGN SPECIFICATIONS

## Introduction

## Composite Viewpoint

Figure 2 Composite Viewpoint

## Logical Viewpoint

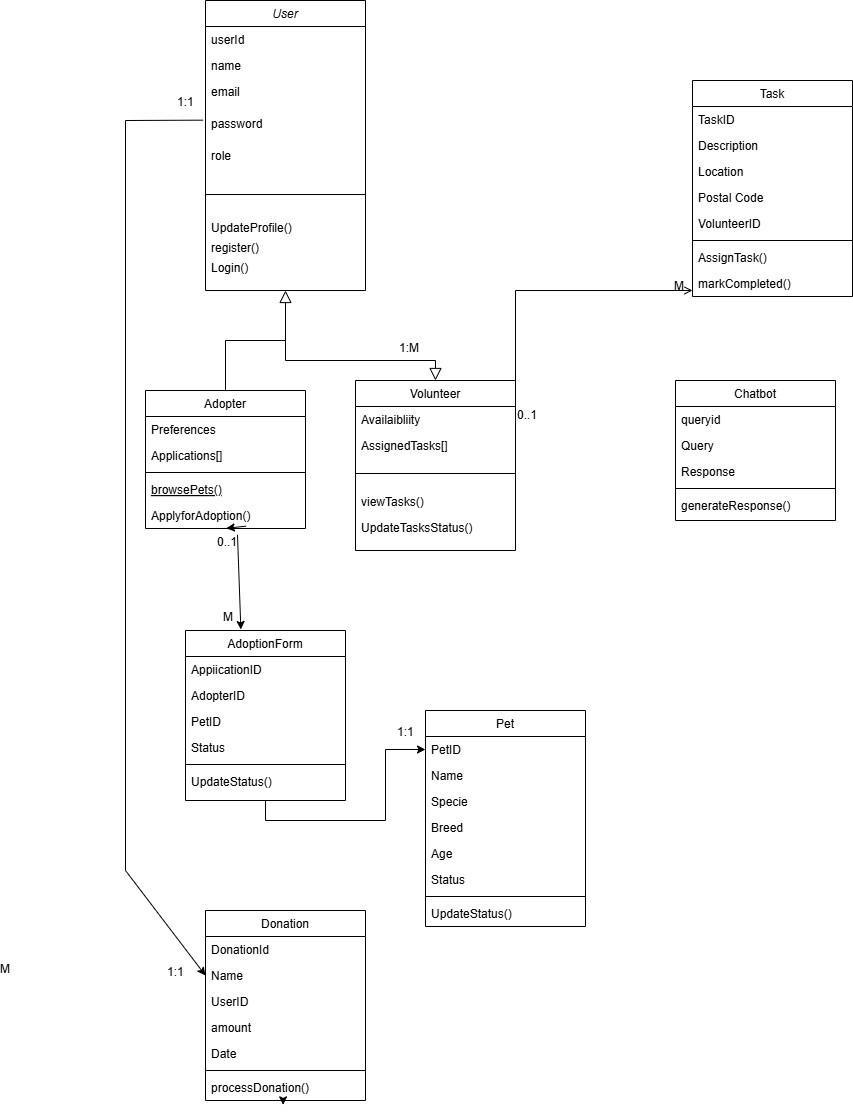


Figure 3 Logical Viewpoint

## Information Viewpoint

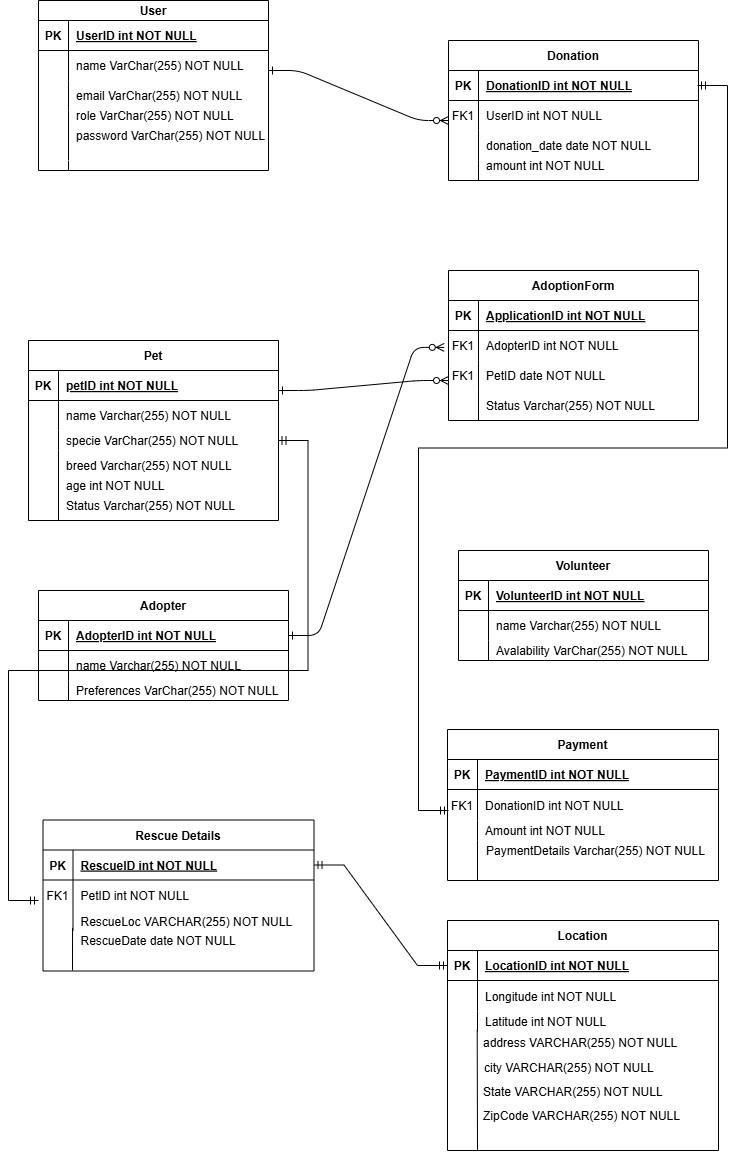


Figure 4 Information Viewpoint

## Interaction Viewpoint

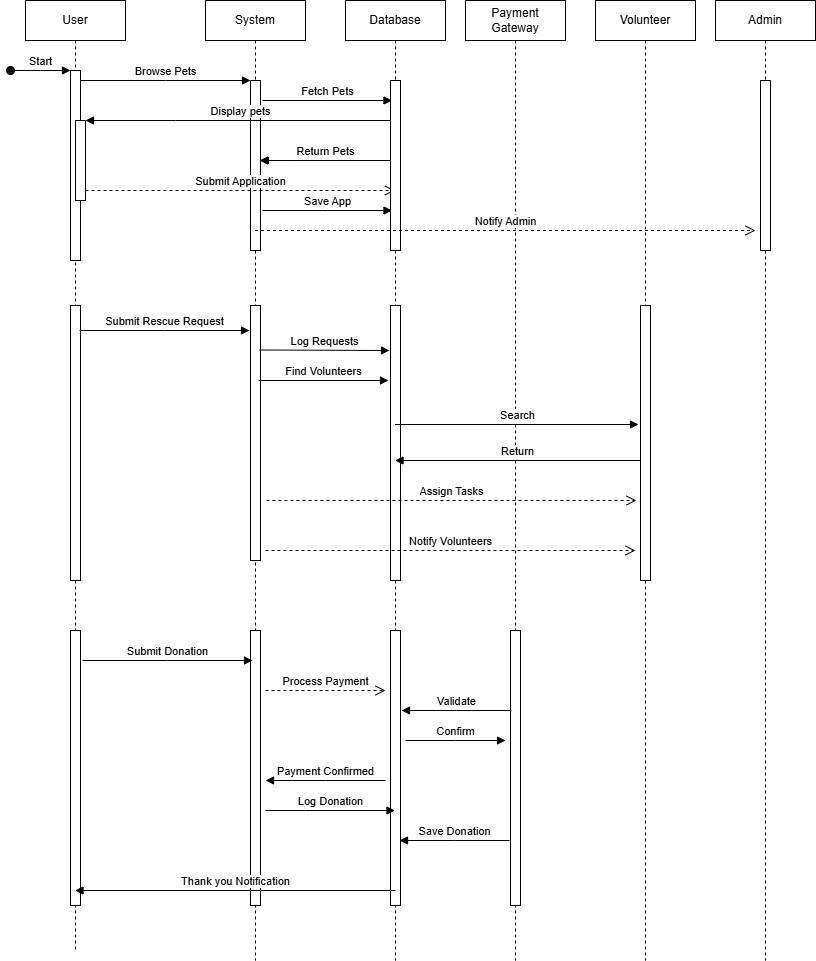


Figure 5 Interaction Viewpoint

## State Dynamics Viewpoint

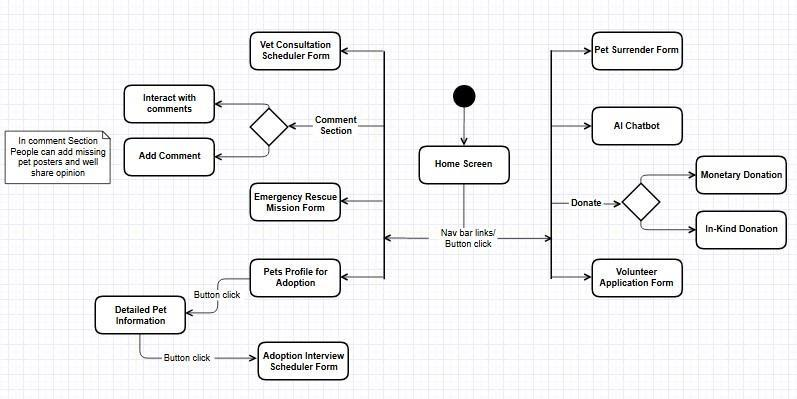


Figure 6 State Dynamic Viewpoint

|  |  |  |  |
| --- | --- | --- | --- |
| **Current State** | **Event/Trigger** | **Next State** | **Action Performed** |
| **Home page** | User Searched for Pawtect website | Viewing Available options | Displaying available options |
| **Browsing** | The user clicks on "Adopt a Pet" | Viewing Pet Listings | Display a list of available pets for adoption. |
| **Viewing Pet Listings** | The user selects a pet | Adoption Details | Show pet details and adoption request form. |
| **Adoption Details** | User submits adoption  request | Request Submitted | Record adoption requests in the system. |
| **Browsing** | User clicks on "Surrender a Pet" | Pet Surrender Form | Display the pet surrender form. |
| **Pet Surrender Form** | User submits surrender form | Surrender Request Logged | Save the pet surrender request in the system. |
| **Browsing** | The user clicks on "Donate" | Donation Options | Display financial and physical donation options. |
| **Donation Options** | User selects a financial donation | Financial Donation  Form | Display donation amount input and payment options. |
| **Donation Options** | User selects the physical supply donation | Supply Donation  Details | Record details of donated supplies and pickup location. |
| **Browsing** | User clicks on "Volunteer" | Volunteer Form | Display volunteer sign up form. |
| **Volunteer Form** | User submits volunteer form | Volunteer Registered | Save volunteer information in the system. |

Table 1 State Viewpoint table

## Algorithmic Viewpoint

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Condition** | **C1: User views pets** | **C2: User submits adoption request** | **C3: User surrenders pet** | **C4: User donates** | **C5: Volunteer signs up** | **C6: Emergency pickup** |
| **Action A1:**  **Display pets** | yes | no | no | no | no | no |
| **Action A2: Submit request** | no | yes | no | no | no | no |
| **Action A3:**  **Collect pet info** | no | no | yes | no | no | no |
| **Action A4: Process donation** | no | no | no | yes | no | no |
| **Action A5: Register volunteer** | no | no | no | no | yes | no |
| **Action A6:**  **Collect rescue**  **Information** | no | no | no | no | no | yes |

Table 2 Algorithm Viewpoint Table

### Explanation of Decision Table

* **Conditions (C1, C2, C3, etc.)**: Represent specific user or admin actions.

**Actions (A1, A2, A3, etc.)**: Represent the system's responses to the conditions.

# CHAPTER 4

# DEVELOPMENT AND TOOLS

# DEVELOPMENT AND TOOLS

This chapter discusses the practical execution of the Pawtect project. It includes an overview of how development was carried out, team responsibilities, tools and technologies used, and potential areas for future development or extension. The reader will find a structured breakdown of the team’s workflow, software environment, and planned improvements based on real-world needs.

## Introduction

This chapter outlines the complete development journey of the Pawtect project. It highlights the responsibilities of each team member, the tools and technologies utilized, and the planned milestones that guided the implementation. The chapter also concludes with insights into how the platform can be further extended or scaled in the future.

## Development Plan

This project was developed by a team of two members:

1. **Anza Malik**
2. **Romysa Siddiqui**

The development plan was divided into key milestones and responsibilities to ensure efficient collaboration. Below is the tabulated breakdown of tasks and roles:

| **Task** | **Assigned To** | **Timeline** | **Status** |
| --- | --- | --- | --- |
| Requirement Gathering & UI | Anza Malik | Week 1 | Completed |
| Frontend (React UI + Navigation) | Anza Malik | Week 2–3 | Completed |
| Backend API Setup (Node.js) | Romysa Siddiqui | Week 3–4 | Completed |
| MongoDB Integration | Both | Week 4 | Completed |
| Form Handling (Rescue, Surrender, Volunteer) | Anza Malik | Week 5 | Completed |
| Adoption Dashboard | Romysa Siddiqui | Week 6 | Completed |
| Notification System | Romysa Siddiqui | Week 7 | Completed |
| Vet Dashboard & Appointment Module | Anza Malik | Week 8 | Completed |
| AI Chatbot Integration (FastAPI + FAISS and flan-t5) | Anza Malik | Week 9 | Completed |
| Admin Dashboard & Scheduling Features | Romysa Siddiqui | Week 10 | Completed |
| User Dashboard | Romysa Siddiqui | Week 11 | Completed |
| Testing, Debugging & Deployment | Both | Week 11–12 | Ongoing |

Table 3 Development Plan

## Development Tools

Frontend: React.js, Bootstrap, CSS

Backend: Node.js, Express.js

Database: MongoDB (via MongoDB Compass)

Chatbot & AI: Python (FastAPI), FastAPI, FAISS and flan-t5

Email Services: Nodemailer (SMTP)

IDE & Editors: Visual Studio Code, Google Colab

Design: Canva

## Conclusion and Future Work/Extensions

Pawtect successfully fulfills its core goal of supporting pet welfare by providing modules for adoption, rescue, surrender, and veterinary consultation, backed by automation and AI assistance. The project offers a centralized solution for both users and administrators to manage pet-related processes in an organized and compassionate way.

Future extensions may include a dedicated **mobile app**, **real-time chat with vets**, **user login and history tracking**, **payment gateway integration**, and an **AI-based pet matching algorithm** to suggest adoptable animals based on user lifestyle and preferences. Additionally, implementing **JWT-based authentication**, **admin analytics dashboard**, and **role-based access control** will enhance both security and usability.

# CHAPTER 5

# QUALITY ASSURANCE

## 

# QUALITY ASSURANCE

## Introduction

## Traceability Matrix

| **Requirement ID** | **Launch Activity** | **Press Button** | **Stores Data** | **Machine Learning Model** | **Vet Appointment** | **Pet Adoption** | **Displaying Data** | **Pet Recommendation** | **# of Test Cases** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Display Pages | ✔ | ✔ |  |  |  |  |  |  | 2 |
| MongoDB Connection |  |  | ✔ |  |  |  |  |  | 1 |
| FastAPI working |  |  |  | ✔ |  |  |  |  | 1 |
| Automatic Emails |  |  | ✔ |  | ✔ | ✔ |  |  | 2 |
| Admin Dashboard |  |  | ✔ |  |  | ✔ | ✔ |  | 2 |
| Vet Dashboard |  |  | ✔ |  | ✔ |  |  |  | 1 |
| User Profile |  | ✔ | ✔ |  |  | ✔ | ✔ |  | 2 |

Table 4 Traceability Matrix

## Test Plan

| **Test ID** | **TC\_1** |
| --- | --- |
| **Test Name** | Application Startup |
| **Date of Test** | 01/05/2025 |
| **Application** | Pawtect |
| **Description** | Home screen loads with all modules |
| **Input** | Launch application in browser |
| **Expected Output** | Home page displayed with nav links |
| **Actual Output** | As expected |
| **Test Role** | Developer |
| **Test Verified By** | Team Member |

Table 5 Test Case for Application Startup

| **Test ID** | **TC\_3** |
| --- | --- |
| **Test Name** | Adoption Form Submission |
| **Date of Test** | 01/05/2025 |
| **Application** | Pawtect |
| **Description** | Submitting the adoption form |
| **Input** | Valid user details in form |
| **Expected Output** | Confirmation message, email sent |
| **Actual Output** | As expected |
| **Test Role** | End User |
| **Test Verified By** | Team Member |

Table 6 Test Case for Adoption Form Submission

| **Test ID** | **TC\_11** |
| --- | --- |
| **Test Name** | AI Chatbot Emergency Tip Response |
| **Date of Test** | 02/05/2025 |
| **Application** | Pawtect |
| **Description** | Ask chatbot for dog choking help |
| **Input** | “Common Cat Illnesses” |
| **Expected Output** | Dental issues |
| **Actual Output** | Relevant response shown |
| **Test Role** | End User |
| **Test Verified By** | QA Team |

Table 7 Test Case for Chatbot Response

# CHAPTER 6

# USER MANUAL

# USER MANUAL

<*It is highly recommended that you write brief summary of every chapter as a preamble explaining what a reader would find in this chapter. Thus, when a reader reads summary of your chapter at the beginning, the forthcoming contents shall become clear.*>

This chapter is an end-user guide on how to properly use the Pawtect web platform. It includes everything from system requirements and getting started to step-by-step instructions on every key feature, troubleshooting tips for common problems, and support access. The aim is to have an easy and seamless experience for every type of user, from adopters to pet owners, volunteers, and animal welfare organizations.

## Introduction

<*Briefly describe what this chapter includes.*

This chapter is an end-to-end guide on how to utilize Pawtect. It touches on what one needs to get started, accessing the system, and utilizing the major features—such as the community page, adoption, surrender, donations, and so forth. With precise, step-by-step directions, it guides the user through the platform with ease and troubleshoots common issues independently.

## Hardware/Software Requirements for the System

<*Specify all Software/Hardware requirements for installation and execution of the application.*

**Hardware Requirements for system:**

* Minimum 4 GB RAM.
* dual-core processor.
* stable internet connection.
* screen resolution of at least 1024×768 is recommended.
* Pawtect is fully responsive on tablets and smartphones.
* For donation payments, a credit/debit card or PayPal account may be required.

**Software Requirements for system:**

* No installation required; access via web browser.
* OS should be compatible with Chrome, Firefox, Safari, and Edge browsers.
* Stable Internet Connection.
* A modern internet browser (e.g. Chrome, Firefox, Edge, or Safari) on any device.

## Installation guide for Application

<*This section should include comprehensive guidelines for installation procedure.*

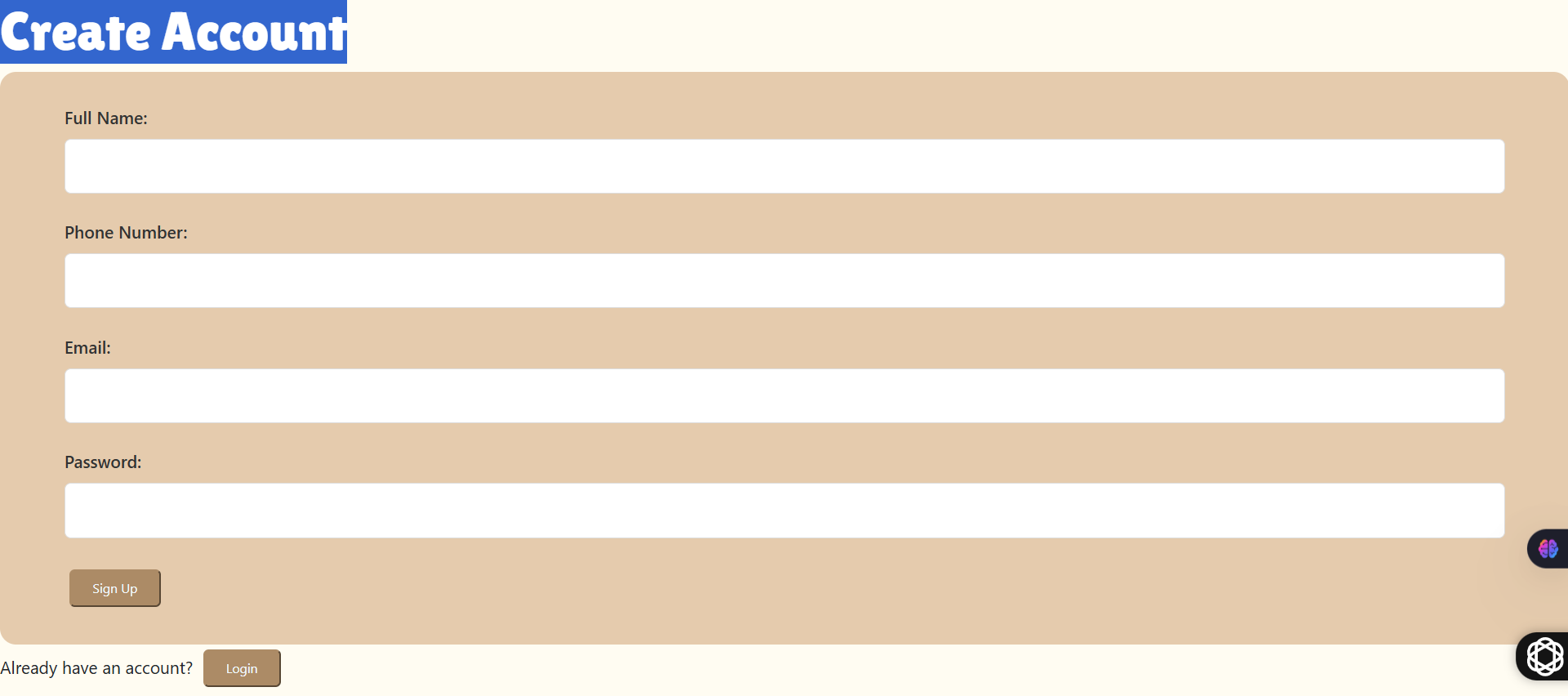
TO DO: Specify all steps in bullet form along with screenshots.>

Pawtect is online web-based, which means no download of software is required. Use the following instructions to access and begin using Pawtect:

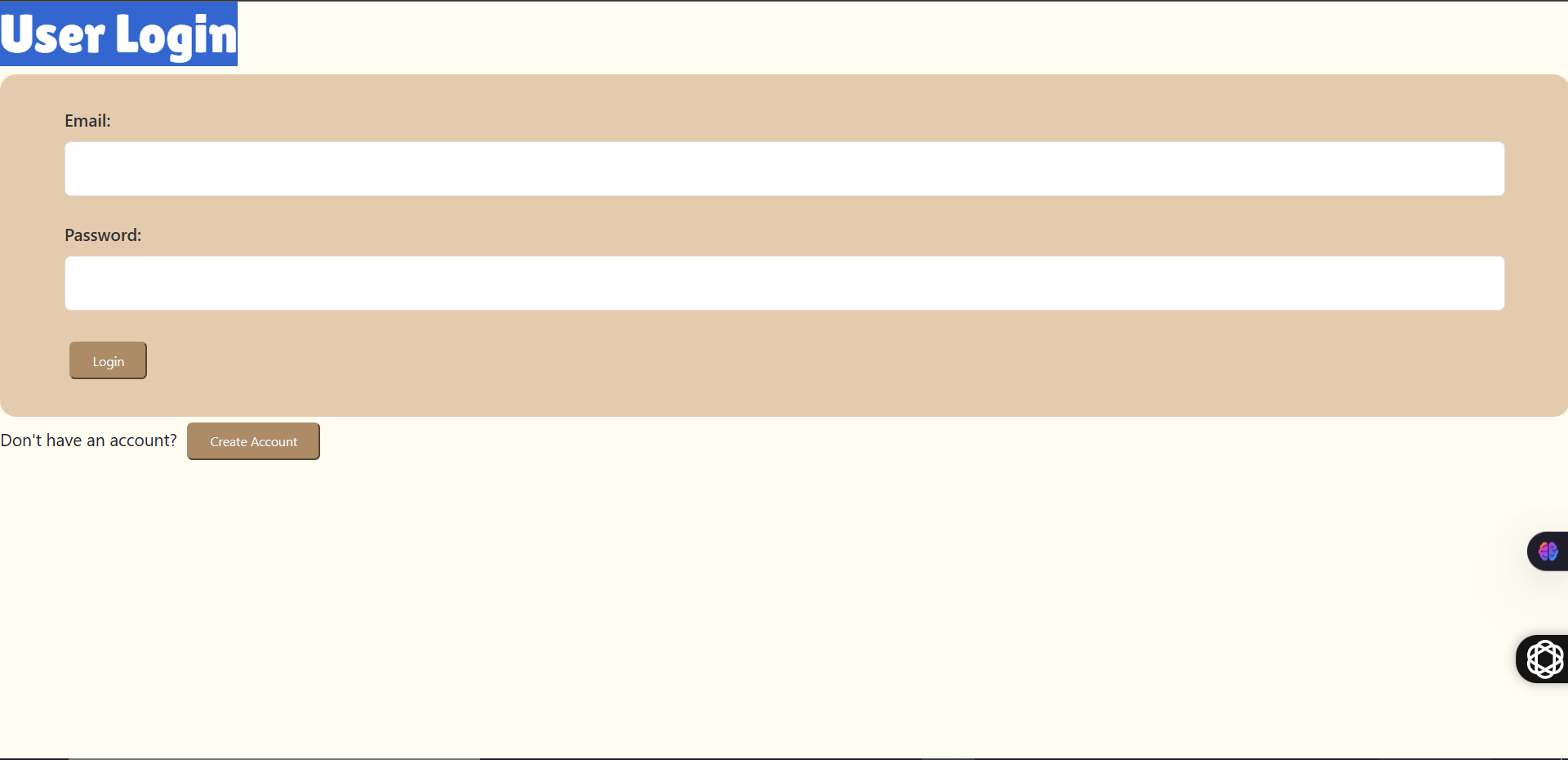
* + - Open your internet browser (Google Chrome, Firefox, Safari, etc.). Make sure it's up to date with the current stable version.
    - Visit the Pawtect website. Type in the web address (e.g. https://www.pawtect.org/) into the address bar and click Enter. The homepage for Pawtect will appear. **(homepage)**:



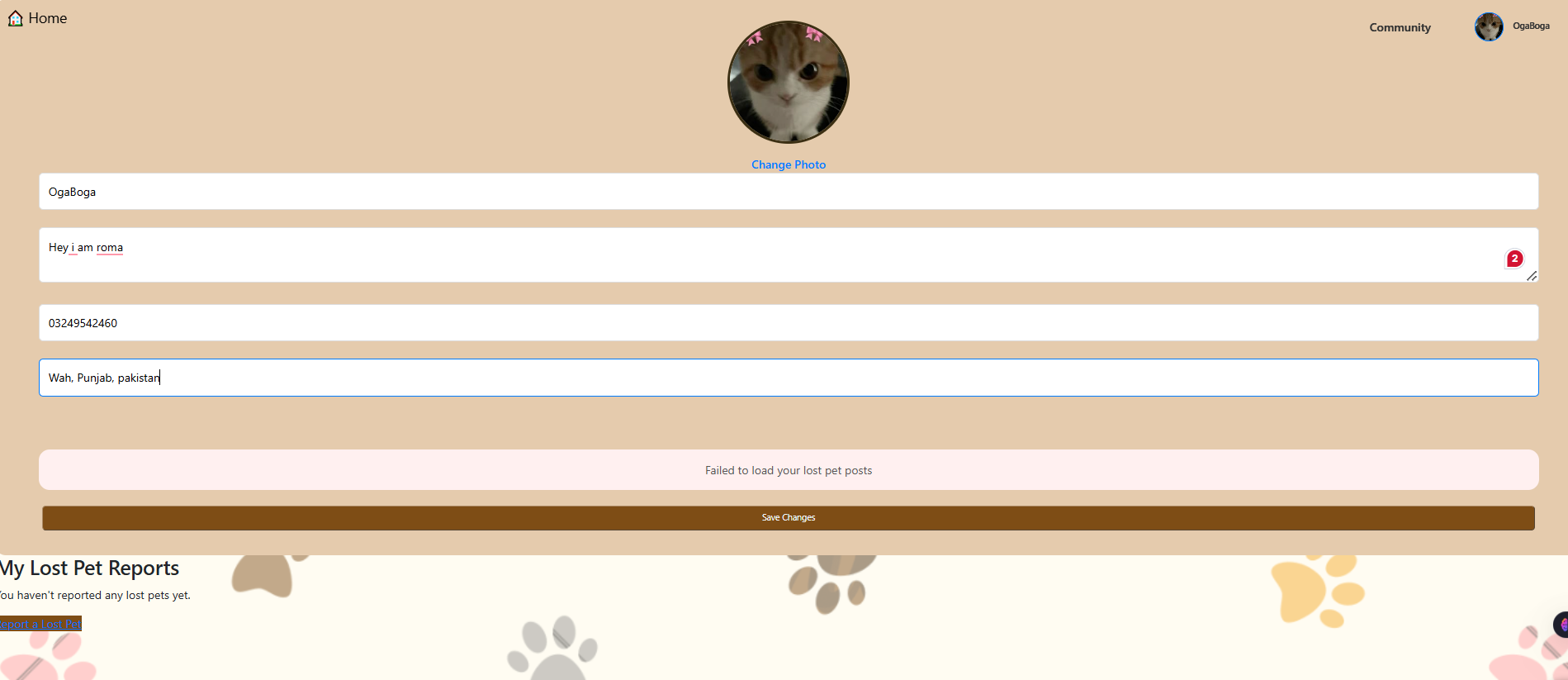
* + - Create an Account: If you are signing up for the first time, click the Register or Sign Up button. Enter your name, email, password, and other requested information. Submit the form. You will get a verification email; click the verification link in the email to activate your account. **(SignUp form):**



* + - Log In: If you have a registered account, click Login. Type in your registered email and password, and click Submit or Sign In.



* + - Set Preferences: In account settings, you can change your profile details, contact address, and notification settings. This allows Pawtect to be in touch and personalize the experience. **(user dashboard)**:



* + - After login, you can navigate the site and access its features. Log out after every session in case you are working with a shared computer.

## Operating Manual

<*This section should include comprehensive guidelines to operate the application.*

This section describes how to use each Pawtect feature. Screenshots (placeholders) illustrate the interface; read the step-by-step instructions to perform tasks easily.

# Homepage Navigation:

On the Pawtect homepage or dashboard, you will see the main navigation menu (usually at the top or side of the page) with links to all sections (Adoption, Surrender, Pickup, Donations, Volunteers, Vets, Chatbot, etc.).

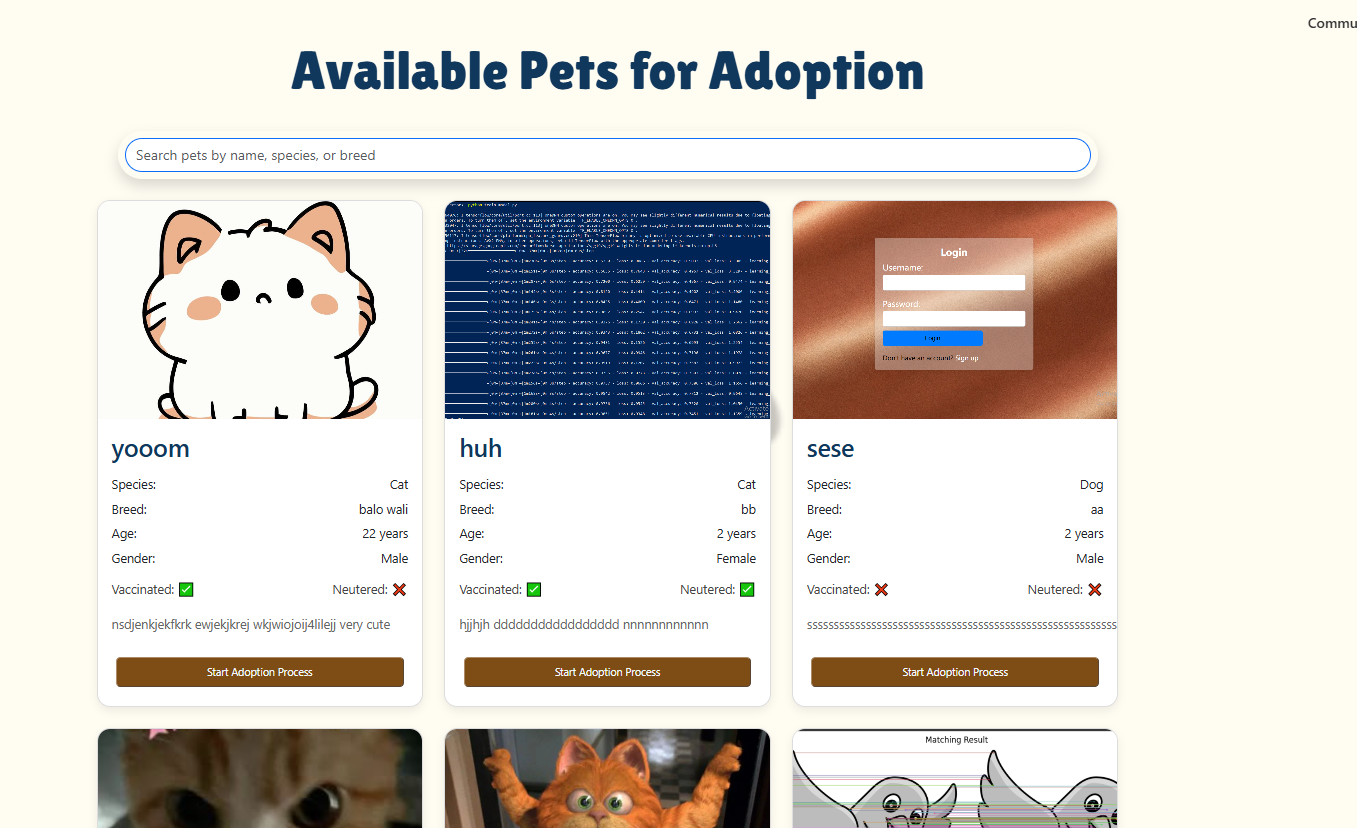
For example, click **Adopt** to view available pets, or **Donate** to support our cause.

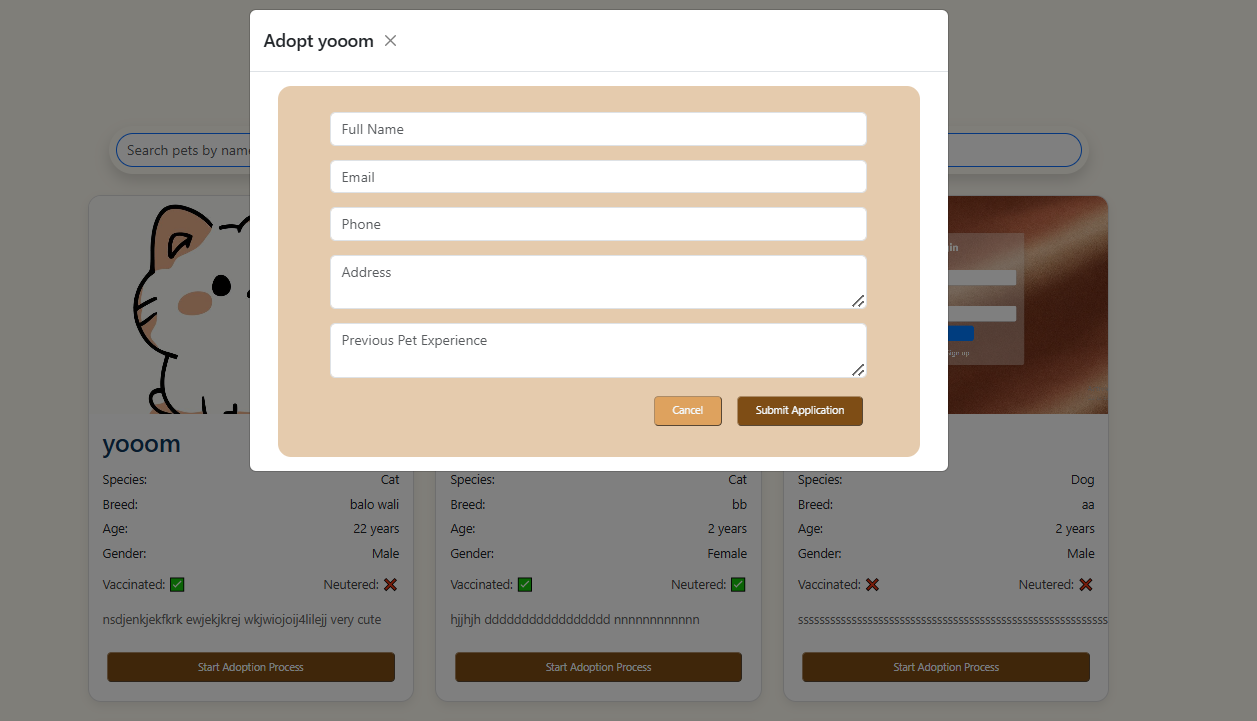


# Pet Adoption:

To adopt a pet, click the **Adopt** or **Find Pets** section.

* A catalog of available animals will appear, often filterable by species, age, or location.
* Browse the listings and click a pet’s photo or name to view details (e.g. description, age, health, shelter info).
* To proceed, click **Apply to Adopt** on the pet’s detail page and fill out the adoption form with your information and any required questionnaire
* . Submit the form.
* You can check the status of your application on your profile or via email updates.

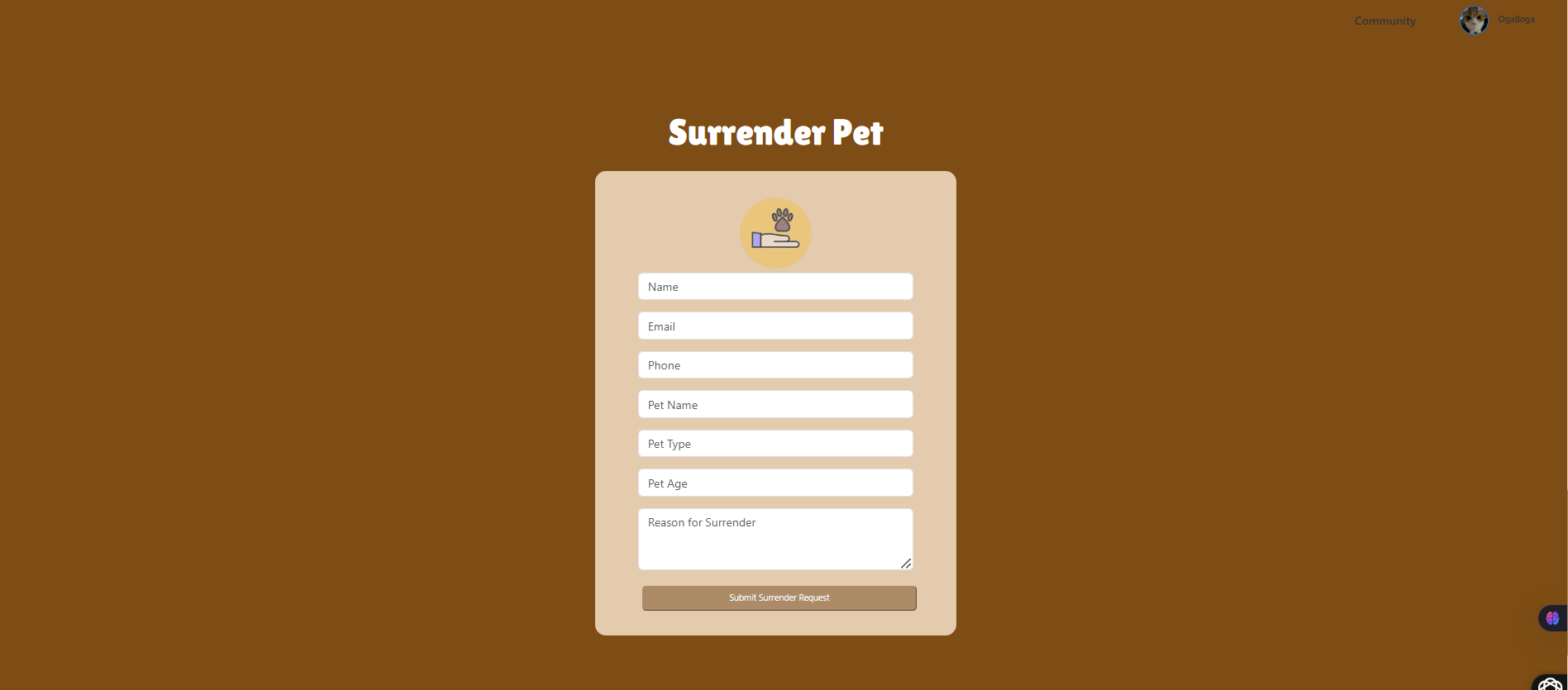




# Pet Surrender:

If you need to surrender a pet:

* go to the **Surrender Pet** section.
* Read the guidelines and click **Begin Surrender Form**.
* Enter details about the pet (species, breed, age, health history, reason for surrender) and your contact information.
* Upload a photo if possible (ensuring it meets size guidelines).
* Review the terms and submit the request.
* Pawtect staff or volunteers will contact you with next steps (e.g. shelter drop-off instructions or pickup arrangements).



### Animal Rescue:

To report a stray animal or request pickup (e.g., in a natural disaster).

* Select **Stray/Disaster Pickup**.
* You may see a form. Enter the pet’s location, description, and any urgent details.
* You can also attach a photo.
* Submit the report.
* The Pawtect rescue team or volunteers will be alerted and will coordinate pickup.
* You can view the status of your request in this section (e.g. “Volunteer assigned”, “Pickup scheduled”).



### Donations:

Visit the **Donations** page to support Pawtect.

* You may see suggested giving options (one-time, monthly, or special campaigns).
* Enter your donation amount, payment method (credit/debit or PayPal), and billing information. Review any gift designations or notes, then click
* **Donate** to complete the transaction.
* You will receive an email receipt.

[add screenshot]

### Volunteer Registration:

To volunteer, click **Volunteer** or **Get Involved**.

* You will see a volunteer application or sign-up form.
* Fill in your information (name, skills, availability, interests).
* Some tasks may allow you to choose specific roles (fostering, events, rescue team, etc.).
* Submit the form.
* The Pawtect coordinator will review and follow up by email or phone.

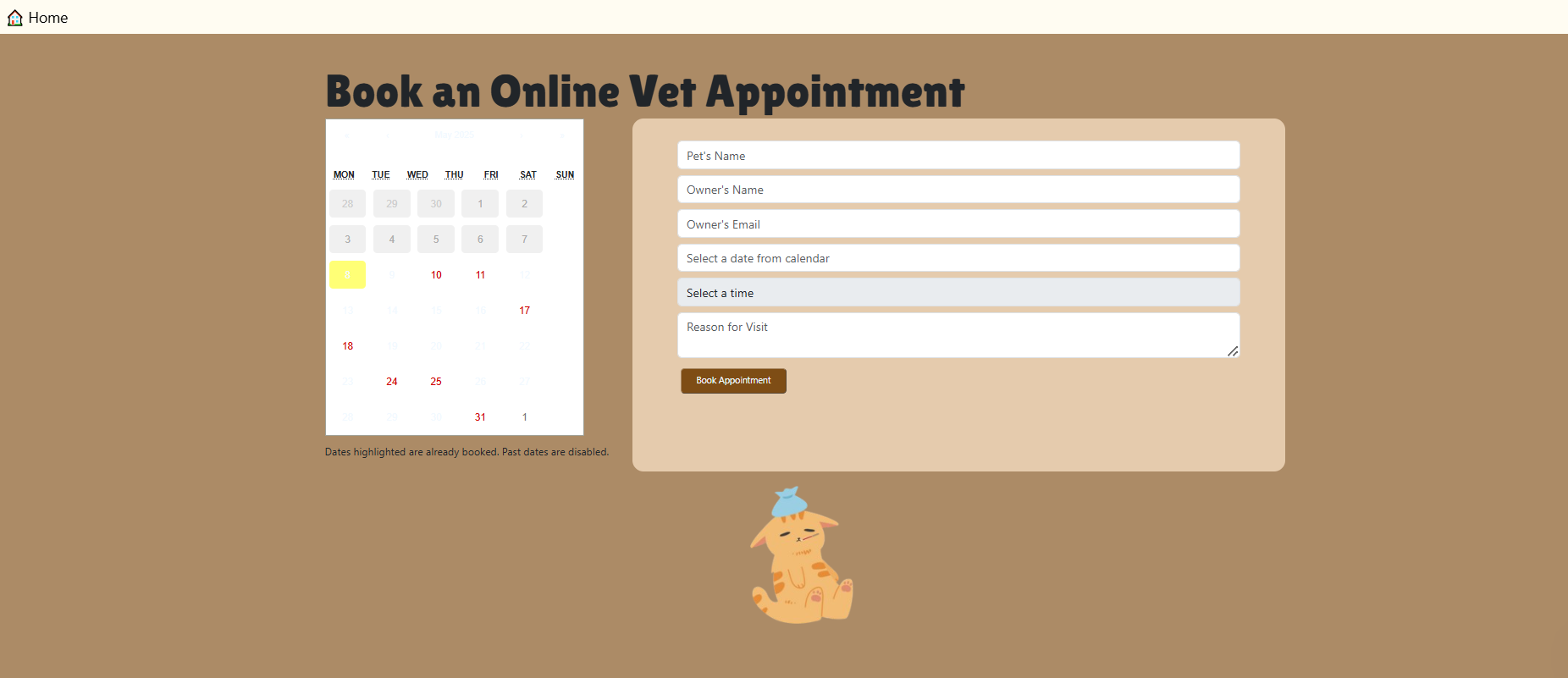
You can also browse upcoming volunteer events or opportunities on this page and sign up directly.



### Veterinary Consultations:

If Pawtect offers veterinary or behavioral consultations, go to **Vet Consult**.

* Here you can browse available vets or schedule an online consultation.
* Choose a service (e.g. health check, vaccine, behavior advice).
* You may select a date/time from a calendar.
* Enter the pet’s details and your concerns.
* Submit the request.
* The vet or veterinary partner will confirm the appointment via email.



### AI Chatbot:

The AI Chatbot is available on all pages (often as a chat icon in the corner).

* Click the chatbot icon to open the chat window. You can ask questions about Pawtect.
* for example, “How do I start the adoption process?” or “Can I donate offline?” Type your question and press Enter.
* The AI chatbot will respond with helpful information or links to the relevant sections. If the chatbot cannot answer, it may suggest contacting support.
* Use the chatbot for quick answers at any time.

