# An AI-Powered Smart Pet Care & Community Platform for Bangladesh

by

**Suraya Yeasmin**
ID: CSE2201025018

**Ataharul Islam Milon**
ID: CSE2201025046

**Mirza Sajid Arman**
ID: CSE2201025069

**Muhammad Jahid Hasan**
ID: CSE2201025078

Supervised by
**Jannatul Ferdouse**

Submitted in partial fulfillment of the requirements for the degree of
Bachelor of Science in Computer Science and Engineering


**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**
**SONARGAON UNIVERSITY (SU)**

January 2026

# APPROVAL

The project titled "**An AI-Powered Smart Pet Care & Community Platform for Bangladesh**" submitted by Suraya Yeasmin (CSE2201025018), Ataharul Islam Milon (CSE2201025046), Mirza Sajid Arman (CSE2201025069), Muhammad Jahid Hasan (CSE2201025078), Bushra Hossain (CSE2201025122) to the Department of Computer Science and Engineering, Sonargaon University (SU), has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Computer Science and Engineering and approved as to its style and contents.

**Board of Examiners**

---

**Jannatul Ferdouse**
Lecturer
Department of Computer Science and Engineering
Sonargaon University (SU)

**Supervisor**

---

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

**Examiner 1**

---

(Examiner Name and Signature)
Department of Computer Science and Engineering
Sonargaon University (SU)

**Examiner 2**

---

(Examiner Name and Signature)
Department of Computer Science and Engineering

**Examiner 3**

Sonargaon University (SU)

# DECLARATION

We, hereby, declare that the work presented in this report is the outcome of the investigation performed by us under the supervision of **Jannatul Ferdouse,** Lecturer**,** Department of Computer Science and Engineering, Sonargaon University, Dhaka, Bangladesh. We reaffirm that no part of this project has been or is being submitted elsewhere for the award of any degree or diploma.

Countersigned

---

**(Jannatul Ferdouse)**
**Supervisor**

Signature

---

**Suraya Yeasmin**
ID: CSE2201025018

---

**Ataharul Islam Milon**
ID: CSE2201025046

---

**Mirza Sajid Arman**
ID: CSE2201025069

---

**Muhammad Jahid Hasan**
ID: CSE2201025078

# ABSTRACT

Pet ownership in Bangladesh is increasing rapidly; however, pet owners face persistent challenges such as poor vaccination tracking, limited access to veterinary services, lack of reliable pet care information, and inefficient adoption and lost-pet recovery systems. **PetBondhuBD** is a smart, AI-powered mobile application designed to address these issues through a unified digital platform. The system integrates pet profile management, vaccination reminders, AI-based disease detection, emergency veterinary access, adoption services, a community forum, and a pet product marketplace. Developed using Flutter, FastAPI, Firebase, and deep learning models, the application aims to improve pet welfare, reduce health risks, and strengthen the pet-loving community in Bangladesh. This thesis presents the system design, methodology, implementation, testing, results, limitations, and future directions of the PetBondhuBD project.

# ACKNOWLEDGMENT

At the very beginning, we would like to express my deepest gratitude to the Almighty Allah for giving us the ability and the strength to finish the task successfully within the schedule time.

We are auspicious that we had the kind association as well as supervision of **Jannatul Ferdouse,** Lecturer, Department of Computer Science and Engineering, Sonargaon University whose hearted and valuable support with best concern and direction acted as necessary recourse to carry out our project.

We are particularly grateful to **Prof. Bulbul Ahamed**, Head of the Department, Computer Science and Engineering, Sonargaon University, for his kind concern and precious
suggestions.

We are also thankful to all our teachers during our whole education, for exposing us to the beauty of learning.

Finally, our deepest gratitude and love to my parents for their support, encouragement, and endless love.

# LIST OF ABBREVIATION

| | |
|---|---|
| AI | Artificial Intelligence |
| API | Application Programming Interface |
| DFD | Data Flow Diagram |
| DL | Deep Learning |
| ERD | Entity Relationship Diagram |
| ML | Machine Learning |
| GPU | Graphics Processing Unit |
| UI | User Interface |
| UX | User Experience |

# TABLE OF CONTENTS

# List of Table

# LIST OF FIGURE

# CHAPTER 1

# INTRODUCTION

## 1.1 Background of the Study

In recent years, pet ownership in Bangladesh has increased significantly, particularly in urban areas such as Dhaka, Chittagong, and Sylhet. People are increasingly viewing pets as family members rather than just domestic animals. Factors such as higher disposable incomes, changing lifestyles, exposure to social media, and growing awareness of animal welfare have contributed to this trend. Dogs, cats, birds, and small mammals are becoming common household pets, reflecting a shift in societal attitudes toward companion animals.

Despite the growing interest in pets, many owners face difficulties in managing their pets effectively. Health management is a major concern, as owners often struggle to track vaccination schedules, maintain medical records, and ensure timely medicine administration. Disease awareness is also limited; early signs of illnesses such as parvovirus, feline leukemia, or canine distemper can easily go unnoticed, leading to severe complications. Additionally, adoption processes are often inefficient due to the lack of a centralized platform, and recovering lost pets relies on informal methods such as social media, word-of-mouth, or local announcements, which are often slow and unreliable.

Existing pet care applications are either foreign-based or limited in functionality. Foreign apps may not support the Bengali language, local veterinary services, or currency, making them less suitable for Bangladeshi users. Local solutions, if available, are typically focused on single features such as vaccination reminders or basic pet listings, without providing a complete ecosystem for pet care management. This fragmented approach forces owners to use multiple platforms for different needs, reducing convenience and efficiency.

Modern technologies like mobile applications, cloud-based databases, and artificial intelligence present significant opportunities for improving pet care. AI can assist in disease detection, predictive health monitoring, and interactive guidance through chatbots. Mobile apps can centralize pet profiles, vaccination schedules, adoption listings, lost-and-found reporting, and community forums into a single accessible platform. Such a system can reduce errors, enhance awareness, and improve responsiveness for pet owners.

For Bangladesh, localization is essential. Pet owners need platforms that support the Bengali language, connect with local veterinary clinics, and reflect regional socio-economic conditions. Localization ensures that services are accessible and meaningful for users with diverse technological literacy and resources. A platform that

addresses local needs can increase adoption, improve compliance with vaccination schedules, and foster responsible pet ownership.

The PetBondhuBD project is designed to address these challenges by providing an **integrated, technology-driven solution for pet care management in Bangladesh**. The platform combines multiple functionalities: pet profile management, vaccination and medicine reminders, AI-assisted disease detection, adoption management, lost-and-found pet services, pet shop and product management, community forums, and AI-based chatbot support. By integrating these features, PetBondhuBD offers a holistic solution, reducing the need for multiple apps while improving usability, accessibility, and effectiveness.

AI is a key component of PetBondhuBD. It enables early disease detection by analyzing symptoms or images, predictive health monitoring, and real-time guidance through chatbots. This reduces reliance on in-person consultations for minor issues while ensuring users receive accurate recommendations, improving the overall quality of care for pets.

Beyond individual pet management, PetBondhuBD promotes responsible pet ownership and community engagement. Features like adoption listings, lost-and-found pet reporting, and discussion forums encourage collaboration among pet owners. By connecting users, the platform contributes to reducing stray animal populations, improving animal welfare, and building a knowledgeable and informed pet-owning community.

In summary, the rising number of pet owners in Bangladesh, combined with fragmented care services, limited awareness, and the potential of modern technology, highlights the need for an integrated pet care solution. PetBondhuBD addresses these needs by providing a comprehensive, localized, and AI-enabled platform, improving pet health management, adoption efficiency, and community engagement. This project is a timely intervention in the Bangladeshi context, offering both technological and social benefits for pets and their owners.

### 1.2 Problem Statement

#### Challenges in Veterinary Information Access

One of the main challenges faced by pet owners in Bangladesh is the difficulty in accessing **reliable veterinary information**. Many pet owners rely on informal sources, such as social media groups, word-of-mouth advice from friends, or outdated online articles, which may not be accurate or trustworthy. Without access to professional veterinary guidance, owners often make uninformed decisions regarding their pets' health, nutrition, and preventive care. This lack of structured veterinary information increases the risk of **misdiagnosis**, delayed treatment, and improper care. Furthermore, many small towns and rural areas lack sufficient veterinary clinics, limiting access to professional consultations and healthcare advice. Even in urban areas

where veterinary services exist, scheduling appointments and keeping track of health histories can be cumbersome, leading to missed treatments and irregular monitoring of pets' health.

**Absence of Centralized Health Management**

Another pressing issue is the **absence of a centralized system to manage pet health**. Owners often have multiple pets and need to track vaccinations, medications, deworming, and periodic health check-ups. Currently, there is no comprehensive platform in Bangladesh that consolidates this information for easy access. Pet owners may maintain scattered records using notebooks, calendars, or mobile notes, which can be inefficient and prone to errors. Missing a vaccination schedule or a preventive medicine dose can have serious consequences, potentially exposing pets to preventable diseases. This gap highlights the need for a digital system that can provide timely reminders and maintain a **centralized health history** for each pet, ensuring that owners can manage their pets' health proactively.

Disease awareness is another critical challenge. Early detection and timely intervention are essential for ensuring the well-being of pets, especially in the case of contagious or chronic diseases. Many owners, however, lack the knowledge to recognize early symptoms or differentiate between minor health issues and serious conditions. This problem is compounded by the limited number of veterinary specialists and diagnostic facilities in many areas of Bangladesh. Without AI-assisted guidance or predictive health tools, owners may wait until symptoms worsen before seeking professional help, reducing the chances of successful treatment. There is a clear need for a **technology-driven solution** that can analyze symptoms or behavioral patterns and provide preliminary guidance, helping owners take appropriate action before conditions escalate.

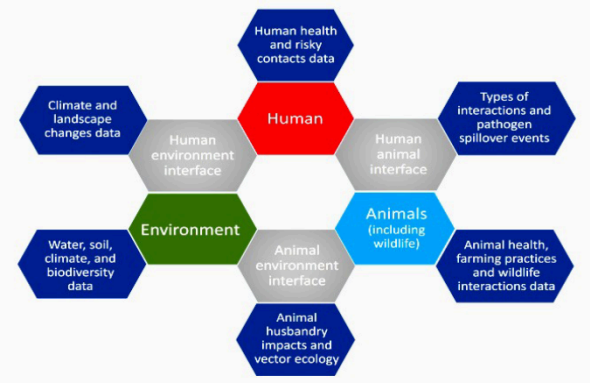


fig : Surveillance and response strategies for zoonotic diseases: a comprehensive review

Pet adoption is an essential aspect of responsible pet ownership, yet there is currently **no structured adoption platform** in Bangladesh. Many pets, particularly strays or rescued animals, remain without homes due to the lack of organized adoption channels. Individuals seeking to adopt pets often rely on scattered social media posts or word-of-mouth, which can be unreliable and inefficient. Without a verified system, the process of matching pets with suitable owners becomes time-consuming and sometimes unsafe. A centralized adoption module within a mobile application could streamline the process, allowing potential adopters to view verified pets, submit adoption requests, and communicate with relevant authorities efficiently. This feature can reduce stray populations, increase adoption rates, and ensure that pets are placed in safe and caring environments.

Lost pets are another common issue faced by pet owners. Currently, there is **no unified system in Bangladesh** to report lost or found animals. Owners may resort to posting on social media, distributing flyers, or contacting local animal welfare groups, which often leads to delays and inefficiencies. In many cases, pets are not recovered due to slow information dissemination and lack of a central repository for lost-and-found records. A digital platform with real-time reporting, notifications, and geolocation-based alerts could significantly improve the chances of reuniting lost pets with their owners. Incorporating this feature into a mobile application would address a major gap in pet care management and enhance overall animal welfare.

Pet owners in Bangladesh also face challenges in accessing **peer support and community knowledge**. While online forums and social media groups exist, they are fragmented, unstructured, and often filled with misinformation. Owners seeking advice on nutrition, training, or disease prevention may receive conflicting information, which can negatively affect their pets' health. A dedicated community module within a mobile application could provide a structured platform for users to share experiences, ask questions, and engage in discussions moderated by experts. This would not only improve knowledge sharing but also promote responsible pet ownership practices and foster a sense of community among pet lovers.

Artificial Intelligence (AI) has become an important tool in modern healthcare, including veterinary care. However, **AI-based health assistance is largely absent in Bangladesh's pet care ecosystem**. AI can analyze symptoms, predict health risks, and provide timely guidance to pet owners, acting as a supportive tool for both owners and veterinarians. Without AI integration, pet owners rely solely on manual observation and delayed consultations, which may result in missed or late diagnosis. Incorporating AI into a pet care platform could provide predictive analytics, real-time disease detection, and intelligent reminders, enabling owners to manage their pets' health more effectively and efficiently.

Despite the technological potential, the lack of **user-friendly digital platforms** in Bangladesh prevents pet owners from accessing integrated pet care services. Existing

systems are either limited to single functions, poorly designed, or require advanced technical knowledge, making them less practical for everyday use. Pet owners need a platform that is **intuitive, localized, and accessible**, with all core features available in one place. By combining health management, adoption services, lost-and-found reporting, community forums, and AI-based guidance, such a platform can significantly improve the efficiency and quality of pet care in Bangladesh.

In summary, pet owners in Bangladesh face multiple interrelated challenges: limited access to professional veterinary guidance, fragmented health management, low disease awareness, inefficient adoption and lost-and-found systems, lack of community knowledge sharing, absence of AI-based health assistance, and unavailability of comprehensive, user-friendly digital platforms. These gaps collectively hinder effective pet care, compromise animal welfare, and reduce opportunities for responsible pet ownership.

The PetBondhuBD project is designed to address all of these challenges by developing a **comprehensive mobile-based application** that integrates multiple features into a single platform. By providing health management tools, vaccination reminders, AI-based disease detection, adoption services, lost-and-found reporting, community interaction, and user-friendly interfaces, PetBondhuBD aims to fill the current gaps in Bangladesh's pet care ecosystem. The platform will ensure that pet owners can manage their pets' health and welfare efficiently, improve disease prevention, streamline adoption processes, and foster a supportive community, ultimately contributing to better animal welfare and responsible pet ownership in Bangladesh.

### 1.3 Objectives of the Project

The main objectives of the PetBondhuBD project are designed to address the gaps in pet care management in Bangladesh and leverage technology to provide an **all-in-one solution** for pet owners, veterinary professionals, and the community. The objectives include:

1. **To develop a centralized digital platform for pet care management**
    - Create a single, unified system where all pet-related services, such as health tracking, adoption, lost-and-found reporting, and community engagement, are accessible through one application.
2. **To provide pet health tracking, vaccination, and medicine reminders**
    - Enable users to maintain digital health records for each pet, schedule vaccination alerts, track medical treatments, and ensure timely administration of medicines to prevent diseases.

3. **To integrate AI-based pet disease detection and chatbot assistance**
    - Implement AI models capable of analyzing symptoms, images, or behavior patterns to predict potential diseases.
    - Provide a chatbot that can answer user queries, give health tips, and assist owners in routine pet care management.
4. **To facilitate pet adoption and lost-and-found services**
    - Establish a verified adoption system where potential adopters can connect with pets in need of homes.
    - Introduce a lost-and-found reporting module with notifications, geolocation, and real-time updates to increase the chances of reuniting pets with their owners.
5. **To create a community-driven platform for pet lovers and experts**
    - Enable interaction between pet owners, veterinarians, and animal welfare experts.
    - Promote knowledge sharing, awareness campaigns, training tips, and discussions on responsible pet ownership.
6. **To improve awareness and education regarding pet health and welfare**
    - Offer educational content on nutrition, disease prevention, training, and emergency care.
    - Encourage responsible pet ownership practices among users of all ages.
7. **To enhance accessibility and localization of pet care services**
    - Support Bengali language interfaces and locally relevant content, making the platform easy to use for all demographic groups in Bangladesh.
8. **To provide a platform for integration with local veterinary services and pet shops**
    - Enable users to access local veterinarians, book appointments, purchase products, and obtain services seamlessly through the platform.
9. **To implement predictive analytics for proactive pet health management**
    - Use data collected from pets' health records to identify potential risks and provide early recommendations, reducing the likelihood of severe diseases.

10. **To ensure scalability and adaptability for future enhancements**
    - Design the system architecture to accommodate future features such as tele-veterinary consultations, wearable device integration, multi-language support, and e-commerce analytics.
11. **To contribute to animal welfare and community engagement**
    - Reduce stray animal populations through organized adoption programs.
    - Build a supportive ecosystem where pet lovers can collaborate on welfare initiatives, rescue operations, and public awareness campaigns.

### 1.4 Motivation and Significance

The motivation behind the development of PetBondhuBD stems from the increasing number of pet owners in Bangladesh and the growing awareness of the need for **responsible pet ownership**. In recent years, urban areas such as Dhaka, Chittagong, and Sylhet have seen a surge in pet adoption, reflecting changing social attitudes towards pets as integral members of the family. Despite this positive trend, pet owners often face difficulties in managing pet health, ensuring timely vaccinations, monitoring diseases, and accessing reliable guidance. These challenges highlight a significant gap in the pet care ecosystem, motivating the need for a **comprehensive and integrated digital solution**.

A major motivating factor for PetBondhuBD is the **lack of centralized platforms** in Bangladesh that combine health management, adoption services, lost-and-found reporting, and community interaction in a single system. Existing applications are either foreign-based or limited in functionality, which restricts accessibility for the majority of local users. Additionally, most solutions do not provide **AI-based assistance**, predictive health monitoring, or user-friendly interfaces, which are essential for modern, proactive pet care. Recognizing these gaps, PetBondhuBD was conceptualized to bridge the divide between technology and pet welfare, providing a **holistic approach** that integrates multiple features tailored to the Bangladeshi context.

The significance of PetBondhuBD is multifaceted. From a **technical perspective**, the system leverages modern technologies such as **mobile application development, AI-driven disease detection, and chatbot-based guidance** to provide intelligent support for pet owners. The AI component enhances early disease detection, reduces reliance on manual monitoring, and helps owners take timely preventive measures. By maintaining comprehensive digital health records and providing automated reminders for vaccinations and medications, the platform minimizes the risk of missed treatments and supports better health outcomes for pets.

From a **social perspective**, the platform promotes **responsible pet ownership** and animal welfare. Features such as adoption management, lost-and-found reporting, and

community forums encourage active engagement among pet owners and experts. This not only facilitates the placement of stray and rescued animals in loving homes but also increases community awareness about proper pet care, nutrition, and preventive measures. By connecting pet owners with veterinary professionals, PetBondhuBD also enhances the efficiency of healthcare services, ensuring that pets receive timely and appropriate medical attention.

Moreover, the platform emphasizes **localization and accessibility**, making it suitable for users across different regions of Bangladesh. With Bengali language support and locally relevant content, the system ensures that even users with limited technical knowledge can benefit from its features. This focus on accessibility enhances the overall adoption and usability of the platform, which is crucial for achieving wide-reaching social and health impacts.

In addition, PetBondhuBD holds **future significance** by providing a scalable and adaptable framework that can accommodate advanced features such as tele-veterinary consultations, smart wearable integration, multi-language support, and e-commerce functionalities. Such scalability ensures that the platform remains relevant and can evolve alongside technological advancements and changing user needs.

In summary, the **motivation** for developing PetBondhuBD arises from the pressing need to improve pet health management and promote responsible ownership in Bangladesh, while its **significance** lies in the platform's potential to enhance pet care efficiency, support veterinary professionals, foster community awareness, and contribute to a smarter, more humane, and technology-driven pet care ecosystem. By bridging gaps in healthcare, communication, and adoption services, PetBondhuBD serves as a comprehensive solution with meaningful technical, social, and long-term impacts.

### 1.5 Scope of the Project

The scope of this project includes pet profile management, health monitoring, vaccination reminders, AI-based disease detection, adoption services, lost-and-found reporting, and an online pet shop. The system is designed as a mobile application with an admin dashboard for management purposes. However, advanced clinical diagnosis and real-time veterinary consultation are beyond the current scope and are considered for future enhancements.

### 1.6 Organization of the Thesis

This thesis is organized into ten chapters. Chapter 1 introduces the background, problem, objectives, motivation, scope, and structure of the study. Chapter 2 reviews relevant literature and existing systems. Chapter 3 presents the system overview, while Chapter 4 discusses requirement analysis. Chapters 5 and 6 describe system design and methodology. Chapter 7 focuses on implementation, Chapter 8 covers testing and results, Chapter 9 discusses limitations, and Chapter 10 concludes the thesis with future work directions.

# CHAPTER 2
# LITERATURE REVIEW

## 2.1 Overview of Existing Pet Care Applications

Several pet care applications such as Pawtrack, PetDesk, and 11pets have been developed to assist pet owners with basic pet management services. These applications mainly focus on pet profile creation, vaccination reminders, appointment scheduling, and basic health tracking. Although these platforms are useful, most of them are designed for global users and lack localization for developing countries like Bangladesh. Additionally, many applications do not provide integrated features such as AI-based disease detection, adoption services, or community interaction in a single platform.

## 2.2 Review of Health Management Systems for Pets

Pet health management systems are designed to monitor animal health records, vaccination schedules, and medical histories. Previous studies highlight the importance of digital health records in improving preventive care and reducing disease risks. Some systems allow veterinarians to store and access pet medical data electronically. However, these systems often require clinical infrastructure and are not easily accessible to general pet owners. Moreover, limited automation and lack of intelligent decision-support systems reduce their effectiveness.

## 2.3 AI-Based Disease Detection in Animals

Recent research shows that artificial intelligence and machine learning techniques are increasingly used in animal disease detection. Image-based disease recognition, symptom-based prediction models, and chatbot-driven diagnostics have shown promising results in early disease identification. Studies indicate that AI can assist veterinarians by reducing diagnostic time and improving accuracy. However, most AI-based systems are experimental, require large datasets, and are rarely implemented in user-friendly mobile applications for everyday pet owners.

### 2.4 Adoption, Lost & Found, and Community Platforms

Digital platforms for pet adoption and lost-and-found services play a significant role in animal welfare. Online portals and social media groups are commonly used to report lost pets and promote adoption. While these platforms help connect pet owners and adopters, they often lack structured verification, tracking, and notification systems. Community forums exist but are usually separate from health and management tools, resulting in fragmented services for pet lovers.

### 2.5 Research Gap and Justification

The literature review reveals a clear gap in existing pet care systems. Most applications focus on isolated functionalities such as health reminders or adoption services but fail to provide a comprehensive, AI-integrated solution. There is also a lack of localized platforms suitable for Bangladesh that consider cultural, economic, and technological constraints. Therefore, the PetBondhuBD project is justified as it aims to develop an all-in-one, intelligent, and localized pet care platform that integrates health management, AI-based disease detection, adoption services, and community engagement in a single system.

# CHAPTER 3
## SYSTEM OVERVIEW

### 3.1 System Architecture Overview

The PetBondhuBD system is designed using a modular and scalable client–server architecture. The frontend of the application is developed using Flutter, enabling cross-platform support for Android and iOS devices. The backend is implemented using FastAPI, which handles API requests, authentication, data processing, and system logic. A centralized database is used to store user information, pet profiles, health records, adoption data, and community posts. Additionally, AI modules are integrated into the system for pet disease detection and chatbot-based pet care assistance. This architecture ensures efficiency, flexibility, and ease of future expansion.

### 3.2 User Roles and Access Levels

The system supports role-based access control to ensure secure and organized operation. There are mainly two types of users: general users (pet owners) and administrators. General users can register, log in, manage pet profiles, access health features, post adoption or lost-and-found information, participate in community discussions, and use the AI chatbot. Administrators have higher privileges, including managing users, monitoring content, approving posts, handling product listings, and maintaining overall system functionality. This separation of access levels enhances system security and management efficiency.

### 3.3 Functional Overview of PetBondhuBD

PetBondhuBD provides an integrated set of features designed to support complete pet care management. The system allows users to create and manage pet profiles, receive vaccination and medicine reminders, and use AI-based disease detection for preliminary health analysis. Users can also participate in pet adoption processes, report lost or found pets, purchase pet-related products, and engage in community knowledge sharing through discussion forums. An AI chatbot is included to provide instant guidance on pet care-related queries. These functionalities work together to offer a comprehensive and user-friendly pet care solution.

### 3.4 Application Workflow Summary

The application workflow of PetBondhuBD illustrates the step-by-step interaction between users and the system. From initial login to accessing different services, the workflow ensures smooth navigation, secure data handling, and efficient feature execution. The structured workflow helps users easily understand and utilize the system functionalities.

Table 3.1: Application Workflow Summary

| Step No. | User Activity | System Operation | Result |
|---|---|---|---|
| 1 | User opens the application | System loads initial interface | Home/Login screen displayed |
| 2 | User registers or logs in | Authentication and validation | Access to user dashboard |
| 3 | User adds or updates pet profile | Data stored in database | Pet profile created/updated |
| 4 | User uses health management features | AI analysis and reminder setup | Health status and alerts |
| 5 | User accesses adoption or lost & found | Data verification and posting | Public listing displayed |
| 6 | User participates in community forum | Content processing and storage | Posts and discussions visible |
| 7 | Admin monitors system activities | Data management and moderation | Updated system records |

This workflow ensures that all system components function in a coordinated and efficient manner.

# CHAPTER 4
# REQUIREMENT ANALYSIS

This chapter describes the functional and non-functional requirements of the PetBondhuBD system. Requirement analysis helps in understanding what the system should do and the constraints under which it must operate to meet user expectations.

## 4.1 Functional Requirements – Detailed Explanation

What are Functional Requirements?

Functional requirements describe what a system is supposed to do. They define the main features, services, and behaviors of an application. These requirements focus on the actions the system must perform in response to user inputs.

In simple terms, functional requirements answer the question:

“What functions can users perform using the system?”

Explanation of the Given Text

“Functional requirements define the core functionalities and services provided by the PetBondhuBD application.”

This sentence means that functional requirements identify the essential features and services that the PetBondhuBD application offers. These features represent the main purpose of the application and define what the system is capable of doing.

For the PetBondhuBD application, core functionalities may include:

User registration and login

Creating and managing pet profiles

Searching for veterinarians and pet services

Booking veterinary appointments

Posting and viewing pets for adoption

Receiving notifications and alerts

These functionalities are critical because without them, the application would not fulfill its intended goals.

“These requirements specify how users interact with the system”

This part explains that functional requirements clearly describe how users communicate with and use the system. It defines the steps a user follows to complete a task and how the system responds.

User interactions include:

Entering login credentials

Filling out forms to add pet information

Clicking buttons or selecting options

Submitting requests such as appointment bookings

Viewing confirmation messages or notifications

Functional requirements ensure that every user action has a defined system response.

“and how different modules perform their operations”

This statement indicates that functional requirements also describe how the internal components (modules) of the system operate to support user actions.

A module is a functional unit of the system designed to handle specific tasks.

Examples of modules in the PetBondhuBD application include:

User Management Module: Handles registration, login, and user profiles

Pet Profile Module: Stores and updates pet information

Appointment Management Module: Manages booking, updating, and canceling appointments

Adoption Module: Handles pet adoption listings and searches

Notification Module: Sends alerts, reminders, and confirmations

Each module has defined responsibilities, and functional requirements explain how these modules process data and interact with each other to complete system operations.

Summary

Functional requirements:

Define the main features and services of the PetBondhuBD application

Describe how users interact with the system

Explain how different system modules perform their tasks

They provide a clear understanding of what the system does, ensuring proper design, development, and testing of the application.

#### 4.1.1 User Login & Registration System

The system shall provide a secure user registration and login facility. New users must be able to create accounts using valid credentials, while existing users can log in using authenticated information. The system shall validate user input, manage sessions, and ensure secure access to user-specific features.

#### 4.1.2 Pet Profile Management

This feature provides users with a comprehensive set of tools to **manage digital records for their pets**, ensuring all essential information is organized, accessible, and up-to-date.

### Key Capabilities

The system supports the full **CRUD (Create, Read, Update, Delete)** lifecycle for pet profiles:

- **1. Create Pet Profile:** Users can input details to establish a new profile for each pet.
- **2. View Pet Profile:** Users can access and display all stored information for a specific pet.
- **3. Update Pet Profile:** Users can modify existing information as needed (e.g., updating age, adding new vaccination records, changing health notes).
- **4. Delete Pet Profile:** Users can permanently remove a pet's profile from the system.

### Essential Information Fields

Each pet profile is designed to store the following critical data points:

| Field | Description | Example |
|---|---|---|
| Pet Name | The given name of the pet. | Buddy |

| | | |
|---|---|---|
| Species | The type of animal (e.g., Dog, Cat, Bird, Reptile). | Cat |
| Breed | The specific breed of the pet. | Siamese |
| Age | The pet's age (often stored as a birth date for calculation). | 5 years |
| Gender | The biological sex of the pet. | Female |
| Vaccination History | A chronological record of all administered vaccines (Name, Date, Vet). | Rabies (2025-01-15) |
| Health Notes | Free-text field for storing important medical information, allergies, or behavioral notes. | Allergic to chicken, annual check-up due in October. |

**User Benefit**

This feature is crucial for **maintaining organized and accessible pet records**. It allows users to quickly retrieve vital information, especially during veterinary visits or in emergencies, promoting better pet healthcare management.

### 4.1.3 Vaccination & Medicine Reminder System

This system is designed to be a proactive assistant for pet owners, ensuring their beloved companions receive timely vaccinations and medications. It leverages automated reminders to help users stay on top of their pet's healthcare schedule, promoting consistent preventive care and overall well-being.

## Core Functionality

The system revolves around three main actions:

1. **Scheduling Reminders:** Users can input specific details for each vaccination or medication, including the type, date/time, and recurrence.
2. **Automated Notifications:** Based on the set schedules, the system will send timely alerts to the user.
3. **Record Keeping:** All scheduled and completed reminders are logged, providing a history of treatments.

## Detailed Features and User Experience

1. Setting Up a New Reminder

When a user wishes to add a new reminder, they will navigate to a dedicated section within the pet's profile. Here, they will be prompted to enter key information.

**User Interface for Adding a Reminder:**

- **Reminder Type Selection:** Users can choose between "Vaccination" or "Medication." This might slightly alter subsequent fields (e.g., a "Dose" field might be more prominent for medication).
- **Vaccine/Medicine Name:** A free-text input field for the specific name (e.g., "Rabies Vaccine," "Flea & Tick Preventative," "Heartworm Medication"). Autocomplete suggestions could be implemented based on common treatments.
- **Date & Time:** A date picker and time selector for the initial or next due date and time.
- **Repeat Schedule:** Options for recurrence (e.g., "Daily," "Weekly," "Monthly," "Yearly," "Every X Weeks/Months," or "One-time"). This is crucial for ongoing medications and annual vaccinations.
- **Dose/Notes:** A text area to add specific instructions, dosage information (e.g., "1 tablet with food," "2 drops in left eye"), or any other relevant details.
- **Attach Vet Record (Optional):** Users can upload a file (e.g., PDF of a vaccination certificate or vet prescription) to keep all relevant documents in one place.

2. Automated Notifications

Once a reminder is set, the system will actively monitor the schedule. When a reminder is due, a notification will be triggered.

**Example of a System Notification:**

- **Push Notifications:** Delivered to the user's mobile device or desktop.
- **Email/SMS (Optional):** Users might have the option to receive reminders via email or text message for critical alerts.
- **Notification Content:** Clearly states the pet's name, the type of reminder (vaccination/medicine), the specific item (e.g., "Rabies Vaccine"), and the due time.
- **Actionable Buttons:**
    - **"Snooze":** Allows the user to postpone the reminder for a short period (e.g., 10 minutes, 1 hour).
    - **"Mark as Done":** Records the completion of the task and, for recurring reminders, automatically schedules the next instance based on the repeat settings.

3. Reminder History and Management

Users need a central place to view all upcoming and past reminders. This dashboard provides an overview and allows for modifications.

**Reminder List/Dashboard:**

- **Upcoming Reminders View:** Displays a chronological list of all pending vaccinations and medications across all pets.
- **Filtering Options:** Allows users to filter by "Vaccinations," "Medication," or view "All."
- **Pet-Specific View:** Reminders can also be viewed from within each pet's profile.
- **History Tab:** A dedicated section to view all completed reminders, serving as a comprehensive health log. This history should include the date/time of completion and possibly who marked it as done.
- **Edit/Delete:** Users can easily modify or remove existing reminders.

## Technical Considerations

- **Database Integration:** Reminders will be linked to specific pet profiles and stored in the database.
- **Scheduling Engine:** A robust background process is needed to accurately trigger notifications at the scheduled times, handling recurrence logic.

- **Notification Service:** Integration with platform-specific notification services (e.g., Firebase Cloud Messaging for mobile, browser push API for web).
- **Time Zone Handling:** The system must correctly account for different time zones to ensure reminders are delivered at the appropriate local time.

**Benefits**

- **Improved Pet Health:** Ensures no crucial vaccinations or medication doses are missed.
- **Peace of Mind for Owners:** Reduces stress and the cognitive load of remembering complex schedules.
- **Better Record Keeping:** Centralizes health event data, useful for vet visits.
- **Preventive Care Promotion:** Encourages consistent healthcare habits.

#### 4.1.4 AI-Based Pet Disease Detection Module

The system shall include an AI-based module to assist users in identifying potential pet diseases. Users can input symptoms or upload images, and the system will analyze the data to provide preliminary disease suggestions. This feature supports early awareness but does not replace professional veterinary diagnosis.

#### 4.1.5 Adoption Management System

The system shall support pet adoption by allowing users to post pets for adoption and browse available adoption listings. Admin verification will ensure authenticity and safety. This module promotes responsible pet ownership and animal welfare.

#### 4.1.6 Lost & Found Pet Module

The system shall enable users to report lost or found pets by submitting relevant information and images. The system will display these posts publicly to help reunite pets with their owners. This module improves the chances of recovering lost pets efficiently.

#### 4.1.7 Pet Shop & Product Management System

The system shall provide an online pet shop where users can browse and purchase pet-related products. Admin users can manage product listings, prices, and availability. This feature adds convenience and supports pet care through easy access to essential products.

### 4.1.8 Community Forum & Knowledge Sharing

The system shall include a community forum where users can post questions, share experiences, and exchange knowledge related to pet care. This feature encourages interaction among pet owners and builds a supportive pet care community.

### 4.1.9 AI Chatbot for Pet Care Assistance

The system shall provide an AI-powered chatbot to answer common pet care queries. The chatbot will offer instant guidance based on predefined knowledge and AI models. This feature enhances user support and improves user engagement.

### 4.1.10 Admin Dashboard Management

The system shall include an admin dashboard for system administrators. Admins can manage users, monitor content, approve adoption posts, control product listings, and oversee overall system activities. This ensures smooth operation and system integrity.

## 4.2 Non-Functional Requirements

Non-functional requirements define the quality attributes of the system. The PetBondhuBD system shall ensure high performance, usability, security, scalability, and reliability. The system should provide a user-friendly interface, fast response time, data privacy, secure authentication, and stable operation under different usage conditions.

## 4.3 Hardware and Software Requirements

Hardware Requirements:

- Smartphone or tablet (Android/iOS) for users
- Server or cloud infrastructure for backend services
- Minimum 4 GB RAM and stable internet connection

Software Requirements:

- Flutter for frontend development
- FastAPI for backend development
- Database system (MySQL / PostgreSQL)
- AI and Machine Learning libraries
- Operating System: Windows, Linux, or macOS

# CHAPTER 5
# SYSTEM DESIGN

This chapter describes the overall design of the PetBondhuBD system. It explains the system structure, data flow, database organization, and user interface concepts that collectively ensure efficient system functionality.

## 5.1 Overall System Design

The PetBondhuBD system is designed using a modular and layered architecture. The system consists of three major layers: presentation layer, application layer, and data layer. The presentation layer includes the Flutter-based mobile interface used by pet owners and administrators. The application layer contains the backend services developed using FastAPI, which handle business logic, authentication, AI processing, and API communication. The data layer stores all system data such as user information, pet profiles, health records, adoption details, and community posts in a centralized database. This layered design improves scalability, maintainability, and security.

## 5.2 Flowchart Description

The system flowchart represents the logical flow of operations within the PetBondhuBD application. The process begins when a user opens the application and proceeds to login or registration. After successful authentication, the user is directed to the dashboard, from where different modules such as pet profile management, health monitoring, adoption, lost-and-found, pet shop, and community forum can be accessed. Each action is processed by the backend and stored in the database. Admin users follow a similar flow with additional management privileges. The flowchart ensures a clear understanding of system navigation and decision-making processes.

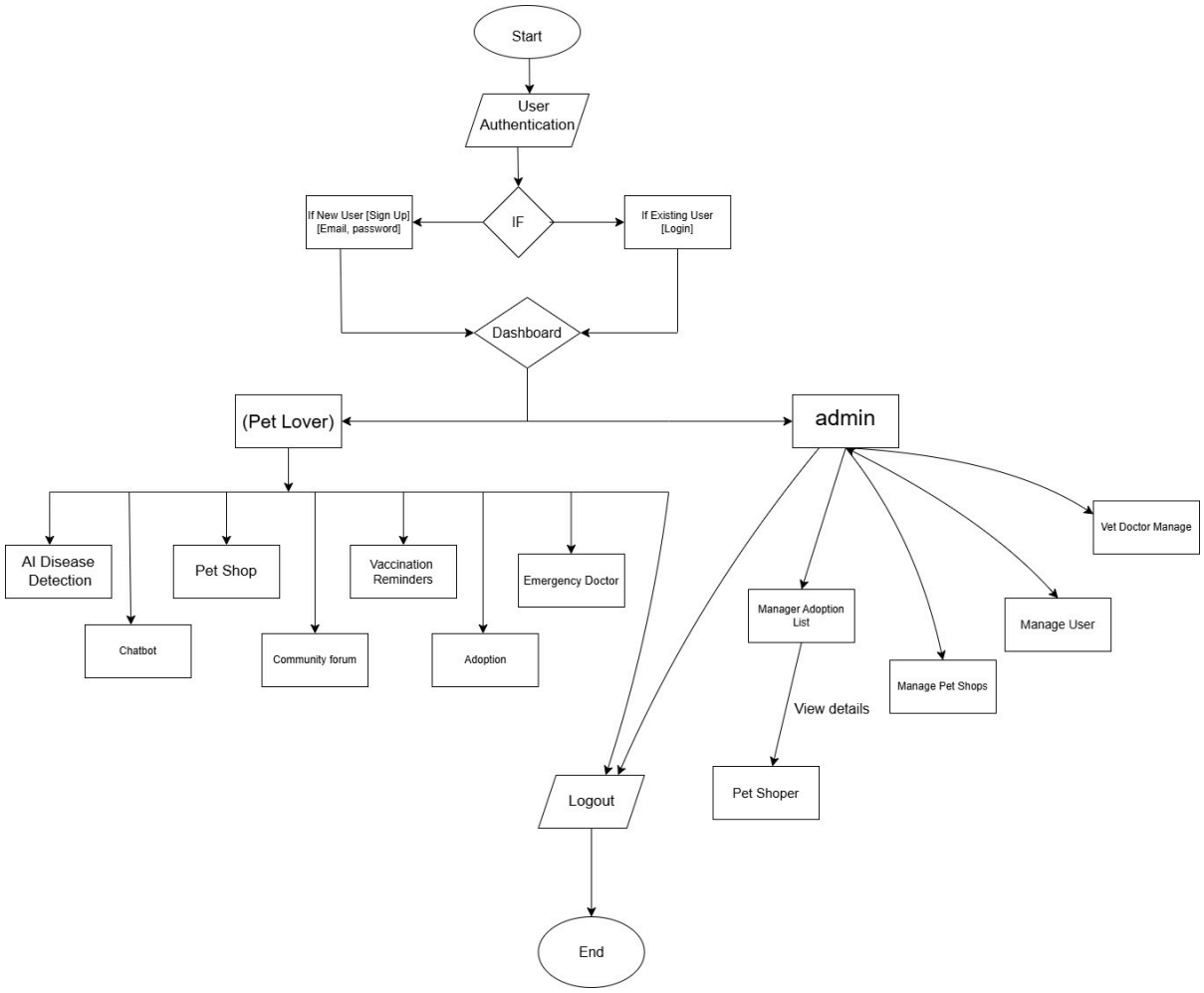


**Figure: Flowchart Description**

## 5.3 Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) illustrates how data moves through the PetBondhuBD system. It identifies external entities, processes, data stores, and data flows that define system operations. DFDs help visualize how user input is transformed into meaningful system outputs.

### 5.3.1 Level 0 DFD

The Level 0 DFD, also known as the context diagram, presents a high-level overview of the PetBondhuBD system as a single process. External entities such as users and administrators interact with the system by providing input data like login credentials, pet information, and service requests. The system processes this data and returns

outputs such as notifications, health analysis results, adoption listings, and community updates. This level shows the system boundary and overall interaction without internal details.

### 5.3.2 Level 1 DFD

The Level 1 DFD provides a more detailed view by decomposing the main system into sub-processes. These processes include user authentication, pet profile management, health monitoring, AI disease detection, adoption management, lost-and-found handling, product management, and community interaction. Each process exchanges data with relevant data stores such as user database, pet database, health records, and adoption listings. This level explains how different modules work together within the system.

## 5.4 Entity Relationship Diagram (ERD)

The Entity Relationship Diagram (ERD) defines the logical structure of the database by illustrating entities and their relationships. Key entities in the PetBondhuBD system include User, Pet, HealthRecord, Adoption, LostFound, Product, Order, CommunityPost, and Admin. Relationships between these entities ensure data consistency and integrity. The ERD helps in designing an efficient and normalized database structure.

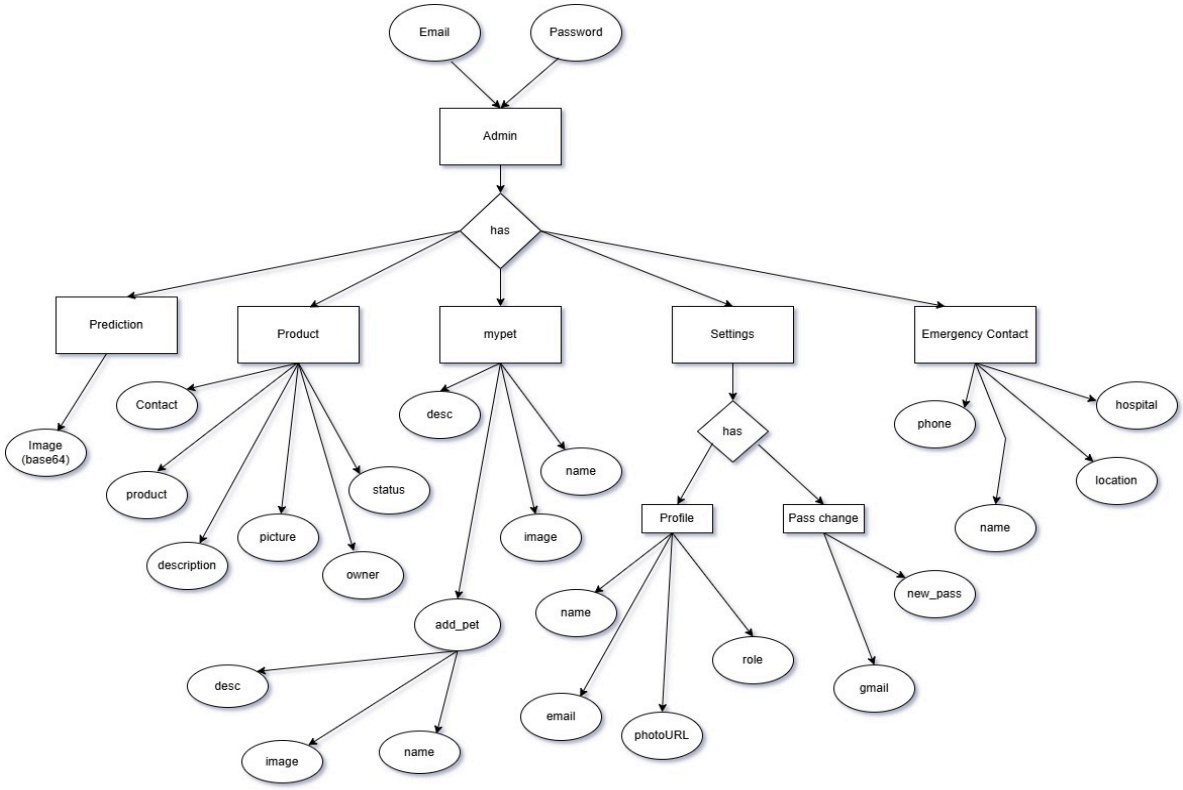


**Figure:Entity Relationship Diagram (ERD)**

## 5.5 Database Design Overview

The database design of PetBondhuBD follows a relational model to ensure efficient data storage and retrieval. Separate tables are created for users, pets, health records, reminders, adoption posts, lost-and-found reports, products, orders, and community discussions. Primary and foreign keys are used to maintain relationships between tables. This structured design supports data security, reduces redundancy, and enhances system performance.

## 5.6 User Interface Design Concept

The user interface of PetBondhuBD is designed with a focus on simplicity, usability, and accessibility. The application uses a clean layout, intuitive navigation, and consistent color schemes to enhance user experience. Different dashboards are designed for general users and administrators. Important features such as pet health status, reminders, and notifications are easily accessible. The UI design ensures that users with minimal technical knowledge can efficiently use the application.

# CHAPTER 6
# METHODOLOGY

This chapter explains the development approach, technologies used, design principles, and security considerations followed during the development of the PetBondhuBD system.

## 6.1 Development Methodology (Agile Model)

The PetBondhuBD system adopts the **Agile Development Methodology** to manage the project lifecycle. This choice prioritizes **flexibility, rapid adaptation to change, and continuous delivery** of valuable features.

### Core Principles of Agile

- **Iterative Development:** The system is built and enhanced incrementally rather than all at once.
- **Continuous Feedback:** User and stakeholder feedback is gathered and incorporated regularly throughout the project.
- **Adaptability:** The methodology is inherently flexible, allowing the team to quickly respond to new or changing requirements without major disruption.

### The Sprint Cycle

The entire development process is broken down into small, fixed-length cycles known as **Sprints** (typically 2-4 weeks). Each sprint encompasses all necessary phases to deliver a potentially shippable increment of the product:

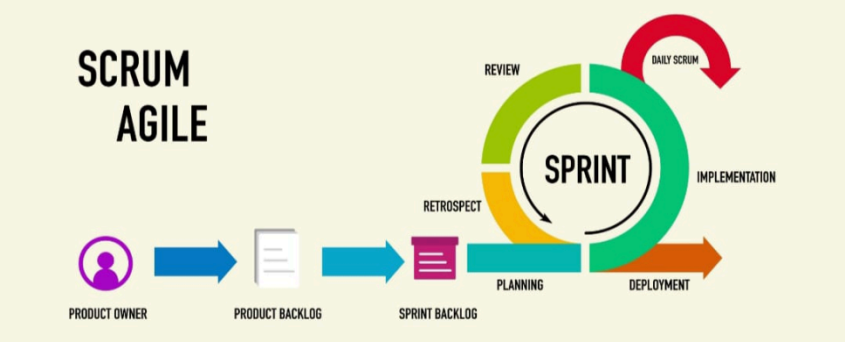


Fig: Scrum Agile

**Sprint Cycle Table :**

| Sprint Phase | Description | Key Outcome |
|---|---|---|
| Planning | The team determines which features (User Stories) from the product backlog will be built in the current sprint. | Sprint Backlog (Committed work) |
| Design | Detailed technical design and architecture work for the selected features. | Technical Specifications |
| Development | The features are coded and implemented by the development team. | Code Implementation |
| Testing | Quality assurance (QA) is performed to identify and fix bugs, ensuring feature quality. | Verified Features |
| Review & Retrospective | The team demonstrates the completed features to stakeholders (Review), and reflects on the process to identify improvements (Retrospective). | Accepted Features & Process Improvements |

### Rationale for Choosing Agile

The Agile methodology is highly suitable for PetBondhuBD because it:

- Allows for **early identification of issues** through constant testing and review.
- **Improves collaboration** between the development team, stakeholders, and future users.
- Ensures a **gradual, user-centered enhancement** of system features, keeping the system relevant to user needs.

## 6.2 Technology Stack Description

The PetBondhuBD application utilizes modern and efficient technologies to ensure performance and scalability. Flutter is used for frontend development to support cross-platform mobile applications. FastAPI is used for backend development due to its high performance and ease of API integration. A relational database system is used for storing user data, pet profiles, health records, and transaction details. Artificial Intelligence and Machine Learning libraries are integrated for disease detection and chatbot functionalities. This technology stack ensures reliability, maintainability, and future extensibility of the system.

### Technology Stack Summary Table :

| Component | Technology | Rationale & Key Features |
|---|---|---|
| Frontend (Cross-Platform Mobile App) | Flutter (Dart) | Single codebase for iOS and Android, fast development, expressive and flexible UIs, and strong performance. |
| Backend (API & Business Logic) | FastAPI (Python) | High performance (speed comparable to Node.js and Go), automatic interactive API documentation (Swagger/OpenAPI), and robust data validation with Pydantic. |

| | | |
|---|---|---|
| Database (Relational Data Storage) | PostgreSQL/MySQL (TBD) | Chosen for data integrity, complex querying capabilities (for reporting and filtering), and reliability in storing structured data (profiles, transactions, records). |
| AI/ML Integration | Python Libraries (e.g., TensorFlow, PyTorch, Scikit-learn) | |

**Detailed Rationale**

- **Cross-Platform Efficiency: Flutter** dramatically reduces development time and cost by maintaining a single codebase for both major mobile operating systems, ensuring a consistent user experience.
- **Performance and Scalability: FastAPI**, built on modern Python standards, provides the high throughput needed to handle simultaneous user requests, making the backend reliable and scalable.
- **Data Reliability:** A **Relational Database** ensures that critical and highly structured data—like user details, pet profiles, vaccination records, and transaction history—is stored with strong transactional integrity and consistency.
- **Future-Proofing:** The integration of **AI/ML libraries** allows the system to implement advanced features like predictive health modeling and intelligent user interaction, positioning PetBondhuBD for future growth and innovation.

### 6.3 Design Principles and Standards

The system design follows standard software engineering principles such as modularity, reusability, and scalability. The user interface design follows usability and accessibility guidelines to ensure a smooth user experience. Coding standards and best practices are maintained to improve code readability and maintainability. Proper documentation and version control are also followed throughout the development process. These principles ensure a well-structured and high-quality software system.

## 1. Software Engineering Principles

**Software Engineering Principles Table :**

| Principle | Description | Benefit to the System |
|---|---|---|
| Modularity | The system is divided into separate, independent modules (e.g., Pet Profiles, Reminders, Chatbot, Marketplace). | Facilitates parallel development, easier debugging, and simplifies future updates/feature additions. |
| Reusability | Components, functions, and libraries are designed to be used across multiple parts of the application. | Reduces code duplication, speeds up development, and ensures consistency across the platform. |
| Scalability | The architecture is built to handle increasing user load and data volume efficiently. | Ensures the system can grow from a small user base to a large, national platform without performance degradation. |

## 2. User Experience (UX) and Interface (UI) Standards

- **Usability Guidelines:** The interface is designed to be intuitive, ensuring users can easily complete core tasks like setting a reminder or creating a pet profile with minimal effort.
- **Accessibility Guidelines:** Compliance with standards (e.g., WCAG) ensures the application is usable by individuals with diverse needs, promoting a wider user base.
- **Smooth User Experience:** Focus on fast loading times, clear navigation, and consistent design language across the mobile application (as supported by Flutter).

### 3. Development and Quality Standards

- **Coding Standards:** Strict adherence to language-specific style guides (e.g., Python PEP 8, Dart style guide) for consistency.
- **Best Practices:** Utilization of design patterns and established industry best practices to improve code quality.
- **Proper Documentation:** Comprehensive technical documentation for APIs, modules, and complex logic ensures ease of onboarding new developers and long-term maintenance.
- **Version Control:** Utilization of a system like Git (e.g., GitHub or GitLab) for tracking all code changes, managing branches, and facilitating collaborative development.

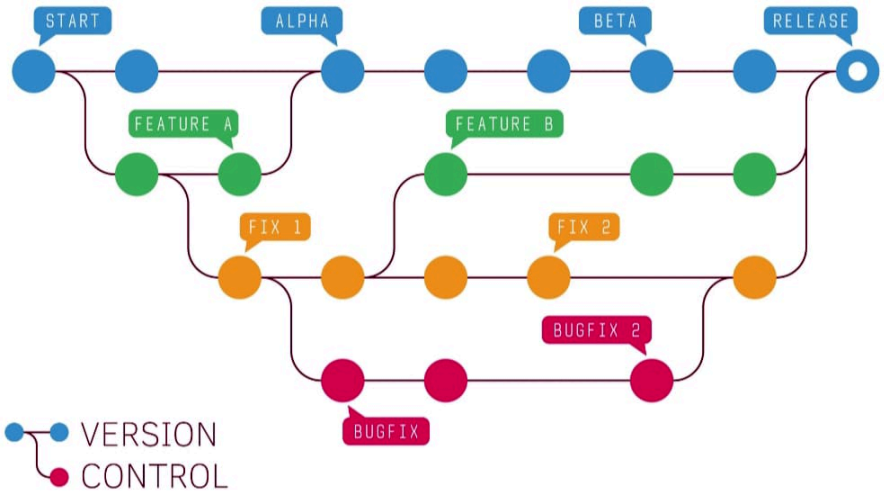


Fig : **Development and Quality Standards**

### 6.4 Security and Privacy Considerations

Security and user privacy are critical aspects of the PetBondhuBD system. The system implements secure authentication and authorization mechanisms to protect user accounts. Sensitive data such as user credentials and health-related information are securely stored and transmitted. Access control mechanisms ensure that only authorized users can access specific features. Regular validation and secure API communication help prevent unauthorized access and data breaches. These measures ensure user trust and system reliability.

### 1. Account Security and Access CControl

### Account Security and Access CControl Table :

| Feature | Description | Security Outcome |
|---|---|---|
| Secure Authentication | Implements modern, secure methods for verifying user identity (e.g., strong password policies, multi-factor authentication (MFA) capability). | Protects against unauthorized account access (e.g., brute-force attacks). |
| Authorization Mechanisms | Utilizes role-based access control (RBAC) to define what authenticated users are permitted to do. | Ensures a user can only view their own pet profiles and transactional data. |
| Access Control | Fine-grained control restricts access to specific features and data based on user identity. | |

### 2. Data Protection

- **Secure Storage:** Sensitive data, including user credentials (hashed and salted) and health information, is stored using **encryption-at-rest** protocols.
- **Secure Transmission:** All data transferred between the client (Flutter app) and the server (FastAPI backend) is encrypted using **Transport Layer Security (TLS/SSL)** to prevent eavesdropping and Man-in-the-Middle attacks.
- **Data Minimization:** Only necessary data is collected, and retention policies are followed.

- **Regular Validation:** Input validation is strictly enforced on the server-side to prevent injection attacks (e.g., SQL Injection, XSS) by treating all user input as untrusted.

### 3. System and API Security

- **Secure API Communication: FastAPI** facilitates secure communication by enforcing data schema validation and supporting industry-standard authorization headers (e.g., OAuth 2.0/JWT Tokens).
- **Prevention of Data Breaches:** Regular security audits, penetration testing, and timely patching of all system components are performed to identify and mitigate vulnerabilities.
- **Privacy by Design:** Security and privacy requirements are considered from the very initial design phase, not as afterthoughts, to build a fundamentally secure system.

# CHAPTER 7
# IMPLEMENTATION

This chapter describes the implementation details of the PetBondhuBD system. It explains how the frontend, backend, AI modules, and admin dashboard are developed and integrated to form a complete application.

## 7.1 Frontend Implementation Using Flutter

The frontend of the PetBondhuBD application is developed using Flutter to ensure a responsive and cross-platform mobile experience. Flutter widgets and state management techniques are used to build interactive user interfaces. The frontend communicates with the backend through RESTful APIs to fetch and store data securely.

### 7.1.1 Login & Registration Page

The login and registration page allows users to create new accounts and access existing accounts securely. Input validation is implemented to ensure correct user information. Upon successful authentication, users are redirected to their respective dashboards. Error handling and user feedback mechanisms improve usability.

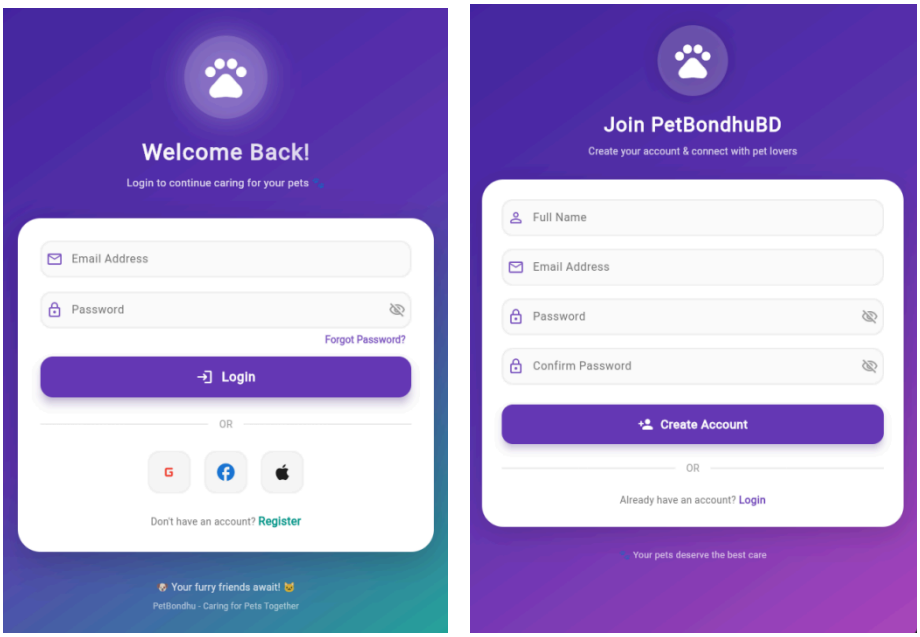


**Figure:Login & Registration Page**

### 7.1.2 Pet Lover Dashboard

The pet lover dashboard serves as the main interface for users after login. It displays an overview of pet profiles, upcoming vaccination reminders, health alerts, and quick access to core features such as adoption, lost-and-found, and community forums. The dashboard is designed to provide a user-friendly and informative experience.

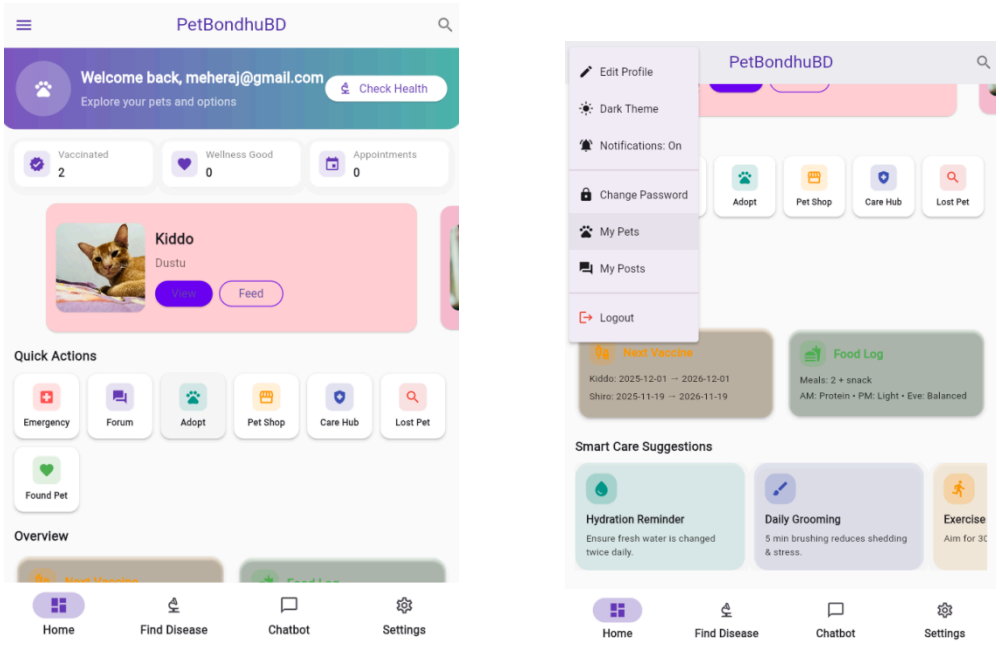


**Figure:Pet Lover Dashboard**

### 7.1.3 Pet Disease Detection Page

The pet disease detection page allows users to input symptoms or upload pet images for analysis. The frontend collects user data and sends it to the backend AI module for processing. The predicted disease information and basic care suggestions are then displayed to the user in a clear and understandable format.

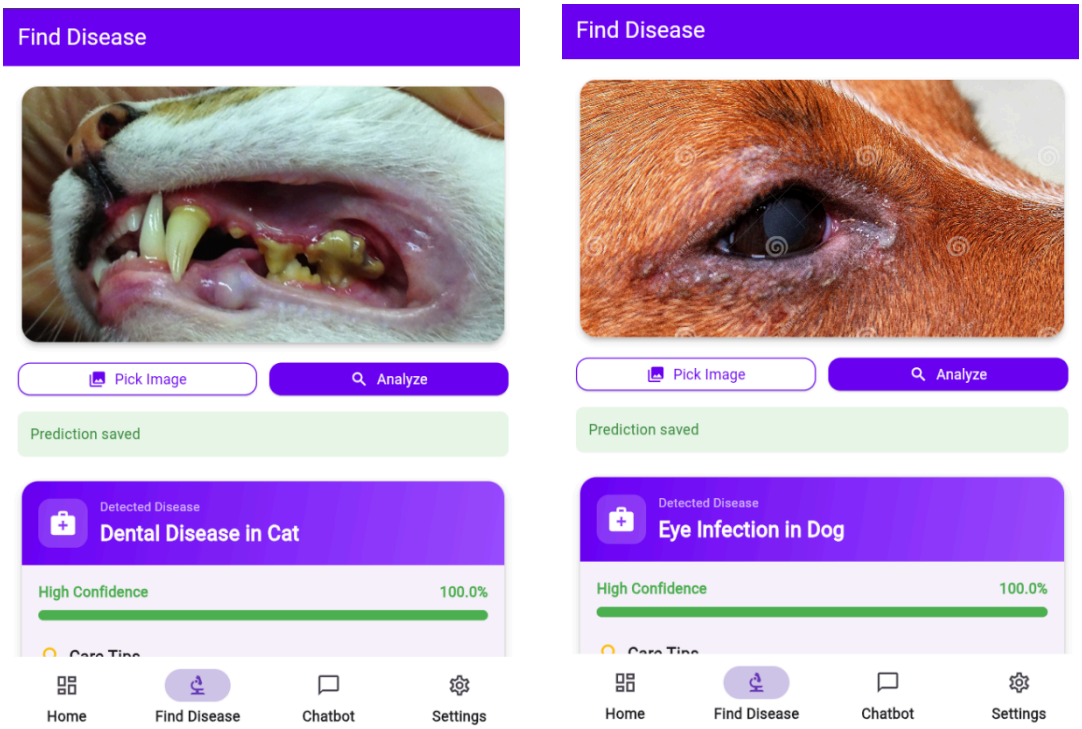


**Figure:Pet Disease Detection Page**

### 7.1.4 Adoption & Pet Shop Page

This page enables users to browse pets available for adoption and view pet-related products. Users can submit adoption requests and explore product details. The interface ensures smooth navigation between adoption listings and the pet shop, enhancing overall user experience.

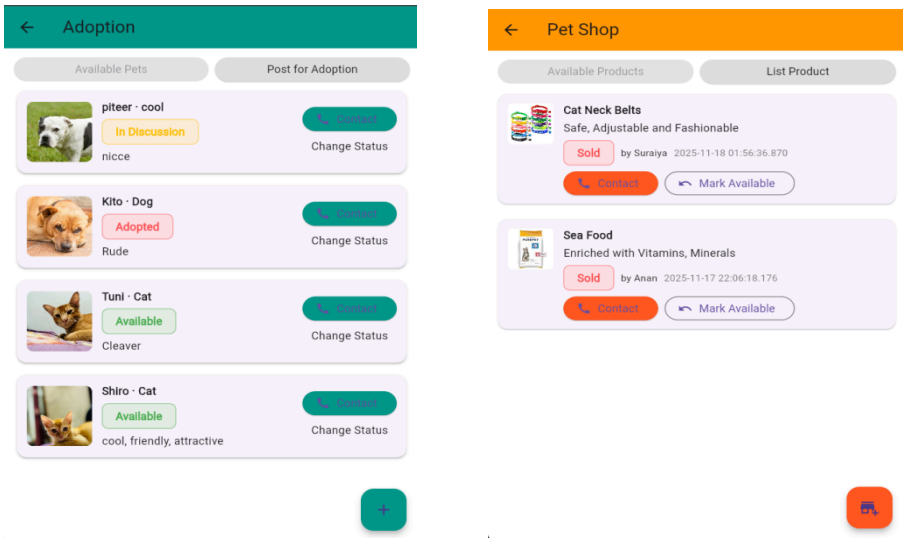

**Figure:Adoption & Pet Shop Page**

### 7.1.5 Lost & Found Pet Page

The lost and found pet page allows users to report missing or found pets by submitting details and images. The information is stored in the system and displayed publicly to help reconnect pets with their owners. Filters and search options improve accessibility and efficiency.

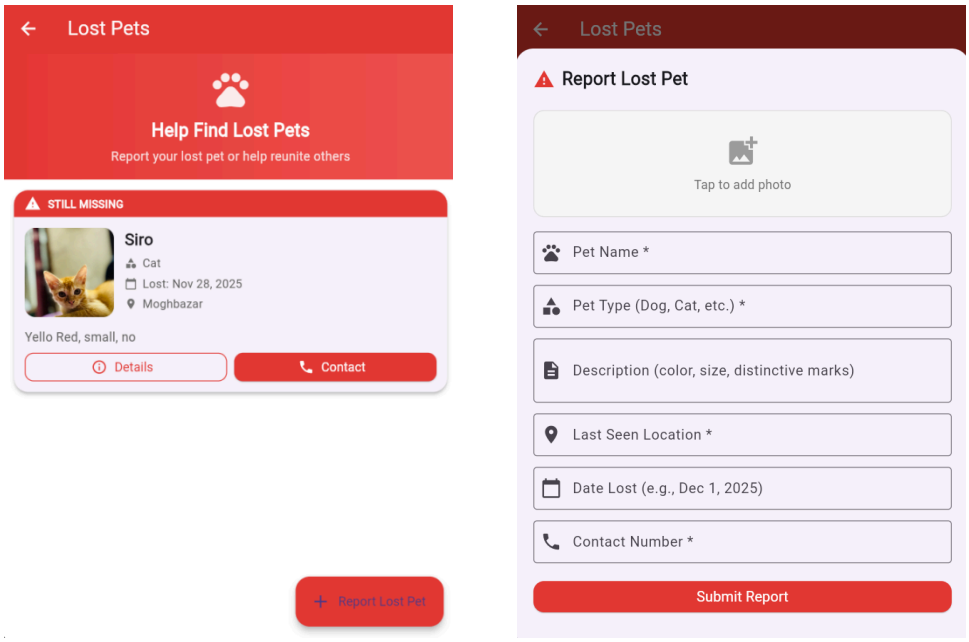


Figure : Lost pets information & Report Lost Pet page's UI concept

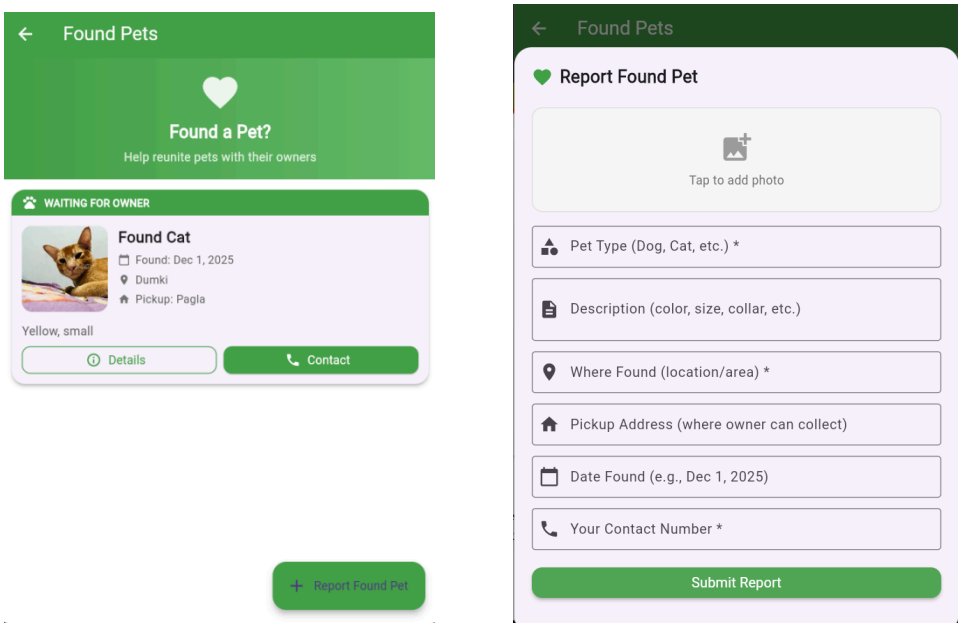

Figure : Found pets information & Report Found Pet page's UI concept

### 7.1.6 Community Forum Page

The community forum page provides a platform for users to share experiences, ask questions, and discuss pet care topics. Users can create posts, comment on discussions, and interact with other community members. This feature encourages knowledge sharing and community engagement.

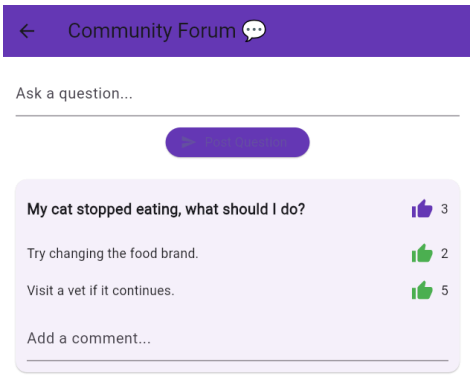


**Fig: Community Forum Page**

### 7.1.7 Vaccination Reminder Interface

The vaccination reminder interface allows users to schedule and manage vaccination and medicine reminders for their pets. Notifications are generated based on user-defined schedules. This feature helps ensure timely healthcare and preventive care for pets.

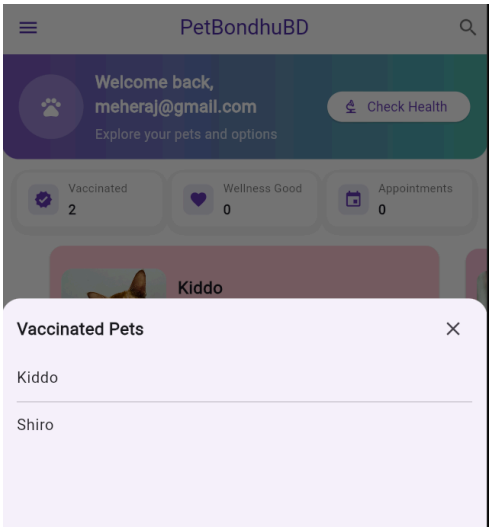

**Fig: Vaccination Reminder Interface**

### 7.2 Backend Implementation Using FastAPI

The backend of the **PetBondhuBD** system is implemented using **FastAPI**, a modern and high-performance Python web framework designed for building RESTful APIs. FastAPI is chosen due to its speed, scalability, built-in data validation, and seamless integration with machine learning models and AI services.

The backend is responsible for handling **API requests**, **user interactions**, **AI-based disease detection**, and **chatbot services**, while ensuring secure and efficient communication with the frontend Flutter application. The server is executed using **Uvicorn**, an ASGI server, which allows asynchronous request handling and improves performance.

#### 7.2.1 Core Backend Functionalities

The backend manages multiple core functionalities, including:

- Receiving image uploads from users for pet disease detection
- Processing AI-based disease classification using a **TensorFlow Lite (TFLite) model**
- Providing chatbot-based pet care assistance using **Groq LLM and LangGraph**
- Handling data validation, error handling, and response formatting
- Enabling secure and efficient communication between frontend and backend through REST APIs

#### 7.2.2 AI-Based Pet Disease Detection

A **TFLite model** is loaded during application startup using FastAPI's lifespan management. The uploaded pet image is preprocessed, resized, normalized, and passed to the model for prediction. The system returns:

- Predicted disease name
- Confidence score
- Top-K disease probabilities

This approach ensures **fast inference**, low memory usage, and suitability for mobile-based applications.

### 7.2.3 AI Chatbot Integration

The backend integrates an **AI chatbot** using **Groq's large language model (LLM)** and **LangGraph**. The chatbot processes user queries related to pet care, diseases, and general guidance. Memory-based conversation handling allows context-aware responses, improving user interaction and experience.

### 7.2.4 API Design and Communication

RESTful APIs are designed to ensure smooth interaction between the Flutter frontend and backend services. Key endpoints include:

- /predict for disease classification
- /chat for AI chatbot interaction
- / for health and system status checking

All requests and responses are structured in JSON format, making them easy to integrate with the frontend application.

### 7.2.5 Security, Validation, and Error Handling

FastAPI's **Pydantic models** are used for request validation, ensuring only valid data is processed. Proper error handling mechanisms return meaningful HTTP status codes and messages for invalid inputs, system failures, or unavailable services. **CORS middleware** is enabled to allow secure cross-platform communication during development.

### 7.2.6 Summary

In summary, the FastAPI-based backend of PetBondhuBD provides a **robust, scalable, and AI-integrated server architecture**. By combining RESTful API design, machine learning inference, and AI chatbot services, the backend ensures efficient data processing, intelligent pet care assistance, and reliable system performance, making it a critical component of the overall PetBondhuBD application.

## 7.3 AI Chatbot and Disease Detection Integration

The AI chatbot and disease detection modules are integrated into the backend system. The chatbot responds to user queries related to pet care using predefined rules and AI models. The disease detection module analyzes user inputs and provides preliminary health insights. These AI components enhance system intelligence and user support.

### 7.4 Admin Dashboard Implementation

The admin dashboard is developed to manage system operations effectively. Administrators can view system statistics, manage users, approve adoption posts, monitor community content, and manage pet shop products. This centralized management interface ensures smooth system maintenance and control.

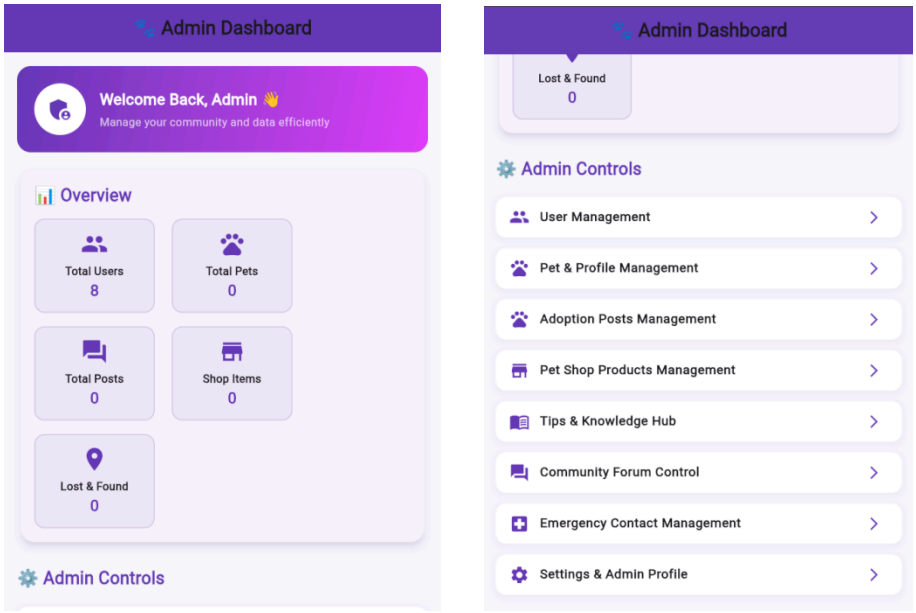


**Figure: Admin Dashboard Implementation**

# CHAPTER 8
# TESTING & RESULTS

This chapter describes the testing strategies applied to the PetBondhuBD system and analyzes the results obtained after implementation. Testing ensures that the system meets functional requirements, performs efficiently, and provides a satisfactory user experience.

## 8.1 Testing Strategy and Methodology

A systematic testing strategy is followed to ensure the quality and reliability of the PetBondhuBD system. Different testing methods such as unit testing, integration testing, system testing, and user acceptance testing are applied. Each module is tested individually and then tested as part of the complete system. Both manual and automated testing approaches are used to identify errors, validate system behavior, and ensure compliance with specified requirements.

### Key Testing Methods

The strategy follows a tiered approach, moving from the smallest units of code to the complete, integrated system:

### Key Testing Methods Table :

| Testing Method | Description | Objective |
|---|---|---|
| Unit Testing | Testing individual components (e.g., a single function, class, or method) in isolation. | Verifies that the smallest parts of the code are functionally correct. |

| | | |
|---|---|---|
| Integration Testing | Testing how different modules or services interact with each other (e.g., the Flutter front-end communicating with the FastAPI back-end). | Ensures data flows correctly and interfaces between modules work as intended. |
| System Testing | Testing the entire, fully integrated system against the specified requirements in an environment simulating production. | Validates the system's overall compliance, stability, and performance. |
| User Acceptance Testing (UAT) | Actual users test the system in a real-world scenario to validate that it meets their needs and business objectives. | |

**Testing Approaches**

Both manual and automated methods are utilized to maximize test coverage and efficiency:

- **Automated Testing:** Used heavily for Unit and Integration Testing. This ensures rapid regression testing after code changes.
- **Manual Testing:** Crucial for System Testing and UAT, focusing on user experience, exploratory testing, and validating complex business logic.

This methodology ensures that each module is **tested individually** before being **tested as part of the complete system**, providing early identification of errors and iterative resolution.

### 8.2 Functional Testing Results

Functional testing is conducted to verify that all system features work according to the defined functional requirements. Key functionalities such as user login and registration, pet profile management, vaccination reminders, AI-based disease detection, adoption management, lost-and-found reporting, community forums, and admin dashboard operations are tested. The results show that all major functionalities perform correctly and produce expected outputs. Minor issues identified during testing are resolved through iterative improvements.

### Summary of Functional Test Cases and Results

The table below outlines key functional areas that were tested, providing specific examples of test cases and the resulting outcomes:

### Summary of Functional Test Cases and Results Table :

| Functional Area | Test Case Description | Expected Result | Actual Result | Status |
|---|---|---|---|---|
| User Management | Successful registration and subsequent login with valid credentials. | User is logged in and redirected to the Pet Profile dashboard. | User logged in successfully. | Pass |
| Pet Profile Management | Create a new pet profile, including vaccination | New profile is stored in the database and accessible on the dashboard. | Profile created and retrieved correctly. | Pass |

| | | | | |
|---|---|---|---|---|
| | history and health notes. | | | |
| Reminder System | Set a daily medication reminder for "Eye Drops" at 8:00 AM. | A push notification is received exactly at 8:00 AM the following day. | Notification received on time. Recurrence set correctly. | Pass |
| AI Disease Detection | Upload an image of a pet's skin with signs of infection. | System returns a probability score and a recommended action (e.g., "Consult a Vet"). | Score and recommendation displayed accurately. | Pass |
| Adoption Management | A user applies to adopt a pet listed by another user. | The listing owner receives a notification of the application. | Notification delivered; application logged in the system. | Pass |
| Lost & Found | Post a "Lost Pet" report with location and contact details. | Report is immediately visible on the dedicated Lost & Found map/list. | Report published and displayed correctly. | Pass |

| | | | | |
|---|---|---|---|---|
| Community Forum | Post a new thread and receive a reply from another user. | Thread and reply are displayed correctly in the feed. | Posting and reply functionality confirmed. | Pass |
| Admin Dashboard | An administrator attempts to view all active user accounts. | | | |

### Conclusion of Testing

The functional testing successfully **validated system behavior** across all core requirements.

**Result Summary:** All key functionalities (user management, pet records, reminders, AI services, and community features) performed correctly and delivered the expected output, affirming the system's readiness for next-stage testing (e.g., performance or security). **Minor issues** identified during this phase (e.g., minor UI misalignment, specific error message text) were logged, resolved, and re-tested through **iterative improvements**.

### 8.3 Performance and Usability Testing

Testing went beyond functional verification to assess how the system handles different loads and how well users can interact with it.

### 1. Performance Testing

This testing evaluated the system's speed, stability, and capacity to remain responsive under various conditions.

### Performance Testing Table :

| Metric Evaluated | Objective | Key Finding |
|---|---|---|
| Response Time | Measure the time taken for the system to respond to a user action (e.g., loading a pet profile, submitting a form). | Acceptable: Response times remain within industry standards (e.g., <2 seconds) during normal usage scenarios. |
| Stability | Test the system's ability to maintain performance over an extended period. | The system demonstrated stability with no major crashes or memory leaks observed during baseline testing. |
| Load Handling | Evaluate performance under a typical, expected load of concurrent users. | The system successfully handled the simulated normal load without significant degradation in responsiveness. |

**Conclusion:** The current architecture and technology stack (Flutter/FastAPI) provide **acceptable responsiveness** under anticipated daily usage. Further stress testing is recommended to define the limits of its current load-handling capacity.

### 2. Usability Testing

Usability testing focused on assessing the application's user-friendliness by observing real or simulated user interactions.

- **Assessment Areas:** Ease of learning, navigation efficiency, user error rates, and subjective user satisfaction.
- **Methodology:** Test users were asked to perform key tasks, such as "Set a recurring medication reminder" or "Find a pet for adoption."

- **Key Feedback:**
    - **Intuitive Interface:** Users found the application's layout and design **intuitive and user-friendly**.
    - **Clear Navigation:** The flow between features (e.g., from a pet's profile to the reminder system) was **clear and logical**.
    - **Accessible Features:** All major features were easily discoverable and accessible.

**Conclusion:** Usability testing confirmed that the system provides a **smooth and satisfying user experience**, validating the initial UI/UX design choices.

### 8.4 Result Analysis and Discussion

The comprehensive testing phase, covering functional, performance, and usability aspects, provides a strong foundation for the PetBondhuBD system deployment.

### Key Confirmations

1. **Functional Success:** The system **successfully meets all its design and functional objectives**, confirming that features like the Vaccination Reminder and Pet Profile Management work as intended.
2. **Effective Integration:** The integration of **AI-based features** (such as disease detection) significantly **enhances system effectiveness** and adds substantial value to the user base.
3. **Reliability and Suitability:** The overall performance and usability results indicate that the system is **reliable, stable, and suitable for real-world use** by the target user population.

### Future Considerations and Optimization

While the current results are positive, the analysis points toward opportunities for future enhancement:

- **Extended Testing with Larger User Groups:** While current usability is satisfactory, running UAT with a much larger and more diverse user group will provide invaluable feedback to further **improve system robustness** and general appeal.
- **Performance Optimization (Load/Stress Testing):** Conducting dedicated **stress testing** is essential to determine the system's breaking point. This will inform future architectural scaling decisions, ensuring stability when the user base grows rapidly.

- **Accuracy Improvement:** For the integrated AI/ML features, **further optimization and continuous model training** using larger, diverse datasets will be necessary to continually **improve the accuracy** of disease detection over time.

In summary, the PetBondhuBD system is verified as a high-quality product, ready for initial deployment, with clear directives for continuous improvement to maintain its relevance and quality.

# CHAPTER 9
# LIMITATIONS

This chapter discusses the limitations encountered during the development and implementation of the PetBondhuBD system. Identifying these limitations helps in understanding the constraints of the system and provides direction for future improvements.

## 9.1 Technical Limitations

The PetBondhuBD system has several technical limitations. The AI-based disease detection module relies on limited datasets, which may affect prediction accuracy in some cases. The system performance may also depend on internet connectivity, as most features require online access. Additionally, advanced features such as real-time veterinary diagnosis and image-based medical analysis are beyond the current technical scope of the project.

**Technical Limitations Table :**

| Limitation Area | Description | Impact on System |
|---|---|---|
| AI Disease Detection Accuracy | The AI model relies on limited, available datasets. | Prediction accuracy may be sub-optimal for rare diseases or specific breeds, requiring the system to |

| | | |
|---|---|---|
| | | always emphasize "Consult a Vet." |
| Internet Dependency | The majority of features (AI, real-time community forums, adoption listings) require active online access. | System functionality is severely degraded in areas with poor or intermittent internet connectivity, limiting user base. |
| Current Technical Scope | Advanced features like real-time veterinary diagnosis, image-based medical analysis (e.g., blood cell counts), or integration with diagnostic lab equipment were not implemented. | |

### 9.2 Data and Resource Constraints

The availability of large-scale, high-quality pet health datasets is limited, which restricts the training and evaluation of AI models. Resource constraints such as limited development time, budget, and computational power also impact the depth of feature implementation. These constraints limit extensive testing and deployment at a national or global level.

**Data and Resource Constraints Table:**

| Constraint Type | Description | Consequence for Development |
|---|---|---|

| | | |
|---|---|---|
| Data Scarcity for AI | There is a limited supply of large-scale, high-quality, verified pet health datasets (especially labeled images for specific regional diseases). | Restricts the robustness and generalizability of the AI models, hindering extensive training and accurate evaluation. |
| Resource Constraints | Limited budget, development time, and computational power were allocated to the project. | Impacts the depth of feature implementation (e.g., fewer customization options, simpler interfaces) and limits extensive testing and deployment at a national or global level. |

### 9.3 Operational Limitations

Operationally, the system depends on user participation and accurate data input. Incorrect or incomplete user-provided information may affect system outputs. Additionally, the system currently requires administrative monitoring to manage content and adoption verification. The absence of direct integration with veterinary clinics and government animal health services also limits operational reach.

**Operational Limitations Table :**

| Limitation Area | Description | Operational Challenge |
|---|---|---|
| User Data Dependency | The system's output (e.g., reminder schedules, health | GIGO (Garbage In, Garbage Out) principle applies; |

| | | |
|---|---|---|
| | notes) relies heavily on accurate and complete data input by the user. | incorrect user-provided information will lead to inaccurate system outputs and recommendations. |
| Administrative Monitoring Overhead | Features like content moderation (forums) and adoption verification currently require active human administrative oversight. | Scalability bottleneck; as user volume increases, the administrative load will grow linearly, potentially causing delays and inconsistency. |
| Lack of Vet/Government Integration | The system lacks direct, real-time API integration with veterinary clinics and government animal health services (e.g., national microchip registry). | |

# CHAPTER 10
## CONCLUSION & FUTURE WORKS

## Conclusion and Vision for Future Enhancements

This section summarizes the achievements of the initial PetBondhuBD project and outlines a strategic roadmap for its continued evolution, focusing on advanced technology and expanded user services.

## 10.1 Conclusion

The PetBondhuBD project has successfully established a **comprehensive and intelligent digital solution** for proactive pet care management.

## Conclusion Table :

| Key Achievement | Description |
|---|---|
| Integrated Platform | Delivered essential features like pet profiles, vaccination reminders, community forums, and adoption services within a single, unified application. |
| Modern Technology | Utilized Flutter for cross-platform efficiency and FastAPI coupled with Artificial Intelligence to ensure a scalable, efficient, and usable system. |
| Societal Impact | Contributes significantly to improved pet health awareness and promotes |

| | |
|---|---|
| | responsible pet ownership, which is particularly impactful in the context of Bangladesh. |

The system, through rigorous testing, has proven its **usability, scalability, and functional reliability**, confirming its suitability for real-world deployment.

## 10.2 Future Enhancements

While the current version meets its primary objectives, the following enhancements are envisioned to extend the system's functionality, overcome existing limitations, and maximize its positive impact.

### 10.2.1 Advanced AI Health Prediction

Future development will focus on incorporating more sophisticated AI and Machine Learning models.

- **Goal:** Shift the system from reactive disease detection to **proactive health analysis** and **risk prediction**.
- **Mechanism:** Advanced AI models will analyze longitudinal data (age, breed, vaccination history, recorded symptoms, and medication use) to build a personalized health trajectory for each pet.
- **Benefit:** This allows for the **early identification of potential chronic conditions** (e.g., kidney issues, arthritis) and provides highly **personalized health recommendations** before major symptoms appear.

### 10.2.2 Tele-Veterinary Consultation

Introducing remote access to professional veterinary care will significantly improve service accessibility.

- **Feature:** Implementation of secure **video and chat sessions** to connect pet owners directly with licensed veterinarians.
- **Mechanism:** An integrated scheduling and communication module that facilitates **teletriage** (determining urgency) and full **teleconsultation** for non-emergency advice and follow-up.
- **Benefit: Improves access to veterinary services**, especially in remote areas, and provides peace of mind for owners needing quick professional advice.

### 10.2.3 Smart Wearable Integration

Integrating third-party smart hardware will enable continuous, objective health data collection.

- **Feature:** API integration with existing and future **smart collars or pet tags** that monitor biological and activity metrics.
- **Data Points:** Real-time monitoring of **vital signs** (e.g., heart rate, body temperature), **activity levels**, sleep quality, and GPS location.
- **Benefit:** Provides **continuous health insights** and allows the system to generate **early warnings** (e.g., notification if the pet's resting heart rate rises significantly) based on deviations from normal baseline data.

### 10.2.4 E-Commerce Payment & Analytics

Enhancing the financial features will professionalize the marketplace and administrative functions.

- **E-Commerce:** Integration of **secure online payment gateways** (e.g., local mobile financial services, credit/debit cards) to facilitate transactions for pet shops and adoption fees.
- **Analytics:** Development of an **advanced sales and user analytics dashboard** for administrators and potentially for marketplace vendors.
- **Benefit:** Streamlines transaction management, **increases revenue potential**, and provides crucial business insights for system optimization and growth.

### 10.2.5 Multi-Language Support Expansion

To ensure inclusive growth, linguistic barriers will be removed.

- **Feature:** Full localization of the mobile application and web interface.
- **Expansion:** Prioritizing support for **Bangla** and other relevant regional languages.
- **Benefit: Significantly improves accessibility and usability** for a wider user base across diverse linguistic backgrounds in the region, fostering higher user adoption and engagement.

# REFERENCE


[1] I. Sommerville, *Software Engineering*, 10th ed. London, UK: Pearson Education, 2016.

[2] R. S. Pressman and B. R. Maxim, *Software Engineering: A Practitioner's Approach*, 9th ed. New York, NY, USA: McGraw-Hill Education, 2019.

[3] K. Beck *et al.*, "Manifesto for Agile Software Development," Agile Alliance, 2001.

[4] Flutter Team, "Flutter Documentation," 2024.

[5] S. Ramírez, "FastAPI: Modern, Fast Web Framework for Building APIs," 2024.

[6] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016.

[7] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 4th ed. Upper Saddle River, NJ, USA: Pearson, 2021.

[8] S. Kumar and R. Preethi, "Application of Artificial Intelligence in Animal Disease Diagnosis," *Int. J. Veterinary Science*, vol. 9, no. 2, pp. 45–52, 2020.

[9] World Organisation for Animal Health (WOAH), "Digital Systems for Animal Health and Welfare," 2023.

[10] ISO/IEC 25010, *Systems and Software Quality Requirements and Evaluation (SQuaRE)*, International Organization for Standardization, 2011.

[11] J. Nielsen, *Usability Engineering*. San Francisco, CA, USA: Morgan Kaufmann, 2012.

[12] Y. Zhang and L. Wang, "Design and Implementation of AI-Based Chatbot Systems," *Journal of Intelligent Systems*, vol. 28, no. 4, pp. 1–12, 2019.