# PROJECT REPORT

**(Project Term August- November 2024)**

**AdoptiPaws**

## Submitted by


Name of the Student: **Asad Ali**

Roll No.: **34**

Regn No.: **12200430**

Section: **K22DW**


Course Code: **INT252**

Course Title**: Web App Development With React JS**


## Submitted To

**Name of faculty: Prof. Kedar Nath Singh**

**School of Computer Science and Engineering**

# DECLARATION

We hereby declare that the project work entitled ("AdoptiPaws") is an authentic record of our own work carried out as requirements of the Project for the award of a B.Tech degree in Computer Science from Lovely Professional University, Phagwara, under the guidance of Prof. Kedar Nath Singh during August to November 2024. All the information furnished in this project report is based on my own intensive work and is genuine.

**Project Group Number: NA**

**Name of Student: Asad Ali**

**Registration Number: 12200430**

**Signature of Student:** Asad Ali

**Date: 07-10-2024**

# CERTIFICATE

This is to certify that the declaration statement made by the student is correct to the best of my knowledge and belief. He have completed this Project under my guidance and supervision. The present work is the result of his original investigation, effort and study. No part of the work has ever been submitted for any other degree at any University. The Project is fit for the submission and partial fulfillment of the conditions for the award of B.Tech degree in CSE from Lovely Professional University, Phagwara.

**Signature and Name of the Staff Designation: Prof. Kedar Nath Singh**

**School of Computer Science and Engineering,**

**Lovely Professional University, Phagwara, Punjab.**

**Date: 07-08-2024**

# ACKNOWLEDGMENT

I would like to express my heartfelt gratitude to everyone who supported and guided me throughout the development of AdoptiPaws. This project would not have been possible without the invaluable assistance and encouragement of many individuals.

First and foremost, I am deeply thankful to Prof. Kedar Nath Singh, my faculty advisor, for their guidance, expertise, and continuous support. Their insightful feedback and advice greatly enhanced the quality of this project.

I am also grateful to the animal welfare organizations and rescue groups who provided insights into the pet adoption process, helping me better understand the challenges and needs in this field. Their commitment to animal welfare was a key inspiration for this project.

Lastly, I would like to thank my family and friends for their encouragement and patience throughout this journey. Their belief in my abilities motivated me to push through challenges and complete this project successfully.

# TABLE OF CONTENTS


1. **Chapter 1: Introduction**
   - 1.1 Overview of the Project: Mission and Objectives..... Page 5
   - 1.2 Technologies Used.....Page 5
2. **Chapter 2: Problem Profile and Scope**
   - 2.1 Problem Statement.....Page 6
   - 2.2 Rationale and Scope of the Study.....Page 7
3. **Chapter 3: Problem Analysis**
   - 3.1 Product Definition.....Page 8
   - 3.2 Feasibility Analysis.....Page 9
   - 3.2.1 Technical Feasibility.....Page 9
   - 3.2.2 Operational Feasibility.....Page 10
   - 3.2.3 Economic Feasibility.....Page 10
   - 3.3 Project Plan.....Page 11
   - 3.3.1 Research and Requirements Gathering…….……...Page 11
   - 3.3.2 Design Phase.....Page 11
   - 3.3.3 Development Phase.....Page 11
   - 3.3.4 Testing and Quality Assurance…….....Page 11
   - 3.3.5 Deployment.....Page 11
   - 3.3.6 Post-Deployment Support and Maintenance……..Page 11
4. **Chapter 4: System Design**
   - 4.1 Flowcharts.....Page 12
   - 4.1.1 Adopter Registration Flowchart…….....Page 12
   - 4.1.2 Pet Search Flowchart.....Page 12
   - 4.1.3 Adoption Application Flowchart.....Page 12
   - 4.2 Database Design.....Page 13
   - 4.3 Pseudo Code.....Page 14
5. **Chapter 5: Implementation**
   - 5.1 Detailed Feature Implementation.....Page 17
   - 5.1.1 Landing Page in React and Redirection Mechanism…….....Page 17
   - 5.1.2 Backend Features in PHP.....Page 17
   - 5.1.2.1 User Authentication.....Page 17
   - 5.1.2.2 Pet Listings.....Page 18
   - 5.1.2.3 Adoption Requests.....Page 18
   - 5.1.3 Database Connection Using MySQL.....Page 18
   - 5.2 Conversion Plan.....Page 19
   - 5.3 Post-Implementation and Maintenance.....Page 20
   - 5.3.1 Performance Monitoring and Optimization……...Page 20
   - 5.3.2 Bug Fixes and Security Updates.....Page 20
   - 5.3.3 User Support and Feedback Collection…….....Page 20
   - 5.3.4 Future Feature Development.....Page 20
6. **Chapter 6**: **Snapshots of Project**…….……Page 21
7. **Chapter 7: Conclusion**. (Deployed & GitHub Link)…….....Page 24

# Chapter 1: INTRODUCTION

In recent years, the need for a robust platform to streamline the process of animal adoption has become increasingly apparent. Stray and abandoned animals are frequently rescued by NGOs and individual rescuers, yet they often face significant challenges in finding these animals permanent homes. AdoptiPaws was conceived as a compassionate and efficient solution to address this problem by connecting rescued animals with potential adopters. Through an easy-to-navigate platform, AdoptiPaws brings together animal welfare organizations, rescuers, and adopters, creating a community dedicated to providing animals with a second chance at life.

## 1.1 Project Overview: Mission and Objectives

The mission of AdoptiPaws is to simplify and enhance the pet adoption process by offering a transparent, supportive platform for animal adoption and rescue. By collaborating with NGOs and rescue groups, AdoptiPaws ensures that each animal listed receives the necessary care and attention until it finds a new home. The platform serves both adopters and rescuers, catering to a range of user needs—whether it's registering as a new adopter, searching for pets available for adoption, or reporting an animal in need.

The primary objectives of AdoptiPaws are:

- **1.1.1 To streamline the adoption process:** Simplify and expedite the adoption journey by providing a user-friendly interface where adopters can search for and apply to adopt animals.
- **1.1.2 To support animal welfare organizations:** Enable NGOs and rescue organizations to reach a wider audience by listing animals for adoption, providing details about the pet's health, temperament, and background.
- **1.1.3 To encourage responsible pet ownership:** Educate potential adopters on the commitment required for pet ownership, ensuring that every animal is placed in a safe, caring environment.
- **1.1.4 To create an accessible network for rescuers and adopters:** Offer a platform that is inclusive, connecting NGOs, individual rescuers, and adopters, fostering a sense of community support and collaboration.

These objectives underscore AdoptiPaws' dedication to not only enhancing the adoption process but also supporting animal welfare efforts, ultimately creating a safer and more compassionate environment for animals in need.

## 1.2 Technologies Used

To develop a platform that is both functional and visually appealing, AdoptiPaws incorporates several key technologies:

- **1.2.1 PHP (Hypertext Preprocessor):** PHP is used for building the backend of AdoptiPaws, providing robust server-side scripting capabilities. This enables efficient processing of user actions on the site, such as registration, login, and

adoption requests. PHP is known for its flexibility and integration with databases, making it ideal for handling the dynamic aspects of AdoptiPaws. With PHP, AdoptiPaws can process data requests, manage user interactions, and securely store information in the database.

**1.2.2 MySQL:** MySQL serves as the database management system for AdoptiPaws. It is used to store essential information about users, animals, NGOs, and adoption applications. MySQL's ability to manage complex queries and relationships between tables allows AdoptiPaws to maintain a structured and efficient database, enabling quick data retrieval and updates. The relational structure of MySQL ensures that data integrity is maintained across multiple tables, supporting the platform's need for consistent, reliable data storage.

**1.2.3 React:** The landing page of AdoptiPaws is developed in React, a popular JavaScript library for building user interfaces. React enables the creation of a dynamic and responsive landing page, providing a visually appealing and interactive experience for users upon their first visit to the platform. With React, the landing page is equipped with an "Explore" button, which seamlessly redirects users to the main PHP-powered site, maintaining a smooth, cohesive user experience across different parts of the platform.

By combining these technologies, AdoptiPaws provides a seamless experience for users, from their initial exploration of the site to the adoption application process. PHP and MySQL handle the backend processes and data management, while React enhances the front-end experience on the landing page, resulting in a platform that is both functional and inviting.

In summary, AdoptiPaws represents an innovative solution to the current limitations in animal adoption platforms. With a commitment to transparency, community support, and ease of use, AdoptiPaws stands out as a valuable resource for animal welfare organizations and adopters alike, fostering a network of care that will improve the lives of countless rescued animals.

# Chapter 2: PROBLEM PROFILE AND SCOPE

## 2.1 Problem Statement: Challenges in Connecting Rescued Animals with Adopters

The problem of stray and abandoned animals is a significant concern worldwide, impacting animal welfare and public health. Many animals, often due to abandonment, neglect, or mistreatment, end up on the streets or in shelters, where resources may be stretched thin. For animal welfare organizations, finding suitable homes for these rescued animals can be a daunting task. These organizations, typically non-profit or volunteer-driven, face limited resources, including funding, staff, and time, which can hinder their efforts in facilitating adoptions.

One of the major challenges in the current adoption process is visibility. Many animal rescue centers operate locally and struggle to reach a broader audience, limiting the pool of potential adopters. This lack of visibility not only reduces adoption rates but also results in overcrowding

within shelters. For animals, this can mean prolonged stays in shelters, which may contribute to stress and deterioration in their physical and mental health.

Another challenge is the lack of a streamlined and user-friendly adoption process. For adopters, finding an animal that fits their preferences (age, breed, size, temperament) can be time-consuming and confusing, especially when different organizations have different platforms and processes. Additionally, the lack of transparency in some systems may lead adopters to feel uncertain about an animal's health, background, or suitability for their home environment.

Finally, the reporting process for animals in need is often inefficient. Citizens who wish to report an animal they found abandoned or in distress may not know where to go or how to notify the appropriate authorities or rescuers, potentially delaying assistance to animals in need.

### 2.2 Rationale and Scope: How AdoptiPaws Aims to Solve These Challenges

AdoptiPaws was developed to address these critical challenges by providing a centralized, accessible platform for pet adoption. Through AdoptiPaws, animal welfare organizations can connect with a broader audience, giving rescued animals a better chance of finding a loving home. The platform aims to bridge the gap between rescuers and adopters by offering features that simplify the adoption process, increase transparency, and foster a supportive community around animal welfare.

The main objectives of AdoptiPaws in addressing these challenges are as follows:

- **2.2.1 Enhanced Visibility for Animal Listings**: By creating a single platform where multiple NGOs and rescue organizations can list their animals, AdoptiPaws increases the visibility of each animal. This helps attract more potential adopters, even beyond the local communities, thus expanding the adoption pool.
- **2.2.2 User-Friendly Adoption Process**: AdoptiPaws is designed to simplify the steps involved in finding, viewing, and applying to adopt a pet. Adopters can easily browse animals by filters such as breed, age, size, and location, making it easier to find a pet that suits their lifestyle and preferences. This streamlined process reduces time and confusion for adopters, increasing the likelihood of successful adoptions.
- **2.2.3 Transparency and Trust**: AdoptiPaws emphasizes transparency by allowing organizations to provide detailed profiles for each animal, including medical history, behavior traits, and background information. This fosters trust between adopters and rescuers, helping to ensure that animals are placed in suitable homes.
- **2.2.4 Community Engagement and Support**: The platform allows individuals to report animals in need, creating a direct link between the community and rescuers. This feature enables faster response times for animals in distress and raises awareness about the importance of responsible pet ownership.

### 2.3 Anticipated Social Impact on Animal Welfare

The social impact of AdoptiPaws is substantial, as it addresses several key animal welfare issues. By increasing the adoption rate, the platform reduces the strain on shelters and helps prevent overcrowding. This leads to better overall care for the animals that remain in shelters, as resources can be allocated more effectively. Additionally, by promoting transparency and

responsible adoption practices, AdoptiPaws encourages a culture of compassion and accountability toward animals.

The platform also educates adopters on the responsibilities of pet ownership, reducing the chances of animals being returned or abandoned in the future. By building a network of adopters, NGOs, and community members, AdoptiPaws contributes to a society where animals are treated with respect, and their well-being is prioritized

# Chapter 3: PROBLEM ANALYSIS

## 3.1 Product Definition: AdoptiPaws

AdoptiPaws is an intuitive, web-based platform created to streamline the pet adoption process by connecting animal welfare organizations and individual rescuers with adopters. Designed with user experience in mind, the platform aims to simplify the adoption journey and promote responsible pet ownership, providing a reliable resource for people seeking to adopt animals in need.

The main components of AdoptiPaws include:

I. **Landing Page**: The landing page of AdoptiPaws, built using React, offers a visually appealing and interactive first touchpoint for users. The page highlights the mission of AdoptiPaws and encourages visitors to explore the platform. Key elements on the landing page include:
   a. **Mission Statement**: A short, impactful description of the purpose of AdoptiPaws, emphasizing the platform's dedication to supporting animal welfare and helping rescued animals find loving homes.
   b. **Explore Button**: A prominent "Explore" button invites users to begin their journey on the platform. This button redirects them to the main site, seamlessly transitioning them to the PHP-based portion of the website where they can access detailed pet listings, registration forms, and adoption information.
   c. **Navigation Options**: Links for various pages such as "About Us," "Contact," and "Get Involved" give users an overview of the platform and available resources, ensuring easy navigation.

II. **User Interface and Experience**: The AdoptiPaws platform provides a straightforward and user-friendly interface where adopters can browse animals and learn about the adoption process. Key features that enhance the user experience include:
   a. **Detailed Animal Profiles**: Each animal listed for adoption has a profile page that provides important details such as age, breed, size, temperament, health status, and any special needs. Photos and descriptions help adopters connect with the animals and make informed choices.
   b. **Advanced Search Filters**: AdoptiPaws offers search filters that allow adopters to refine their search results by factors such as breed, age, location, and adoption status. This functionality helps users quickly find animals that meet their preferences, improving the overall user experience.

c. **NGO Profiles**: Registered NGOs and rescue organizations have profiles that provide information about their mission, location, and the animals they have available for adoption. This fosters trust and transparency, encouraging potential adopters to connect directly with reputable organizations.

III. **Adoption and Reporting Process**:

a. **Adoption Application**: The platform includes an easy-to-navigate adoption form that guides users through the steps needed to apply for adoption. The form captures necessary details such as the adopter's contact information, experience with pets, and preferred animal characteristics.

b. **Report Animal in Need**: This feature allows community members to report animals they encounter in distress or in need of assistance. Upon receiving a report, registered NGOs and rescuers can take appropriate action, ensuring that more animals receive timely help and care.

### 3.2 Feasibility Analysis

The feasibility analysis for AdoptiPaws covers three essential aspects: technical, operational, and economic feasibility, each playing a crucial role in determining the platform's viability and long-term sustainability.

#### 3.2.1 Technical Feasibility

The choice of PHP, MySQL, and React was carefully considered to meet the technical requirements of AdoptiPaws, ensuring the platform is both functional and scalable.

**a.** **PHP (Hypertext Preprocessor)**: PHP is a widely used server-side scripting language known for its flexibility and integration with MySQL databases, making it ideal for backend development in web applications. It allows for dynamic content creation and efficient handling of requests, which is critical for tasks such as user registration, login, and data retrieval on the AdoptiPaws platform.

**Benefits for AdoptiPaws**: PHP's open-source nature and robust community support offer cost savings and ease of implementation. Its compatibility with MySQL allows AdoptiPaws to store and retrieve data efficiently, ensuring a smooth experience for users.

**b.** **MySQL**: As a reliable relational database management system, MySQL is well-suited for handling structured data, which is essential for storing information about users, animals, NGOs, and adoption applications. MySQL provides data integrity, allowing AdoptiPaws to maintain accurate and up-to-date records across its various tables.

**Benefits for AdoptiPaws**: MySQL's stability and support for complex queries enable the platform to quickly search and filter animal profiles, which is a crucial part of the user experience. Its ability to manage relationships between tables helps maintain data consistency, supporting the platform's need for reliability.

**c.** **React**: React is a popular JavaScript library used for building user interfaces, especially those requiring interactivity and responsiveness. It is used to build the landing page of AdoptiPaws, creating an engaging, dynamic experience that captures user interest.

**Benefits for AdoptiPaws**: React enables quick page loading, which improves user engagement. Its component-based architecture allows for reusability, making future updates to the landing page easier to implement.

In summary, the combination of PHP, MySQL, and React provides a technically sound foundation for AdoptiPaws, supporting the platform's goals of functionality, scalability, and a seamless user experience.

### 3.2.2 Operational Feasibility

The success of AdoptiPaws hinges on its usability and the benefits it offers to both adopters and animal welfare organizations. The platform was designed with the following operational objectives:

a. **Ease of Use for Adopters**: The platform's intuitive interface and straightforward navigation make it easy for users to browse animals, submit adoption applications, and interact with rescue organizations. The advanced search functionality and detailed animal profiles ensure that adopters can find pets that match their specific preferences, creating a positive user experience.

b. **Streamlined Process for NGOs**: Registered NGOs benefit from having a centralized platform where they can list animals and manage adoption requests. By eliminating the need to operate separate websites or social media pages, AdoptiPaws simplifies the administrative tasks involved in pet adoption, allowing NGOs to focus on their core mission of animal rescue and care.

c. **Community Engagement**: The platform also serves as a tool for community engagement, allowing individuals to report animals in need. This feature enables NGOs to respond to community reports promptly, strengthening the bond between animal welfare organizations and the communities they serve.

Overall, the platform is highly operationally feasible due to its user-centered design and the convenience it provides for both adopters and NGOs.

### 3.2.3 Economic Feasibility

The economic feasibility of AdoptiPaws is assessed based on development costs, anticipated operational expenses, and the cost savings it provides to NGOs.

a. **Development Costs**: The use of open-source technologies like PHP, MySQL, and React keeps the development costs relatively low. Since these technologies are widely supported and have strong communities, future updates and maintenance can be managed affordably.

b. **Operational Costs**: The main operational costs are associated with web hosting, domain registration, and ongoing maintenance. Hosting the platform on a shared or cloud-based server provides cost-effective scalability, enabling AdoptiPaws to grow as the user base expands.

c. **Cost Savings for NGOs**: By consolidating adoption listings into a single platform, AdoptiPaws reduces the need for NGOs to maintain individual websites or invest in marketing. This saves NGOs both time and money, allowing them to focus their resources on animal rescue and care.

In conclusion, AdoptiPaws is economically feasible, as it minimizes development and operational expenses while providing substantial savings and value for NGOs.

### 3.3 Project Plan

The development of AdoptiPaws follows a structured timeline with key phases to ensure the successful delivery of the platform. Below is an overview of each phase:

1. **Research and Requirements Gathering:**
    - Conduct research on the needs of adopters and NGOs.
    - Define technical requirements and determine the features needed to create an effective adoption platform.
2. **Design Phase**:
    - Design the user interface for the landing page, animal profiles, and user dashboards.
    - Develop flowcharts, wireframes, and the database schema to ensure all elements align with user needs.
3. **Development Phase**:
    - **Frontend Development**: Build the landing page in React, ensuring it is responsive and visually appealing.
    - **Backend Development**: Implement the PHP server, user registration and login, animal listing, and adoption application features.
    - **Database Setup**: Configure MySQL to handle data storage for users, animals, and organizations.
4. **Testing and Quality Assurance**:
    - Conduct unit testing, integration testing, and user acceptance testing to ensure all features work as expected.
    - Gather feedback from a small group of users, including adopters and NGO staff, to refine the platform.
5. **Deployment**:
    - Deploy the platform to a live server, ensuring all functionalities are accessible and the website is optimized for performance.
6. **Post-Deployment Support and Maintenance (Ongoing)**:
    - Monitor platform performance and address any bugs or user issues that arise.

- Plan for future updates, including potential mobile compatibility and new features.

This phased approach ensures the systematic development, testing, and deployment of AdoptiPaws, allowing the team to address any issues and improve the platform iteratively.

# Chapter 4: SYSTEM DESIGN

The AdoptiPaws platform was designed with both functionality and usability in mind, ensuring that adopters and NGOs can easily navigate the system. The system design includes detailed user journey flowcharts, a structured database schema, and core pseudo code descriptions to illustrate the inner workings of AdoptiPaws.

## 4.1 Flowcharts

Flowcharts play an essential role in visualizing the step-by-step processes for key user interactions on AdoptiPaws. Here are the primary user journeys within the system:

### 4.1.1 Adopter Registration Flowchart

- The registration flow is designed to be straightforward, capturing essential details to create a user profile on AdoptiPaws.
- **Flow Steps**:
    1. **Start**: The user clicks the "Register" button on the platform.
    2. **Enter Information**: The user fills in required fields such as name, email, password, and other optional details like phone number.
    3. **Submit Form**: Once completed, the user submits the registration form.
    4. **Server Validation**: The PHP backend validates the information for completeness and checks for any duplicate emails in the database.
    5. **Confirmation**: If validation is successful, a new user profile is created in the database, and a confirmation message is displayed.
    6. **End**: The user is redirected to the homepage, logged in, or given an option to log in.

### 4.1.2 Flow Steps:

**Start**: The user navigates to the search page.

**Set Search Filters**: The user selects desired filters, including breed, age, size, and location.

**Submit Search**: The user submits the search query.

**Database Query**: The PHP backend queries the MySQL database based on the selected filters.

**Display Results**: The filtered results are displayed on the page, showing animal profiles that match the criteria.

**View Animal Profile**: The user can click on an animal's profile for more detailed information.

**End**: The user decides to proceed with an adoption application or returns to adjust filters.

### 4.1.3 Adoption Application Flowchart

This journey captures the steps an adopter goes through to submit an adoption application for a particular animal.

**Flow Steps**:

**Start**: The user clicks "Apply to Adopt" on an animal's profile page.

**Complete Application Form**: The user fills in the application form with details about their experience with pets, household environment, and any special requirements.

**Submit Application**: The form is submitted, sending the data to the backend.

**Application Processing**: The PHP backend validates and saves the application in the database under "adoption requests."

**Confirmation Message**: The user receives a message confirming that their application was submitted successfully.

**End**: The application status is updated, and the user can view pending requests in their profile.

## 4.2 Database Design

The database schema for AdoptiPaws is structured to maintain data integrity and facilitate efficient queries. The key tables include Users, Animals, NGOs, and AdoptionRequests. Each table is designed to store and relate critical information across various system functionalities.

### 4.2.1 Users Table

**Purpose**: Stores information about adopters and NGO representatives.

**Key Fields**:

- user_id: Primary key, unique identifier for each user.
- name: User's full name.
- email: User's email address, also used for login.
- password: Encrypted password for security.
- user_type: Defines if the user is an "adopter" or "NGO".

- created_at: Timestamp for when the user profile was created.

#### 4.2.2 Animals Table

**Purpose**: Stores details of each animal available for adoption.

**Key Fields**:

- animal_id: Primary key, unique identifier for each animal.
- name: Name of the animal.
- age: Age of the animal.
- breed: Breed or species information.
- size: Size of the animal (small, medium, large).
- health_status: Information on any health conditions.
- ngo_id: Foreign key linking to the NGOs table.
- status: Adoption status (available, pending, adopted).

#### 4.2.3 NGOs Table

**Purpose**: Manages registered NGO profiles and information.

**Key Fields**:

- ngo_id: Primary key, unique identifier for each NGO.
- name: Name of the NGO.
- contact_info: Contact details for the organization.
- location: Geographical area where the NGO operates.
- mission_statement: Short description of the NGO's goals.
- created_at: Timestamp for registration.

#### 4.2.4 AdoptionRequests Table

**Purpose**: Tracks adoption requests submitted by users.

**Key Fields**:

- request_id: Primary key, unique identifier for each request.
- user_id: Foreign key linking to the Users table (applicant).
- animal_id: Foreign key linking to the Animals table (animal requested).

- request_status: Tracks the status of the application (submitted, in review, approved, rejected).
- submitted_at: Timestamp for when the request was made.

### 4.3 Pseudo Code

The pseudo code below describes essential functionalities of AdoptiPaws, including how the front-end React components communicate with the PHP backend and perform essential tasks like pet searches.

#### 4.3.1 React Front-End to PHP Backend Communication

The communication between the React landing page and the PHP backend is achieved using API requests. Here is a high-level pseudo code for handling user redirection from the landing page to the main PHP site.

```
<div className="flex justify-center my-10">
  <a
    href="http://adoptipaws.freesite.online/"
    className="bg-gradient-to-r from-green-500 to-green-800 py-3 px-4 mx-3 rounded-md"
  >
    Explore AdoptiPaws
  </a>

</div>
```

#### 4.3.2 Pet Search Functionality

The pet search feature allows users to find animals based on specific criteria. The pseudo code below outlines the backend logic in PHP for processing a search request.

```
<script>
// JavaScript code for filtering animals based on category and location
function filterAnimals() {
  const category = document.getElementById("animalCategory").value;
  const location = document.getElementById("location").value;
  const animalCards = document.querySelectorAll(".animal-card");

  animalCards.forEach(card => {
    const animalCategory = card.getAttribute("data-category");
    const animalLocation = card.getAttribute("data-location");

    if ((category === "all" || animalCategory === category) &&
        (location === "all" || animalLocation === location)) {
      card.style.display = "block";
    } else {
      card.style.display = "none";
    }
  });
}
</script>
```

### 4.3.3 Adoption Application Submission

This pseudo code describes how a user's adoption application is processed by the PHP backend.

```
if(isset($_REQUEST['submitrequest'])){
    // Checking for Empty Fields
    if(empty($_REQUEST['requestinfo']) || empty($_REQUEST['requestdesc']) || empty($_REQUEST['requestername']) || empty($_REQUEST['requesteradd1']) || empty($_REQUEST['requesteradd2']) ||
        // msg displayed if required field missing
        $msg = '<div class="alert alert-warning col-sm-6 ml-5 mt-2" role="alert"> Fill All Fields </div>';
    } else {
        // Assigning User Values to Variables
        $rinfo = $_REQUEST['requestinfo'];
        $rdesc = $_REQUEST['requestdesc'];
        $rname = $_REQUEST['requestername'];
        $radd1 = $_REQUEST['requesteradd1'];
        $radd2 = $_REQUEST['requesteradd2'];
        $rcity = $_REQUEST['requestercity'];
        $rstate = $_REQUEST['requesterstate'];
        $rzip = $_REQUEST['requesterzip'];
        $remail = $_REQUEST['requesteremail'];
        $rmobile = $_REQUEST['requestermobile'];
        $rdate = $_REQUEST['requestdate'];

        // SQL Query to Insert Data
        $sql = "INSERT INTO submitrequest_tb(request_info, request_desc, requester_name, requester_add1, requester_add2, requester_city, requester_state, requester_zip, requester_email,

        if($conn->query($sql) === TRUE){
            // Fetching the auto-generated ID
            $genid = $conn->insert_id;
            $msg = '<div class="alert alert-success col-sm-6 ml-5 mt-2" role="alert"> Request Submitted Successfully. Your Request ID is: ' . $genid . '</div>';

            // Storing the ID in session for potential future use
            $_SESSION['myid'] = $genid;

            // Redirecting to the success page
            echo "<script> location.href='submitrequestsuccess.php'; </script>";
            // Fallback PHP header redirection in case JavaScript fails
            header("Location: submitrequestsuccess.php");
            exit();
        } else {
            // Displaying any SQL error encountered
            $msg = '<div class="alert alert-danger col-sm-6 ml-5 mt-2" role="alert"> Unable to Submit Your Request. Error: ' . $conn->error . '</div>';
        }
    }
}
```

These flowcharts, database schema, and pseudo code descriptions together form the core architecture and functionality of AdoptiPaws. Each component is designed to enhance the efficiency and reliability of the system, providing a seamless experience for adopters and NGOs while ensuring the accurate handling and storage of data across the platform.

With this detailed system design, AdoptiPaws achieves its goal of creating a user-friendly and technically robust adoption platform, setting a new standard in the digital pet adoption experience.

# Chapter 5: IMPLEMENTATION

The Implementation phase of AdoptiPaws involved building out the core functionality of the platform, integrating both front-end and back-end technologies to create a seamless experience for users. Each feature was developed with usability, scalability, and efficiency in mind, ensuring that AdoptiPaws meets the requirements of adopters, NGOs, and rescuers. This section covers the key components: the landing page and redirection, back-end functionalities for user management and adoption processing, the database integration, conversion plan, and post-implementation maintenance.

## 5.1 Detailed Feature Implementation

### 5.1.1 Landing Page in React and Redirection Mechanism

The landing page, built using the React JavaScript library, serves as the first point of interaction for users visiting AdoptiPaws. Designed to be visually appealing and responsive, the landing page introduces users to the platform's mission and core functionality while encouraging them to explore further.

1. **React Components**:
    - The landing page is composed of multiple reusable components, such as Header, MissionStatement, ExploreButton, and Footer. React's component-based architecture allows each section to be modified or updated independently, enhancing maintainability.
    - **ExploreButton**: This prominent button serves as the primary call-to-action, inviting users to navigate to the main AdoptiPaws site. It uses an onClick event listener to trigger a redirection to the PHP-based site.
2. **Redirection Mechanism**:
    - The ExploreButton component leverages React's navigate function to redirect users to the main site. This action is configured to direct users to a specific URL (e.g., https://adoptipaws.com/home), where they can begin exploring available pets and adoption resources.
    - The use of React for the landing page improves the initial loading speed and responsiveness, making the first user interaction smooth and engaging.
3. **Responsive Design**:
    - The landing page is fully responsive, adapting to various screen sizes using CSS media queries and flexible layouts. This ensures that users accessing AdoptiPaws from mobile devices or tablets have an optimal viewing experience, which is essential for attracting and retaining users across multiple platforms.

### 5.1.2 Backend Features in PHP

The PHP backend is the core of AdoptiPaws, handling essential functionalities such as user authentication, pet listings, and adoption requests. Each of these features is built with data security, performance, and user-friendliness in mind.

1. **User Authentication**:
    - **Registration and Login**: New users, including both adopters and NGO representatives, can create accounts by submitting their details through a registration form. PHP validates the form input, hashes the passwords for security using bcrypt, and stores user information in the Users table. For login, PHP verifies the provided credentials and creates a session, enabling secure access to user-specific features.
    - **Session Management**: PHP session variables store essential data, such as user ID and access level, to maintain user sessions securely across different pages. Additionally, session expiration and logout functionality are implemented to enhance security.
2. **Pet Listings**:
    - **CRUD Operations**: The backend supports full CRUD (Create, Read, Update, Delete) operations for managing pet profiles. NGOs can add, update, or remove pet listings through their dashboard. PHP scripts handle these operations, ensuring data consistency and validating user permissions to prevent unauthorized actions.
    - **Profile Details**: Each pet profile includes fields for name, age, breed, size, health status, and an adoption status (e.g., “available,” “pending,” “adopted”). When a user views a pet’s profile, PHP queries the database for these details and dynamically generates the HTML to display the information.
3. **Adoption Requests**:
    - **Application Submission**: When an adopter expresses interest in an animal, they can submit an adoption application through the platform. The application includes information such as the adopter’s experience with pets and their household environment. PHP validates and stores this information in the AdoptionRequests table.
    - **Application Management**: NGO representatives can review adoption requests through their dashboard, with options to approve or reject applications. PHP updates the request status in the database, and the system notifies the adopter via email of the outcome.

### 5.1.3 Database Connection Using MySQL

The MySQL database serves as the backbone of AdoptiPaws, storing critical data across various tables. The database connection is established through PHP’s mysqli or PDO functions, ensuring secure and efficient data handling.

1. **Database Schema**:

- The database schema includes tables such as Users, Animals, NGOs, and AdoptionRequests. Each table has defined relationships to ensure data integrity and streamline querying.
- For instance, the Animals table has a foreign key linking to the NGOs table, identifying the organization responsible for each animal. Similarly, AdoptionRequests links to both the Users and Animals tables, capturing information about the adopter and the pet they wish to adopt.

2. **Data Retrieval and Storage**:
   - **Dynamic Queries**: The PHP backend performs dynamic queries based on user actions, such as retrieving a list of available animals or filtering animals by certain criteria (age, breed, etc.). This functionality is essential for delivering customized search results to adopters.
   - **Prepared Statements**: For security, all database interactions utilize prepared statements to prevent SQL injection attacks. This is particularly important for handling user input in search filters, registration, and adoption applications.
3. **Error Handling**:
   - The PHP code includes error handling to manage issues such as database connection failures or query errors. Error messages are logged for debugging, while user-friendly messages are displayed to maintain a smooth user experience.

### 5.2 Conversion Plan

If any data migration or conversion is needed in the future (e.g., importing data from an external NGO database or integrating with a new platform), a structured plan ensures that data integrity is preserved.

#### 5.2.1 Data Mapping:

Before starting data migration, map the source data to the AdoptiPaws schema to ensure compatibility. For example, map the fields in an external database to fields in the Animals table, ensuring that all necessary data (such as pet age, breed, and health status) is correctly aligned.

#### 5.2.2 Data Transformation:

Transform data to match AdoptiPaws' format as needed. For instance, if age is stored as "months" in an external system, convert it to "years" to match AdoptiPaws' format. Use PHP scripts to perform these transformations during the import process, which allows for automated data processing without manual input.

#### 5.2.3 Data Validation and Testing:

After migration, validate that all data has been imported correctly by running test queries and manually reviewing a sample of records. This ensures that key details, such as animal profiles and adoption history, are accurately represented in AdoptiPaws.

### 5.2.4 Backup and Rollback Procedures:

Perform regular backups of the database before and after any major migrations to prevent data loss. If any issues are detected, a rollback procedure allows for restoring the previous database state.

## 5.3 Post-Implementation and Maintenance

To ensure AdoptiPaws remains reliable and responsive to user needs, a structured maintenance plan is in place. This includes monitoring performance, addressing issues, and planning for future feature upgrades.

### 5.3.1 Performance Monitoring and Optimization:

Monitor server and database performance using tools such as Google Analytics, PHP error logs, and MySQL slow query logs. Regular monitoring helps identify and resolve performance bottlenecks, such as slow-loading pages or high server loads.

Optimize database queries, minimize JavaScript loading times, and compress images to improve overall page speed and user experience.

### 5.3.2 Bug Fixes and Security Updates:

Regularly review user feedback to identify and address any bugs or functionality issues. Bug reports are prioritized, with critical issues addressed immediately to maintain user trust.

Apply security patches and updates for PHP, MySQL, and React to protect against vulnerabilities. This includes regular updates to password hashing algorithms and session management practices to maintain data security.

### 5.3.3 User Support and Feedback Collection:

Provide a dedicated support channel for users to submit feedback or report issues. Implement a feedback form on the website to allow users to suggest new features or improvements.

Regularly review feedback to identify common issues or requested features, using this input to guide future updates and enhancements.

### 5.3.4 Future Feature Development:

- **Mobile Compatibility**: Develop a mobile-friendly version of AdoptiPaws, potentially as a progressive web app (PWA), to reach a broader audience. This would involve adapting the layout, buttons, and forms to fit smaller screens while maintaining a consistent user experience.
- **Enhanced Search Filters**: Plan to introduce additional search filters, such as animal temperament or activity level, to give adopters more precise options for finding compatible pets.
- **Improved Reporting Tools**: Expand the community engagement feature by allowing users to submit more detailed reports on animals in need, including location tagging and image uploads. This would aid NGOs in assessing cases and planning rescue efforts.

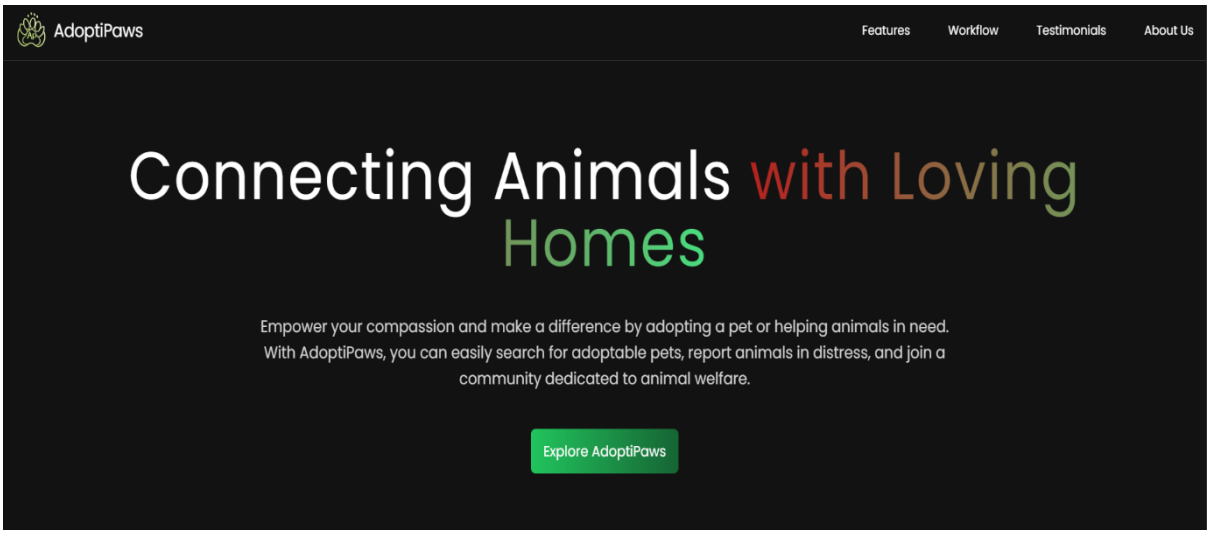
- **Analytics and Data Insights for NGOs**: Provide data insights for registered NGOs, allowing them to track adoption trends, animal turnover rates, and user engagement. This would help NGOs make data-driven decisions and improve their rescue and adoption processes.

### 5.3.5 Scalability and Future Growth:

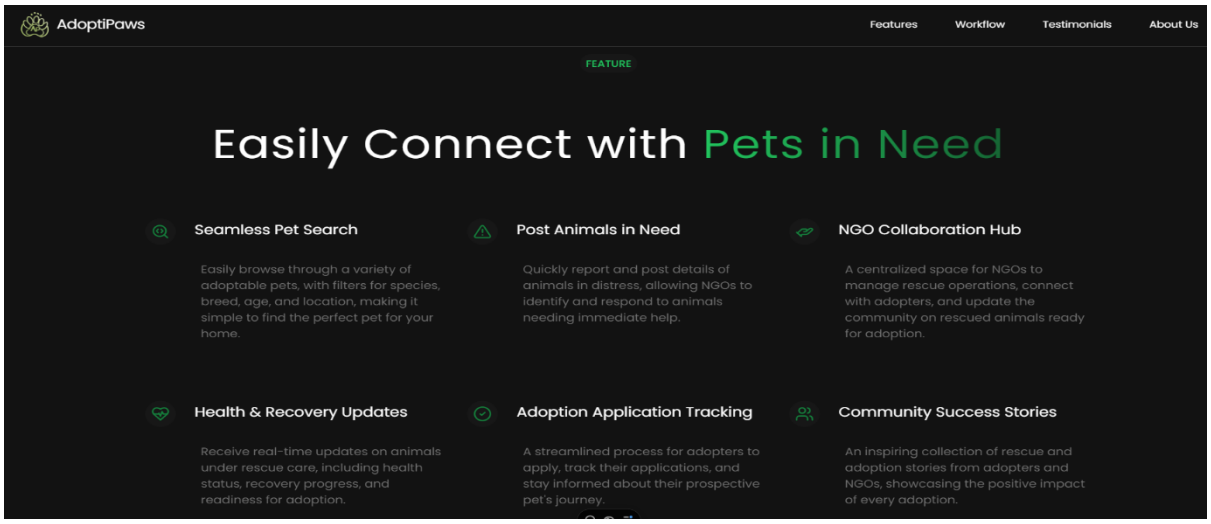
- As user activity grows, upgrade the server infrastructure to handle higher loads. Consider moving to a cloud-based hosting solution (e.g., AWS or Google Cloud) for better scalability and reliability.
- Evaluate the potential integration with external platforms (e.g., pet microchip databases) to enhance the user experience and offer more comprehensive services.

# Chapter 6: SNAPSHOT OF PROJECT

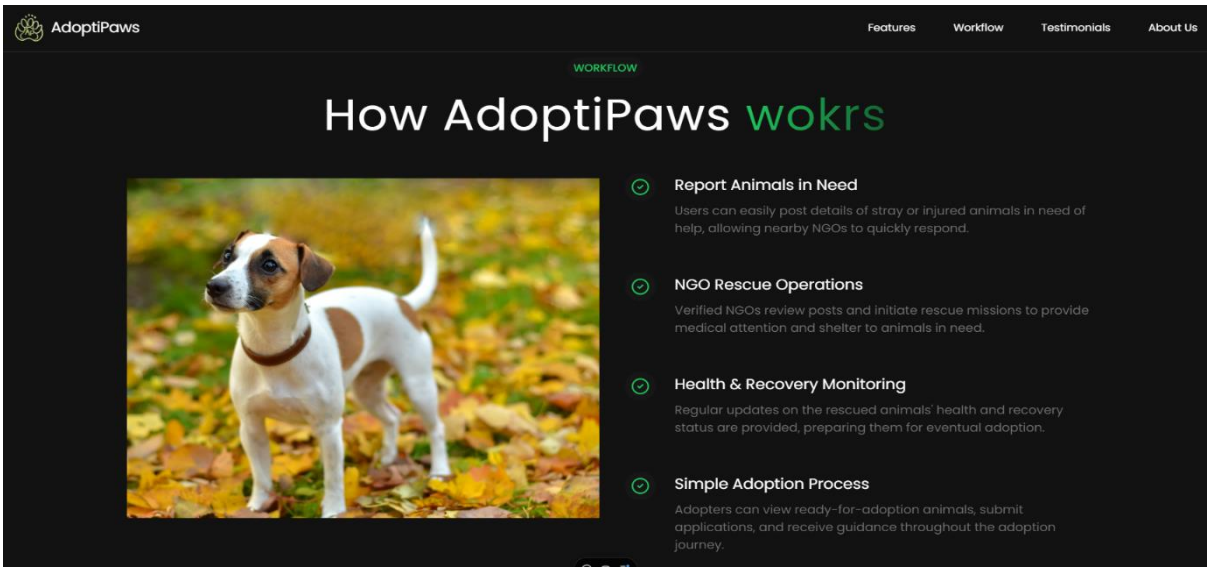
## 6.1 Landing Page



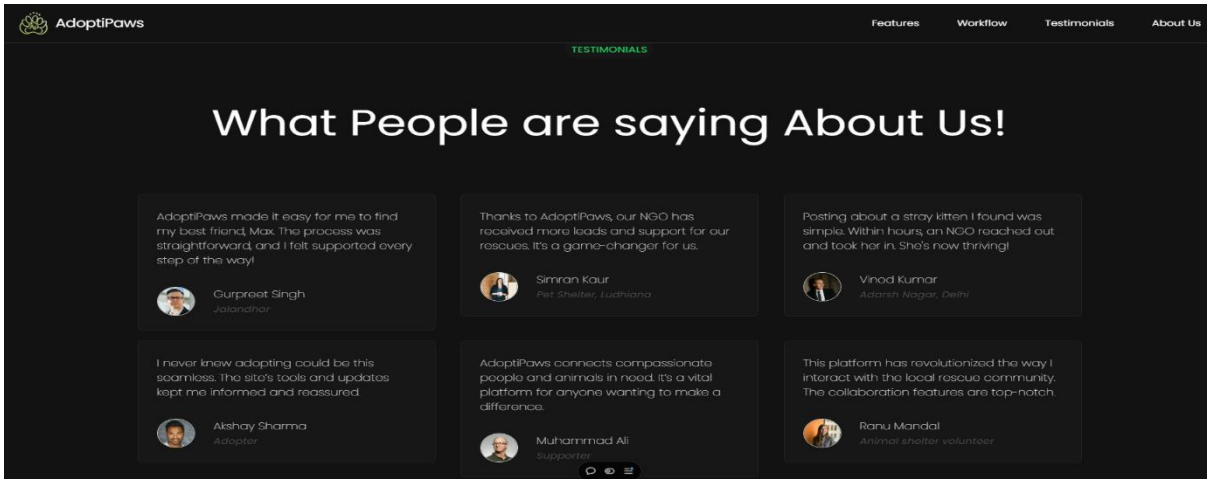

**Feature Section:**



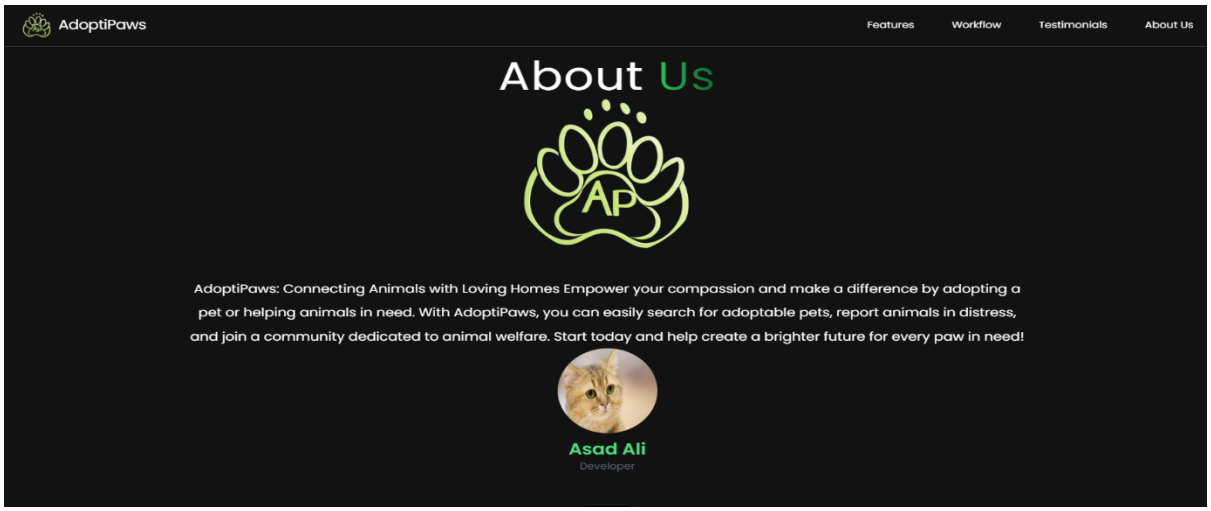
**Workflow Section:**



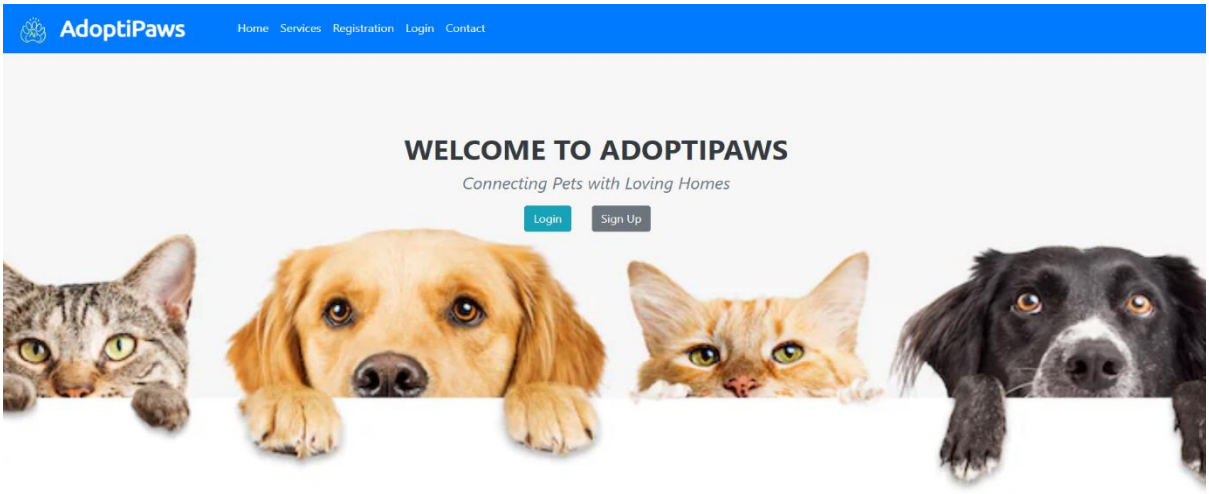

**Testimonial Section:**



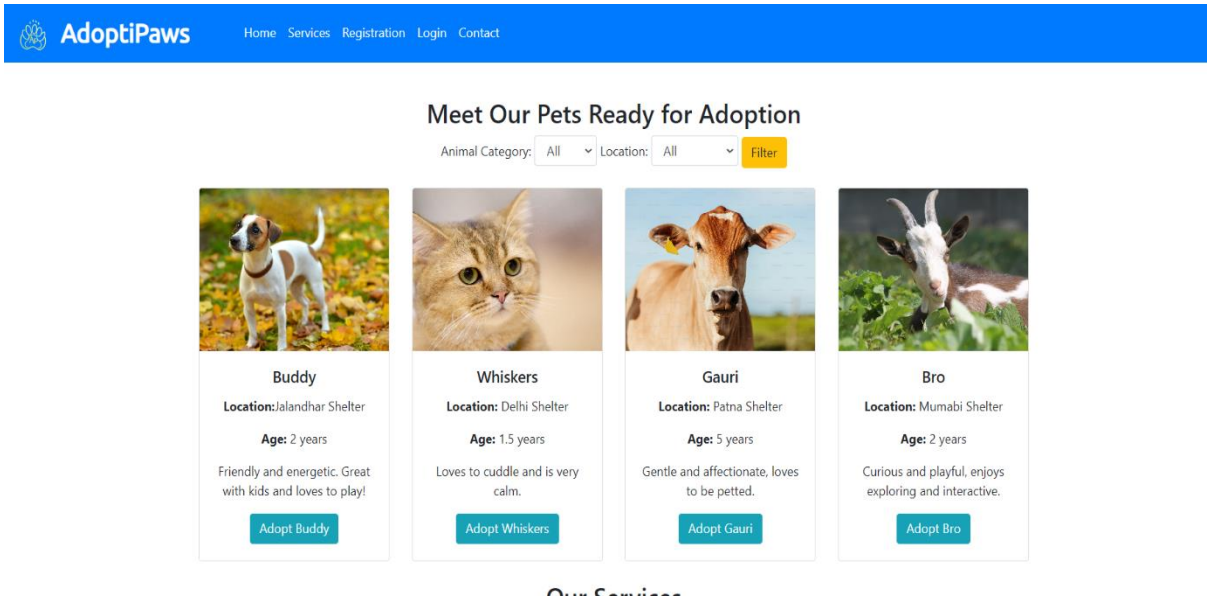

**About Us Section:**



## 6.2 Backed- PHP Page




## Pet Adoption Page:



## Chapter 7: CONCLUSION

AdoptiPaws is a platform dedicated to facilitating pet adoption and supporting animal welfare by connecting rescued animals with potential adopters. The platform was built with a user-centric approach, using technologies like PHP, MySQL, and React to create a seamless, efficient experience for both adopters and NGOs.

Through features such as detailed animal profiles, a streamlined adoption application process, and advanced search filters, AdoptiPaws aims to simplify the journey for adopters while also empowering NGOs to reach a wider audience. Additionally, the platform's design considers future growth, with scalability measures and plans for enhancement, such as pet-matching algorithms and expanded community engagement features.

By providing a centralized, accessible space for adoption and rescue, AdoptiPaws aspires to make a lasting positive impact on animal welfare. As it continues to grow and evolve, AdoptiPaws is committed to supporting responsible pet ownership and helping rescued animals find safe, loving homes. This project represents not only a technical achievement but also a step toward a compassionate future for both people and animals.

- **Deployed Link:** https://adoptipaws-landing-page.vercel.app/
- **React Github Link:** https://github.com/asadali2004/adoptipaws_landing_page.git
- **PHP Github Link:** https://github.com/asadali2004/AdoptiPaws_PHP.git