

FindYourPet Design and Development of a MERN-Based Web Platform for Streamlined Pet Adoption

Pranay Jain¹, Priyal Waghmare², Priyanshi Goyal³, Rachit Shivhare⁴, Rahul Sharma⁵, Kavita Namdev⁶
^{1,2,3,4,5,6}*Department of Computer Science and Engineering Acropolis Institute of Technology & Research
Indore, India*

Abstract—Thousands of animals go to shelters annually with slim prospects of finding new homes due to inadequate controls by current platforms. FindYourPet is an online platform developed using the MERN stack (MongoDB, Express.js, React.js, and Node.js) which provides an admin-controlled approval mechanism for pets' listing and adoption requests. The web application features three types of user accounts (adopters, pet listers, and administrators), OTP email verification, and JWT token authentication to provide assurance. The REST API acts as a bridge between the frontend and the database which comprises pet profiles, user information, and the adoption history. To help potential adopters make informed choices about the pets they wish to adopt, FindYourPet also comes equipped with an AI chatbot which recommends 2-3 pets for consideration based on the context of conversation which consists of living spaces, weather, user schedules, and physical activities. Lastly, a community forum helps users to interact with each other by posting questions and answers to issues regarding pet care.

Index Terms—Pet Adoption, MERN Stack, Web Application, Animal Welfare, REST API, JWT Authentication, Admin Workflow, MongoDB, AI Chatbot, Pet Recommendation System, Community Forum, Natural Language Processing.

I. INTRODUCTION

India has a significant and largely undocumented stray and shelter animal population. Organizations like the Animal Welfare Board of India estimate tens of millions of stray dogs alone, yet the

infrastructure to connect these animals with willing adopters remains fragmented. Most adoptions still happen through word of mouth, local NGOs, or social media posts, none of which offer any standardized process or accountability.

A handful of online platforms do exist, but they largely function as listing boards. There is little to verify who is posting a pet, no structured way for an interested adopter to apply, and no admin oversight to ensure the right match is being made. The result is that many listings go stale, many inquiries go unanswered, and the adoption process remains as informal as it was before the internet.

FindYourPet was built to address this gap in a straight forward way. The platform uses the MERN stack to provide a structured, role-based environment where pet listings are admin-approved before going live, adoption applications are reviewed rather than auto-processed, and users are verified through OTP-based email authentication. The goal was not to build something complex, but something deliberate: a system where each step of the adoption process has a clear owner and a defined outcome.

This paper describes the design, development, and testing of FindYourPet, covering the system architecture, key features, technology choices, and a comparison with existing platforms. The objective is to demonstrate that a well-structured web application, even with a lean technology stack, can bring meaningful improvement to how pet adoption is handled in practice.

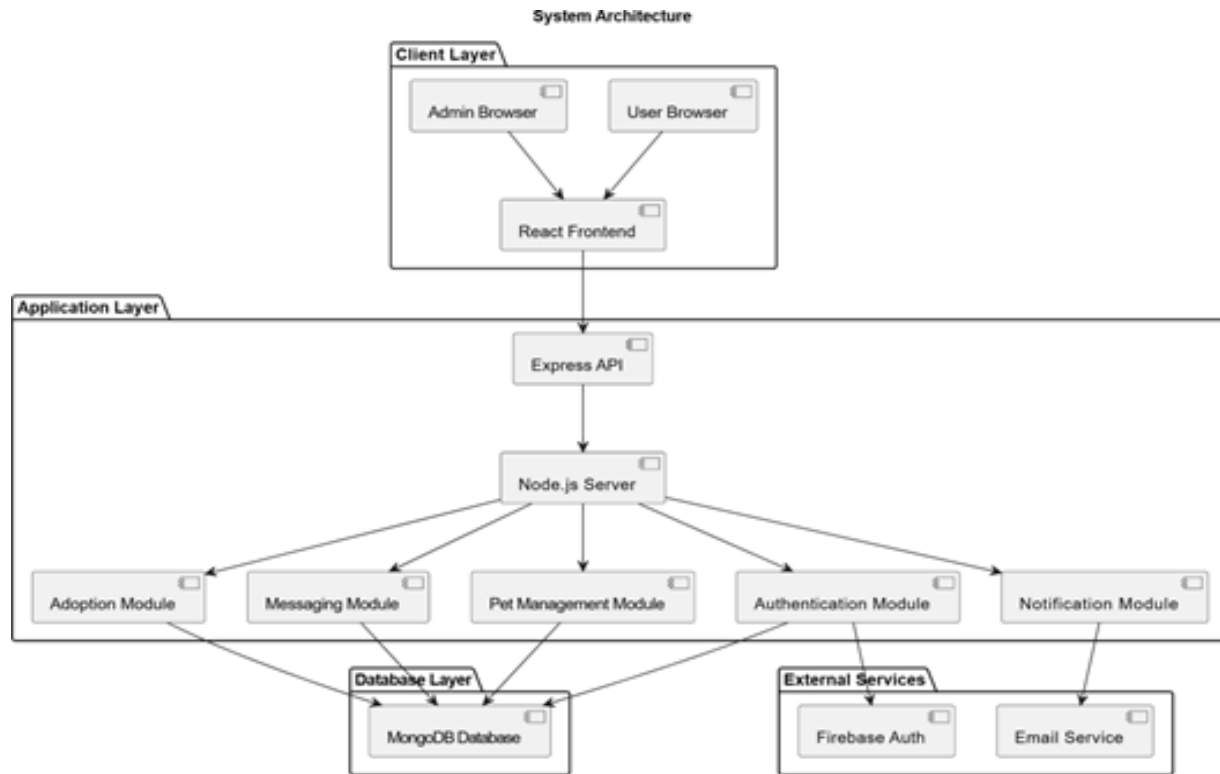


Fig. 1. System architecture

II. LITERATURE REVIEW

Several platforms and studies have explored digital approaches to pet adoption, each contributing useful insights while also revealing consistent gaps:

- Petfinder (petfinder.com): One of the largest pet adoption aggregators, connecting users with shelter animals across an extensive database. Offers detailed pet profiles and adoption resources. However, the open aggregation model results in inconsistent listing quality, and no structured adoption application process exists at the platform level [1].
- Adopt-a-Pet (adoptapet.com): Prioritizes a simplified browsing experience with filters for breed, age, and location, along with real-time notifications for new listings. While user-friendly, the platform lacks detailed pet profiles and does not provide a formal adoption pipeline, leaving the process entirely informal [2].
- Rescue Me (rescueme.org): Focused on rescue networks, allowing organizations to create custom pages for their animals with detailed behavioral and temperament profiles. Despite its strong rescue network, the interface is outdated and user

interaction features are limited, reducing engagement [3].

- Best Friends Animal Society (bestfriends.org): Operates a nationwide database with a strong no-kill advocacy mission and educational resources. However, search customization is limited and user interaction features remain underdeveloped [4].

Prior work highlights that most platforms function as passive listing boards without admin-level quality control or structured adopter evaluation. None of the reviewed platforms implement an approval-based workflow at both the listing and application stage. Table I summarizes the comparative analysis.

TABLE I COMPARATIVE REVIEW OF EXISTING PET ADOPTION PLATFORMS

Platform	Key Features	Advantages	Limitations
Petfinder	Extensive database, pet profiles, adoption resources	Large listings, user-friendly	Inconsistent quality, no adoption pipeline
Adopt-a-Pet	Streamlined browsing,	Simple interface,	Outdated info, lacks

	listing notifications	well organized	structured application
Rescue Me	Rescue profiles, custom pages, global reach	Detailed pet info, strong network	Outdated interface, limited account features
Best Friends	Nationwide database, no-Kill focus, educational resources	Extensive shelter network	Limited search customization, low user interaction
FindYourPet	Admin-approved listings, structured adoption workflow, OTP auth, JWT security, role-based access, AI chat-bot for pet recommendation	End-to-end process control, verified users, community page	Web-only, single language

FindYourPet directly addresses the gaps identified above by introducing admin oversight at both the pet listing and adoption application stage, verified user authentication, and a defined adoption pipeline, none of which are currently implemented together in any of the reviewed platforms.

III. TECHNOLOGY USED

The development of FindYourPet utilized a set of modern, open-source technologies chosen for their compatibility, community support, and suitability for building scalable web applications.

A. Core Stack: MERN

The MERN stack forms the backbone of the platform, chosen for its end-to-end JavaScript consistency, which simplifies development across both frontend and backend layers.

B. Frontend: React.js

React.js was used to build the user interface. Its component-based architecture allows individual

sections of the page, such as pet cards, filters, and forms, to be developed and updated independently. This results in a responsive, dynamic interface without full page reloads, improving the overall user experience.

C. Backend: Node.js and Express.js

Node.js serves as the runtime environment for the server side, enabling asynchronous handling of multiple concurrent requests efficiently. Express.js was used on top of Node.js to define API routes, handle HTTP requests, and manage middleware. Together they form a lightweight but capable backend layer that processes all platform logic including authentication, listing management, and adoption workflows.

D. Database: MongoDB

MongoDB was selected as the database due to its flexible, schema-less document model. Pet profiles, user records, and adoption histories vary in structure depending on the type of animal or user role, making a NoSQL approach more practical than a rigid relational schema. Mongoose was used as an ODM (Object Data Modelling) library to define schemas and interact with MongoDB in a structured way.

E. Authentication: JWT and OTP

User sessions are managed using JSON Web Tokens (JWT), which allow secure, stateless authentication across API requests. For initial identity verification during registration, an OTP is sent to the user's email address, adding a layer of trust to the sign-up process.

F. Styling: Tailwind CSS

Tailwind CSS was used for frontend styling. Its utility-first class system allowed for rapid development of a consistent and responsive UI without writing custom CSS from scratch.

G. Development Tools

Visual Studio Code was used as the primary code editor. Postman was used for API testing during development. GitHub served as the version control and collaboration platform across the team.

H. AI Recommendation Chatbot: Python, Flask, and Google Gemini API

There is a Python-Flask microservice application

dedicated to the integration of the Google Gemini API, which helps generate a chatbot for collecting user information related to lifestyles to provide two or three customized pet suggestions along with their pictures and reasons behind the recommendations via REST APIs.

IV. METHODOLOGY

A. System Design Overview

FindYourPet follows a three-tier architecture consisting of a React.js-based frontend, a Node.js and Express.js backend, and a MongoDB database. All communication between the frontend and backend occurs through a RESTful API, with JWT tokens attached to request headers to authenticate protected routes. The overall system is designed around three user roles: adopters, pet listers (individuals or shelters), and administrators, each with a distinct set of permissions and interface views.

B. Flow Diagram

The user journey through the platform begins at an on-boarding screen where new users register with email OTP verification, and returning users log in directly. Upon authentication, the system identifies the user role and routes them to the appropriate dashboard. Pet listers can submit a pet for adoption by filling out a listing form with details such as pet type, breed, age, location, and a photograph. This submission is held in a pending state until an administrator reviews and approves it. Once approved, the listing becomes visible to all adopters on the platform.

Adopters can browse available pets using search filters, view detailed profiles, and submit an adoption application for a pet they are interested in. The application captures information about the adopter's living situation, prior pet experience, and contact details. The administrator reviews all incoming applications for a given pet and selects the most suitable candidate. Upon completion, the adoption is logged in the system and the pet listing is marked as adopted.

C. Database Design

The MongoDB database, referred to as Pet Adoption DB, contains three primary collections. The Pets collection stores each listing's details including name,

species, breed, age, location, health status, photograph filename, and current adoption status. The Users collection holds records for all registered users with fields for name, email, hashed password, user role, and adoption history. The Adoption Requests collection captures each application submitted by an adopter, storing the

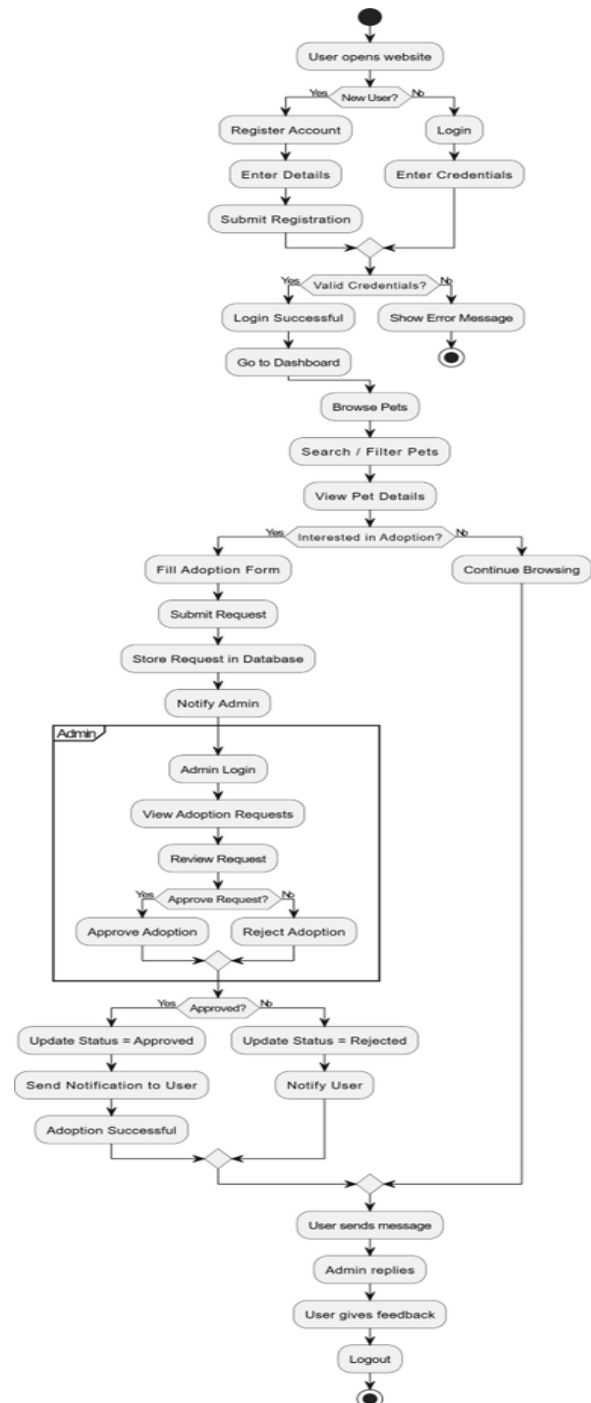


Fig. 2. Complete Workflow Diagram

applicant's contact information, living situation, prior experience, family composition, and a reference to the pet being applied for. This structure keeps data modular and allows the admin panel to query across collections efficiently.

D. API Structure

The backend exposes a set of RESTful endpoints organized around three primary routers: the Pet router handles listing submission, approval, retrieval, and deletion; the AdoptForm router manages the submission, retrieval, and deletion of adoption applications; and the Admin router handles administrator authentication. All write and delete operations on pet listings and adoption records require administrator-level authorization, enforced through JWT middleware on the relevant routes.

E. Authentication Workflow

During registration, users provide their email address which triggers an OTP dispatch through an email service. The OTP is validated server-side before the account is created, ensuring that only reachable email addresses are registered on the platform. Subsequent logins generate a JWT token that is stored client-side and attached to all subsequent API requests requiring authentication. This stateless approach avoids server-side session storage while maintaining request-level security.

V. RESULTS

The implementation produced a functional web platform with role-based access, admin review workflows, and secure user authentication. Representative results are summarized below.

A. User Interface

The design of the FindYourPet website took into consideration an effective user interface. Therefore, the site has been made user-friendly and easy to navigate without being too visually complex. There is a homepage showing available pets, along with a navigation bar giving the options of accessing the services, pets, and a contact page. The call-to-action button gives the users the chance to either adopt or list pets depending on their preference right on the landing page, as is illustrated in Fig. 3. The listing of a pet for adoption page has clearly outlined paths to

either adopt a pet or to register your pet on the site. To submit a pet for adoption, all that is needed is to fill the pet name, age, species type, picture, and location as shown in Fig. 4. Accessing the administrator features requires using a login page that is specifically designed for this purpose; hence, only administrators will have access to the admin functionalities as shown in Fig. 5. Once the administrator is logged into the system, he/she will see a list of pet submission request that show all the details such as the pet's owner contacts, breed, location, and the

B. Testing Observations

Testing was carried out across all core modules of the platform under expected usage conditions. The registration and OTP verification flow completed successfully across multiple

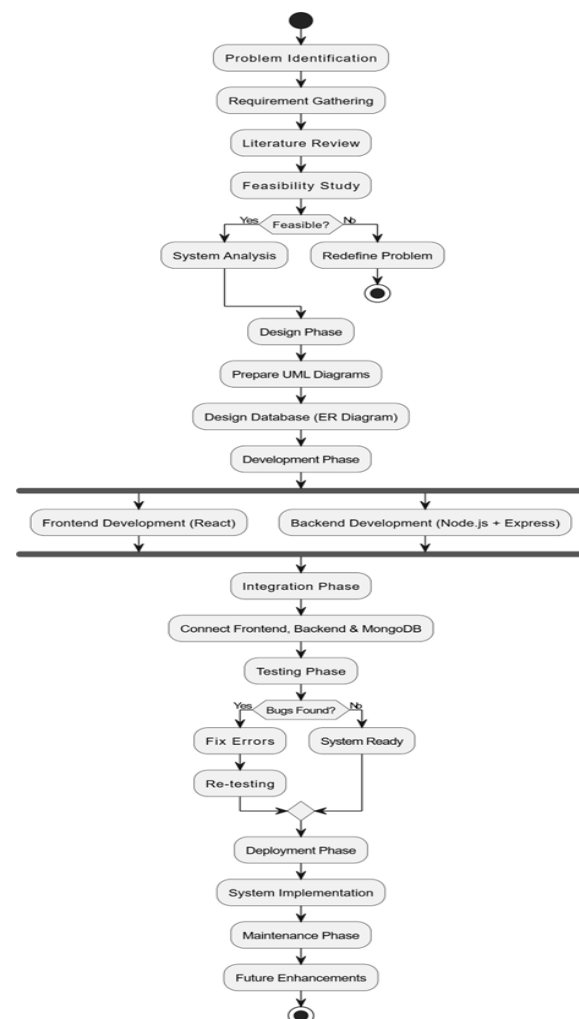


Fig. 3. Methodology Flowchart

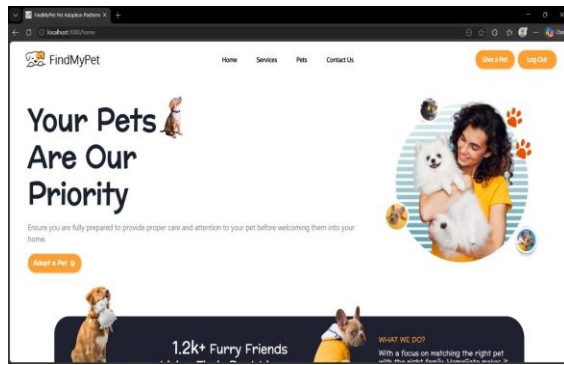


Fig. 4. Home Page

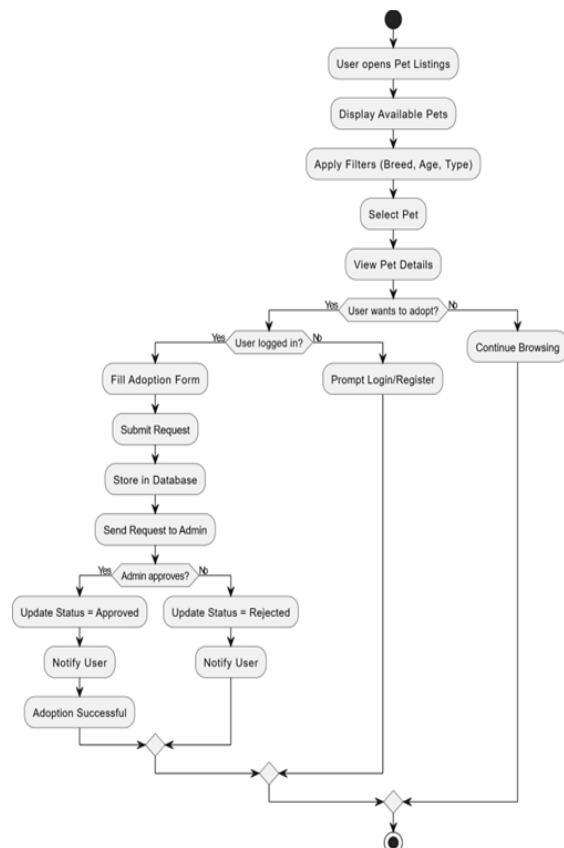


Fig. 5. Pet listing and adoption workflow

test accounts, with OTP delivery confirmed via email and invalid OTP entries correctly rejected by the server.

Pet listing submissions were correctly held in a pending state following submission and only appeared in the public listing after administrator approval. Rejected submissions were removed from the queue without affecting other listings. The adoption application form validated all required fields before submission and returned appropriate

error messages for incomplete entries.

The admin panel accurately reflected real-time changes to both pet listings and adoption requests. Approval and rejection

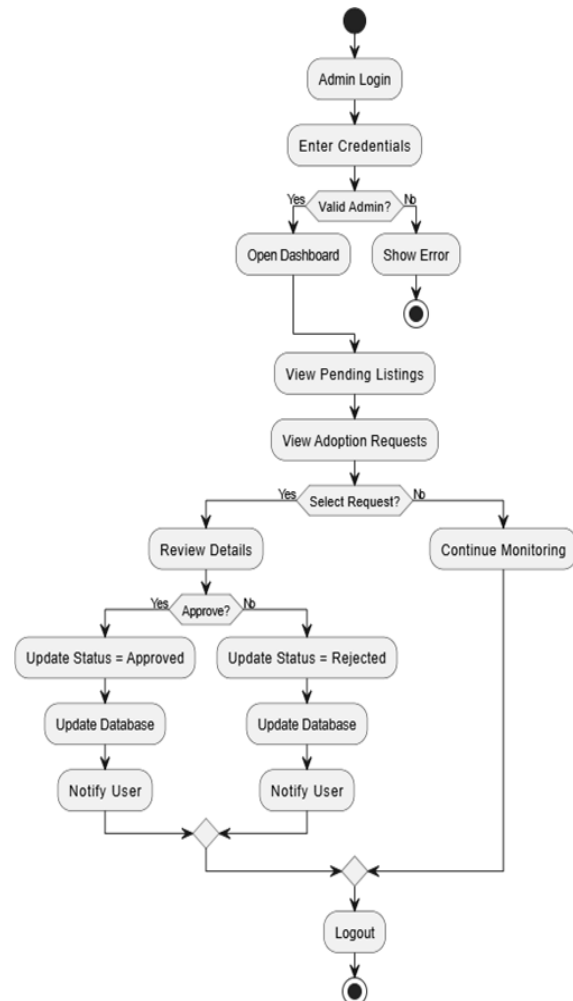


Fig. 6. Administrator Approval Workflow

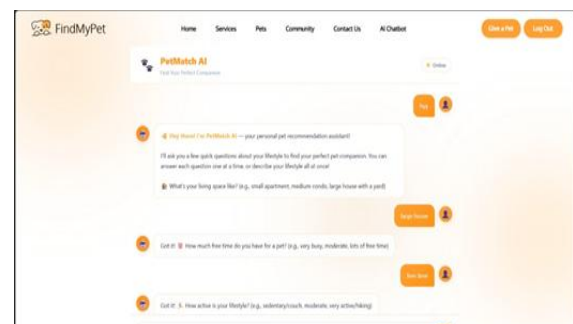


Fig. 7. AI Chatbot for Pet Recommendation

actions were processed immediately and the corresponding records were updated in the database

without requiring a page refresh. Adopted pets were correctly moved to the adoption history and no longer appeared in the active listings.

Overall, all tested modules performed consistently and as intended. No critical failures were observed during functional testing. Minor usability improvements, such as clearer confirmation messages and improved mobile layout on certain form pages, were noted for future iterations.

VI. CONCLUSION

From the implementation of FindYourPet, it is evident that there can be considerable enhancement of the pet adoption experience through an organized web application with the help of MERN. There was consistent functionality observed with all the key modules, and due to the use of the role-based system, the responsibilities were well distributed among all kinds of users. In addition to improving the overall experience associated with pet adoption, FindYourPet also integrates an AI recommendation chatbot powered by a microservice written using Python Flask, along with the Google Gemini API, which allows users to get recommendations on pets based on their personal preferences, living environment, and free time. This makes it unlikely for the adopted pets to be incompatible with the potential adopter or unsuitable to the owner's requirements, and thus ensures successful pet adoption. Moreover, by adding a community forum, FindYourPet not only facilitates knowledge exchange among users but also enhances its ability to serve as a digital solution for pet adoption in India.

REFERENCES

- [1] Petfinder, "Find a Pet," [Online]. Available: <https://www.petfinder.com>.
- [2] Adopt-a-Pet, "Adopt a Pet," [Online]. Available: <https://www.adoptapet.com>.
- [3] Rescue Me, "Animal Rescue," [Online]. Available: <https://www.rescueme.org>. [Accessed: 2025].
- [4] Best Friends Animal Society, "Pet Adoption," [Online]. Available: <https://bestfriends.org>.
- [5] Y. Liu, "Design and Implementation of Intelligent Pet Adoption and Rescue Platform," in Proc. IEEE Int. Conf. on Computer Science and Educational Informatization, 2024. [Online]. Available: <https://ieeexplore.ieee.org/document/10733211>.
- [6] N. M. Chaudhari, S. M. Jagtap, and D. Rahane, "Stray Love: Animal Adoption System," Int. J. for Research in Applied Science and Engineering Technology, vol. 13, 2025. doi: 10.22214/ijraset.2025.75513.
- [7] R. Alsuwailem et al., "Leen: Web-based Platform for Pet Adoption," in Proc. 2022 IEEE IAS Global Conference on Emerging Technologies (GlobConET), IEEE, pp. 775–780, 2022.
- [8] M. Luzardo et al., "Agapet: Supporting a Responsible Adoption Process for Stray and Abandoned Pets in Urban Areas," in Extended Abstracts of the 2023 CHI Conference on Human Factors in Computing Systems, ACM, 2023. doi: 10.1145/3544549.3583845.