Front End Engineering-II /Artificial Intelligence and Machine Learning

Project Report

Semester-IV (Batch-2022)

Title of the Project

Supervised By:

Mr. Rahul

Submitted By:

Arshdeep,2210991364 (G-22)

Arshdeep,221099166(G-22)

Ishika,2210991686(G-22)

Ishika,2210991688(G-22)

Department of Computer Science and Engineering
Chitkara University Institute of Engineering & Technology,
Chitkara University, Punjab

Abstract/Keywords

This report presents the development of a comprehensive Pet Adoption System utilizing the MERN stack. The system enables users to view available pets, create profiles, engage in real-time communication through a chatbox, and participate in discussions regarding pet adoption. The project focuses on creating a user-friendly interface that enhances the overall adoption experience for prospective pet owners. By leverageing modern web technologies, the system aims to provide a seamless and efficient platform for both adopters and shelters.

The project emphasizes the importance of an intuitive interface that enhances the overall user experience, making the adoption process more accessible and efficient. By leveraging modern web technologies, the Pet Adoption System aims to streamline the interaction between adopters and shelters, reducing the barriers often associated with traditional adoption methods.

Keywords: Pet Adoption System, MERN stack, Real-time chat, User profiles, Discussion forum.

Introduction to the Project

Background

The demand for pet adoption has significantly increased in recent years, with many individuals and families seeking to provide loving homes for pets in need. However, the traditional adoption process can often be cumbersome, involving multiple steps and a lack of effective communication channels between potential adopters and shelters. Many shelters still rely on outdated methods for managing pet listings and communicating with interested adopters, leading to missed opportunities and inefficiencies. This project aims to streamline the pet adoption process by providing a centralized platform that connects potential adopters with available pets, making the process more accessible and efficient..

Objective

- **Streamline the Adoption Process:** To create a centralized platform that simplifies the pet adoption process, making it easier for potential adopters to find and connect with available pets in a timely manner.
- **Enhance User Experience:** To design a user-friendly interface that allows users to navigate the system effortlessly, ensuring a positive experience for both adopters and shelters.
- **Facilitate Real-Time Communication:** To implement real-time chat functionality that enables direct communication between potential adopters and shelters, fostering immediate responses to inquiries and concerns.
- **Promote Community Engagement:** To develop a discussion forum where users can share their experiences, ask questions, and provide advice, thereby creating a supportive community for pet adopters.

Problem Statement

The pet adoption process is often fraught with challenges that hinder potential adopters from finding suitable pets and connecting with shelters effectively. One of the primary issues is fragmented communication, as many shelters and foster homes rely on outdated methods such as phone calls and emails. This reliance can lead to delays and misunderstandings, resulting in frustration for both potential adopters and shelter staff.

Additionally, potential adopters frequently struggle to discover pets that match their preferences and lifestyle due to a lack of centralized information. Existing systems may not provide comprehensive listings or detailed profiles, making it difficult for users to make informed decisions. The traditional adoption process can also be lengthy and complicated, involving multiple steps that may discourage potential adopters, leading to missed opportunities for both pets and adopters.

Requirement

Software Requirements

1. Frontend Development:
    - Language: JavaScript
    - Framework: React.js for building user interfaces
    - Styling Libraries: Tailwind CSS and Material-UI for responsive and modern design
2. Backend Development:
    - Language: Node.js for server-side scripting
    - Framework: Express.js for building RESTful APIs
    - Database: MongoDB for storing non-relational data, such as pet profiles, user accounts, chat logs, and notifications
    - Authentication: JSON Web Tokens (JWT) for secure user authentication
    - Real-time Communication: Socket.io for enabling real-time chat functionality
3. Development Tools:
    - Version Control: Git for source code management
    - Development Environment: Visual Studio Code or any preferred IDE
    - Testing Frameworks: Jest and Enzyme for unit and integration testing

Hardware Requirements

1. Server Requirements:
    - Operating System: Linux (Ubuntu preferred) for hosting the application
    - RAM: Minimum 8GB for smooth operation
    - Processor: Intel i5 or equivalent for handling server requests efficiently
    - Storage: SSD with at least 100GB of available space for database and application files

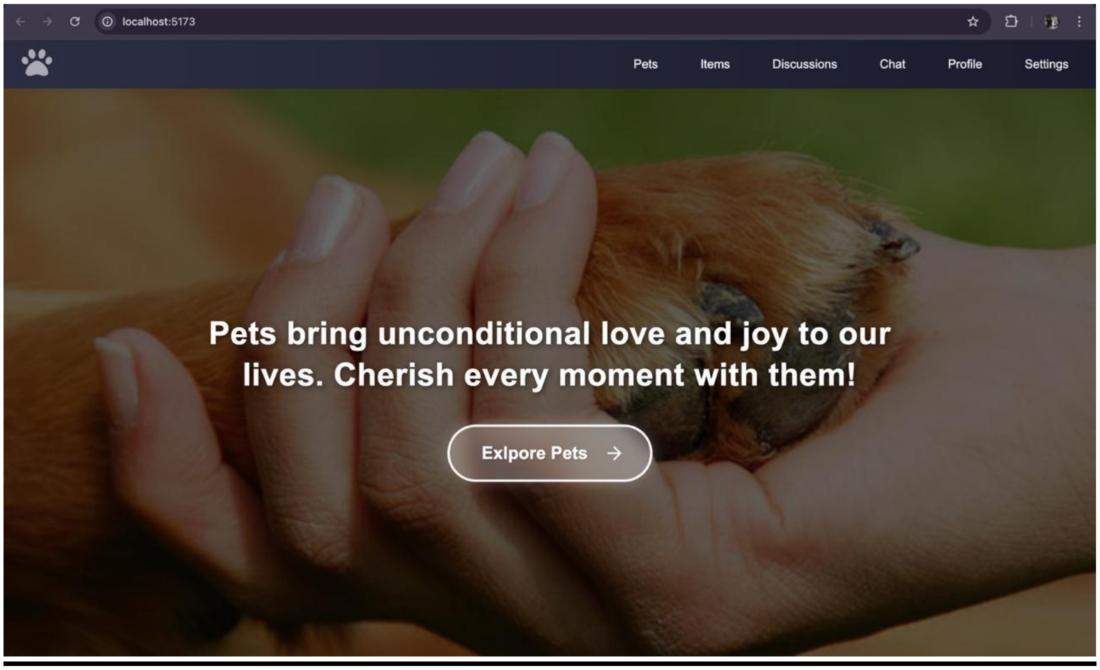
Proposed Design / Methodology

The development methodology for this project follows Agile principles, focusing on iterative development and continuous feedback. The project will be divided into multiple sprints, each delivering key components of the system. Regular meetings will be held to assess progress, gather feedback, and make necessary adjustments to the project plan. This approach ensures that the final product meets the needs of users and stakeholders effectively.

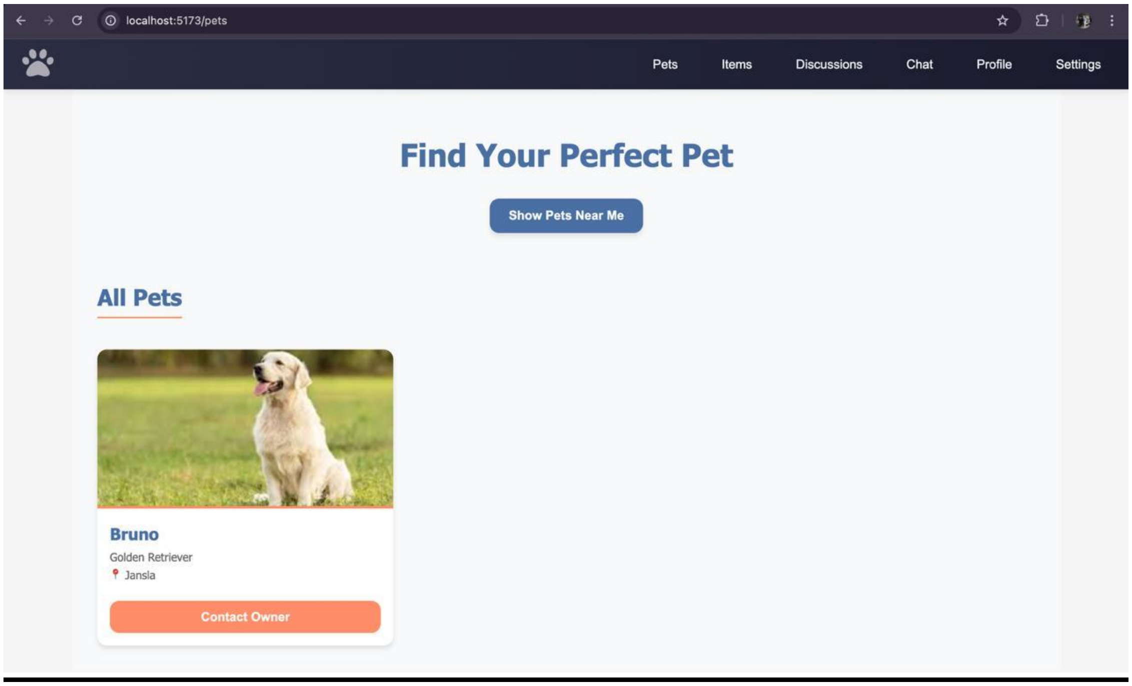
Key development methodologies include:

- **Agile Development**: Iterative and incremental development with regular feedback loops.
- **Test-Driven Development (TDD)**: Writing tests before implementing features to ensure code reliability and maintainability.
- **DevOps Practices**: Automated deployment, monitoring, and scaling using cloud services to handle varying loads.

Results



Pets bring unconditional love and joy to our lives. Cherish every moment with them!



Find Your Perfect Pet

All Pets

Bruno

Golden Retriever

📍 Jansla

References

- MongoDB Documentation. "NoSQL Database for Modern Applications." MongoDB, 2024. https://www.mongodb.com

- React.js Official Documentation. "A JavaScript Library for Building User Interfaces." Meta, 2024. https://reactjs.org

- Node.js Documentation. "JavaScript Runtime Built on Chrome's V8 JavaScript Engine." OpenJS Foundation, 2024. https://nodejs.org

- User Authentication "Understanding JSON Web Tokens (JWT)." Auth0, 2024. https://auth0.com/learn/json-web-tokens/

- Socket.io Documentation: "Real-time Application Framework." Socket.io, 2024. https://socket.io/docs/