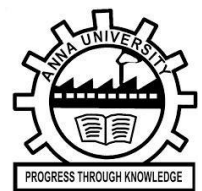


RESQTAIL – AN ANIMAL RESCUE PLATFORM EMPHASIZES THE CORE OF RESQUE

**DESIGN
THINKING
PROJECT**
April/May 2025

**PROJECT WITH DESIGN THINKING (PRODUCT/ SOFTWARE
DEVELOPMENT LIFE CYCLE) REPORT SUBMITTED IN PARTIAL
FULFILLMENT OF REQUIREMENTS FOR THE AWARD OF THE
DEGREE OF BACHELOR OF ENGINEERING IN
COMPUTER SCIENCE AND ENGINEERING
OF THE ANNA UNIVERSITY**



**PROJECT
WORK**

Submitted by
NAVEEN PRASANTH P - 722823104103
NIGUN KARTHI R - 722823104104
SANTHOSH SIVA S - 722823104312

BATCH
2023 – 2027

Under the Guidance of
Ms. J. Keerthika M.E., (Ph.D)

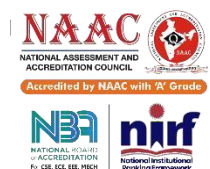
Assistant Professor
Computer Science and Engineering



Sri Eshwar College of Engineering
(An Autonomous Institution)

Kinathukadavu (Tk), Coimbatore - 641 202, Tamil Nadu

Approved by AICTE, New Delhi and Affiliated to Anna University, Chennai





Sri Eshwar
College of Engineering
Coimbatore | Tamilnadu
An Autonomous Institution
Affiliated to Anna University, Chennai



BONAFIDE CERTIFICATE

Certified that this Report titled **“RESQTAIL – An Animal Rescue Platform emphasizes the core of rescue”** is the bonafide work of

NAVEEN PRASANTH P	722823104103
NIGUN KARTHI R	722823104104
SANTHOSH SIVA S	722823104312

who carried out the project work under my supervision.

-----	-----
SIGNATURE	SIGNATURE
Dr. R. Subha, M.E, Ph.D. HEAD OF THE DEPARTMENT Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore – 641 202.	Ms. D. Mohana Priya M.E., SUPERVISOR Assistant Professor, Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore – 641 202.

Submitted for the **Autonomous Semester End Project with Design Thinking**

(Product/Software Development Life Cycle) Viva-Voce held on

INTERNAL EXAMINER

EXTERNAL EXAMINER



Sri Eshwar
College of Engineering
Coimbatore | Tamilnadu
An Autonomous Institution
Affiliated to Anna University, Chennai



QUALITY POLICY

To establish a system of Quality Enhancement, which would on a continuous basis evaluate and enhance the quality of teaching – learning, research and extension activities of the institution, leading to improvements in all processes, enabling the institution to attain excellence.

INSTITUTE VISION

To be recognized as a premier institution, grooming students into globally acknowledged engineering professionals.

INSTITUTE MISSION

- Providing outcome and value-based engineering education
- Nurturing research and entrepreneurial culture
- Enabling students to be industry ready and fulfill their career aspirations
- Grooming students through behavioral and leadership training programs
- Making students socially responsible

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DEPARTMENT VISION

To groom students into globally competent software professionals and meet the ever changing requirements of the industry.

DEPARTMENT MISSION

- Creating a quality academic environment with relevant IT infrastructure and empowering faculty and students with emerging technologies.
- Motivating staff and students to actively involve in lifelong learning and fostering research.
- Inculcating leadership and entrepreneurship skills in students.
- Generating opportunities for students to evolve as competent software professionals with societal consciousness.

PROGRAM EDUCATIONAL OBJECTIVES

PEO1: To prepare graduates for a career in software engineering

PEO2: To prepare students for higher studies, research, entrepreneurial and leadership roles by imparting the quality of lifelong learning

PEO3: To enable students to apply innovative solutions for real-life problems in computer science domain.

PROGRAM OUTCOMES

PO1: Engineering Knowledge: Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.

PO2: Problem Analysis: Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)

PO3: Design/Development of Solutions: Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)

PO4: Conduct Investigations of Complex Problems: Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).

PO5: Engineering Tool Usage: Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)

PO6: The Engineer and The World:Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).

PO7: Ethics: Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)

PO8: Individual and Collaborative Team work: Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.

PO9: Communication: Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences

PO10: Project Management and Finance: Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.

PO11: Life-Long Learning: Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

PROGRAM SPECIFIC OUTCOMES

PSO1. Demonstrate knowledge in open source technologies

PSO2. Develop innovative solutions by adapting emerging technologies for industry oriented applications.

PSO3. Implement SDLC principles for project/product development.

ACKNOWLEDGEMENT

ACKNOWLEDGEMENT

We thank God for his grace and blessing that have been showered upon us to bring success for this project. The success we have accomplished in our project would not have been possible without the timely help and guidance rendered to us by many people. We take this opportunity to thank them all.

We wish to express our deep gratitude **Mr. R. Mohanram**, Chairman, Sri Eshwar College of Engineering, Kondampatti for being kindly permitted us to carry on the project. He has an abiding interest in our wellbeing.

We are grateful to **Dr. Sudha Mohanram M.E., Ph.D** Principal, Sri Eshwar College of Engineering, Kondampatti for having made all provision available to us.

We are thankful to **Mr. R. Rajaram**, Director, Sri Eshwar College of Engineering, Kondampatti for all his support to complete the project effectively.

We would like to express our heartfelt gratitude and humble thanks to Head of the Department **Dr. R. Subha M.E., Ph.D** for having facilitated us to complete the project successfully.

With the deep sense of gratitude, we extend our earnest and sincere thanks to our project guide **Ms. D. Mohana Priya M.E.**, Assistant Professor, Department of Computer Science and Engineering, for his kind suggestion and co-operation for doing the project. Finally, we would like to extend our thanks to our teaching and non-teaching staffs those who helped us in bringing out project successfully.

TABLE OF CONTENT

TABLE OF CONTENT

ABSTRACT	i
LIST OF FIGURES	iii
LIST OF ABBREVIATIONS	v
1.INTRODUCTION.....	166
1.1. Objective.....	1
2.SYSTEM ANALYSIS AND DESIGN.....	2
2.1. Existing System.....	3
2.2. Problem Statement.....	3
3.PROPOSED SOLUTION.....	4
3.1. Overview.....	5
3.2. Block Diagram.....	5
4.SYSTEM SPECIFICATION.....	7
4.1. Software Requirements.....	8
5.PROJECT DESCRIPTION.....	9
5.1. Methodology.....	10
5.2. Implementation.....	11
6.IMPLEMENTATION.....	12
7.RESULTS.....	14
8.CONCLUSION.....	17
9.REFERENCE.....	19

ABSTRACT

With growing urbanization and increasing instances of abandoned or injured animals, there is a pressing need for technology-driven solutions to support timely and effective animal rescue efforts. Traditional rescue operations often rely on fragmented communication, manual reporting, and lack real-time coordination—resulting in delayed interventions and compromised animal welfare. The project titled **“ResQtail: Real-Time Rescue Coordination for Stray and Injured Animals”** addresses these challenges by offering a unified, web-based platform that empowers citizens to report animal emergencies with precision and immediacy. The system integrates GPS-based location tracking, image uploads, weather-aware planning, and machine learning models for injury prediction and animal recognition. Through intelligent automation, ResQtail connects public reports directly to nearby shelters and rescue volunteers, enabling faster response times and streamlined case management. Additionally, features such as real-time chat support, a Lost & Found portal, and push notifications foster an engaged, responsive community around animal protection. With an intuitive interface, admin dashboard, and success tracking, the platform enhances transparency, encourages volunteer participation, and strengthens rescue infrastructure. By digitizing and optimizing the rescue workflow, ResQtail promotes compassionate action, improves animal outcomes, and lays the groundwork for a scalable, tech-enabled ecosystem for animal welfare.

LIST OF FIGURES

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE NO.
3.1	BLOCK DIAGRAM	06
6.1	LOGIN PAGE	13
6.2	HOME PAGE	13
6.3	REPORT PAGE	14
6.4	VOLUNTEER PAGE	14

LIST OF ABBREVIATIONS

LIST OF ABBREVIATIONS

ACRONYM	ABBREVIATION
GUI	Graphical User Interface
API	Application Programming Interface
CNN	Convolutional Neural Network
AI	Artificial Intelligence

CHAPTER 1

INTRODUCTION

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVE

ResQtail: Real-Time Rescue Coordination for Stray and Injured Animals is an intelligent, community-driven platform developed to modernize and streamline the process of animal rescue by leveraging real-time data, geolocation services, and AI-based automation. The primary objective of this system is to provide a centralized, user-friendly web application that empowers citizens to report stray or injured animals while enabling rescue teams to respond quickly and efficiently through intelligent coordination. Much like how ride-hailing apps optimize transport logistics, ResQtail optimizes rescue operations by collecting critical field data—such as images, exact locations, and injury details—and routing it instantly to the nearest available rescuers or shelters. The platform integrates advanced features including ML-based animal recognition, injury prediction using CNN models, and weather-aware planning to support timely and informed decision-making. Users can report emergencies via an intuitive form, track ongoing rescue cases, and receive real-time updates and alerts. ResQtail also includes a Lost & Found board, volunteer registration portal, and success story tracking to foster active community participation and awareness. Key functionalities include real-time chat support, push notifications, interactive case maps, profile management, and an admin dashboard for case monitoring and resource allocation. Built with modern web technologies such as React, TypeScript, and Firebase, the application ensures scalability, security, and low-latency performance.

By digitizing the rescue workflow and introducing intelligent automation, ResQtail reduces response times, enhances coordination, and builds a connected ecosystem for animal protection. The system not only improves operational efficiency but also nurtures a culture of compassion and civic responsibility, paving the way for a smarter, more responsive approach to animal welfare.

CHAPTER 2

SYSTEM ANALYSIS AND DESIGN

CHAPTER 2

SYSTEM ANALYSIS AND DESIGN

2.1 EXISTING SCENARIO

Currently, animal rescue operations rely heavily on manual reporting methods such as phone calls, social media posts, or word-of-mouth communication, which often result in delays, incomplete information, and uncoordinated responses. There is typically no structured system to capture critical details like the exact location, condition of the animal, or photographic evidence at the time of reporting. Rescue teams often face challenges in prioritizing cases, tracking ongoing rescues, and reaching the scene promptly due to a lack of real-time updates and geolocation data. While some organizations use basic digital tools, most lack integration with AI for injury assessment, automated notifications, or centralized case management.

2.2 PROBLEM STATEMENT

Animal rescue operations often face significant challenges due to the absence of a unified, intelligent reporting and coordination system. The current reliance on manual communication methods—such as phone calls, social media alerts, or informal messaging—leads to inconsistent data collection, delayed responses, and poor coordination between citizens, rescuers, and shelters. Critical information like the animal's condition, exact location, and urgency is often incomplete or lost, making it difficult to prioritize and act promptly. Additionally, rescue teams lack real-time insights, centralized case tracking, and intelligent tools for injury assessment, which hampers efficiency and increases response times.

There is a pressing need for a smart, community-enabled platform that integrates AI-based injury recognition, geolocation services, and real-time communication to streamline animal rescue workflows. Without such a system, valuable time is lost, animals remain unattended, and the rescue ecosystem remains fragmented and reactive.

CHAPTER 3

PROPOSED SOLUTION

CHAPTER 3

PROPOSED SOLUTION

3.1 OVERVIEW

The proposed solution aims to transform animal rescue operations through a centralized, AI-enhanced, web-based coordination platform. The project—**ResQtail: Real-Time Rescue Coordination for Stray and Injured Animals**—was developed to enable fast, efficient, and data-driven animal rescue by combining real-time geolocation tracking, machine learning models, and community engagement tools. Unlike traditional rescue efforts that rely on manual communication and unstructured data, ResQtail offers a smart, end-to-end rescue ecosystem accessible to both the public and rescuers.

Users can report animal emergencies via a user-friendly form that captures images, health condition, and exact GPS coordinates. The platform employs machine learning algorithms for injury prediction and animal recognition, while real-time weather data integration helps responders plan effectively. A dynamic dashboard provides live case tracking, rescue status updates, and volunteer coordination. Additional features include a Lost & Found board, push notification system, real-time chat support, and a success stories module that highlights the platform's impact.

Built using a scalable, modular architecture with React (TypeScript) on the frontend and Firebase services on the backend, ResQtail ensures seamless integration of AI components, real-time data flows, and interactive user experiences. This innovative solution modernizes how animal rescues are reported and managed, reduces response time, enhances coordination, and fosters a compassionate, connected community dedicated to animal welfare.

3.2 BLOCK DIAGRAM



Figure 3.1 – Block Diagram

CHAPTER 4

SYSTEM SPECIFICATION

CHAPTER 4

SYSTEM SPECIFICATION

4.1 SOFTWARE REQUIREMENTS

- ❖ Operating System : Windows 10
- ❖ IDE : Visual Studio Code
- ❖ Frontend Framework : React.js
- ❖ Backend Framework : Node.js, Firebase Cloud Functions and Firestore
- ❖ Database : Firebase Firestore (NoSQL, real-time)
- ❖ Authentication : JWT (JSON Web Token)
- ❖ AI Frameworks : TensorFlow (AI Chatbot),
- ❖ **Geolocation Services:** Google Maps API and Firebase GeoFirestore for location tracking
- ❖ Package Manager : npm for frontend & backend

CHAPTER 5

PROJECT DESCRIPTION

CHAPTER 5

PROJECT DESCRIPTION

5.1 METHODOLOGY

Our application is divided into distinct modules to efficiently manage specific tasks. The frontend is built with React.js and TypeScript, using React Router for navigation and Material-UI for styling and responsive design. We use React's Context API for state management and Firebase SDKs to communicate with backend services. The backend is powered by Firebase Cloud Functions and Firestore, handling data storage, authentication, and server-side logic. Authentication is securely managed through Firebase Authentication, supporting multiple login methods. Machine learning models for injury detection and animal recognition are integrated using TensorFlow.js on the frontend or deployed as Firebase functions with Python TensorFlow models. Real-time updates and notifications use Firestore's real-time listeners and Firebase Cloud Messaging for push alerts. Location services are implemented via Google Maps API and GeoFirestore to capture and visualize animal rescue locations. The chat system and other live features leverage Firestore's real-time capabilities for instant communication. We ensure meaningful error handling and user notifications throughout the app. This modular architecture provides scalability, maintainability, and a smooth user experience across all features.

Key Modules:

- flask
- flask_sqlalchemy
- flask_jwt_extended
- flask_uploads
- flask_mail / smtplib
- flask_cors
- react-router-dom

5.2 IMPLEMENTATION


To develop the ResQtail platform, essential packages and services were integrated using npm (for the frontend) and Firebase SDKs (for the backend). Key tools include: react, react-router-dom for routing, firebase for real-time database, authentication, and cloud functions, react-toastify for in-app notifications, and React's Context API for global state management. The frontend is styled using Material-UI for responsive design, while animations and visual feedback are implemented with Framer Motion. Users can report injured or stray animals by filling out a dynamic form that captures images, health information, and exact GPS location using the Google Maps API. Images are processed using AI models built with TensorFlow.js (client-side) or TensorFlow (Python), deployed through Firebase Cloud Functions to identify animal type and predict injury severity. Reports and status updates are stored in Firebase Firestore, enabling real-time syncing across the platform. Authentication is securely handled by Firebase Authentication, supporting email/password and Google sign-in methods. Upon successful report submissions, users receive instant feedback via push alerts powered by Firebase Cloud Messaging (FCM) and visual updates through react-toastify. The platform's dashboard displays live rescue case statuses, rescue team assignments, and an interactive map with reported animal locations using GeoFirestore. A real-time chat module enables coordination between rescuers and citizens through Firestore's real-time listeners. All application data, including rescue case history, user information, and AI predictions, is managed centrally using React's Context API, ensuring efficient state propagation across all components. This modular, cloud-native implementation promotes scalability, reduces latency, and allows seamless integration of future AI models or features.

CHAPTER 6

IMPLEMENTATION

```
// backend/server.js
import express from 'express';
import mongoose from 'mongoose';
import cors from 'cors';
import dotenv from 'dotenv';
import sosRoutes from './routes/sos.js'; // correct if using ES Modules (.js is needed)



dotenv.config(); // Load env variables first


const app = express(); //  create app first


const PORT = process.env.PORT || 5000;

// Middleware
app.use(cors());
app.use(express.json());

// API routes
app.use('/api/sos', sosRoutes);

// MongoDB connection
mongoose.connect(process.env.MONGO_URI, {
  useNewUrlParser: true,
  useUnifiedTopology: true,
})
.then(() => console.log( MongoDB connected))
.catch((err) => console.error( MongoDB connection error:', err));

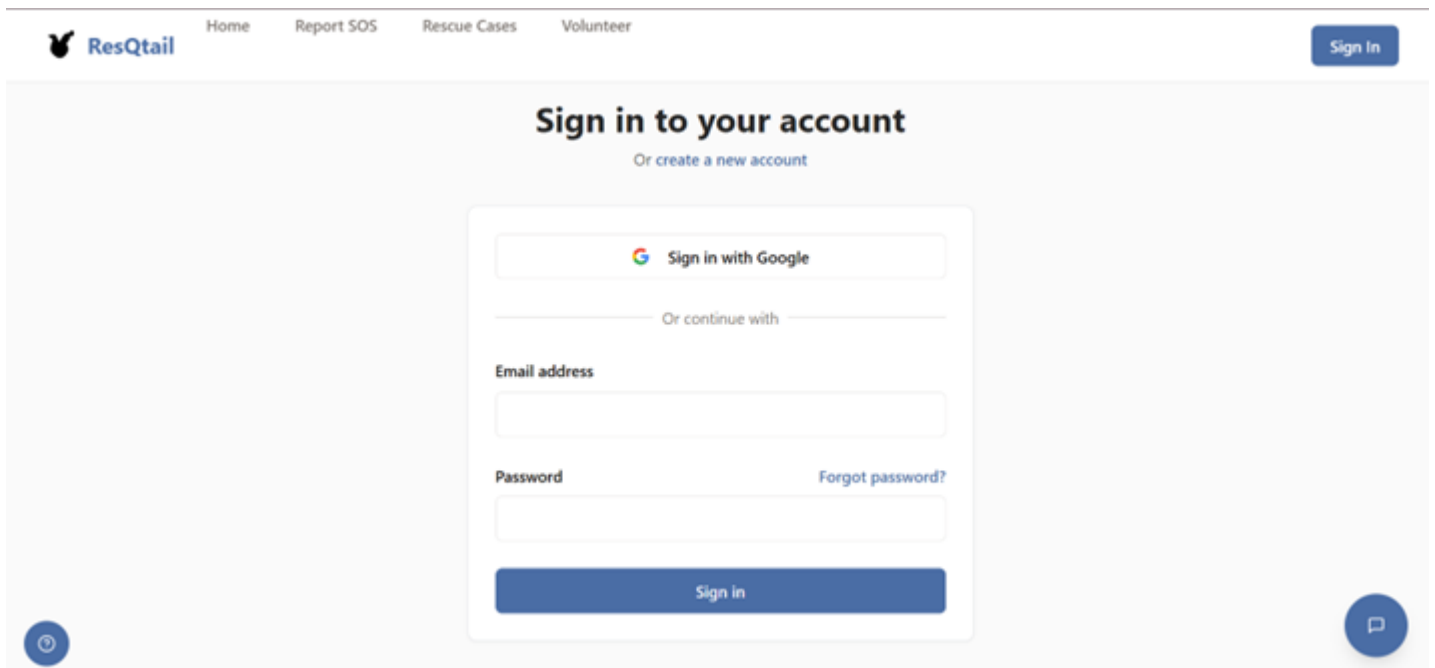
// Example route
app.get('/', (req, res) => {
  res.send( Backend is running!);
});

app.listen(PORT, () => {
  console.log( Server running on http://localhost:${PORT}`);
});
```

CHAPTER 7
RESULTS

CHAPTER 7

RESULTS



The login page features a navigation bar at the top with the ResQtail logo and links for Home, Report SOS, Rescue Cases, and Volunteer. A 'Sign In' button is located in the top right corner. The main heading is 'Sign in to your account', with a sub-link 'Or create a new account'. The login form includes a 'Sign in with Google' button, a 'Or continue with' separator, an 'Email address' field, a 'Password' field, and a 'Forgot password?' link. A 'Sign in' button is at the bottom of the form. Floating action buttons for a camera and a chat icon are positioned at the bottom left and right respectively.

ResQtail Home Report SOS Rescue Cases Volunteer Sign In

Sign in to your account

Or create a new account

Sign in with Google

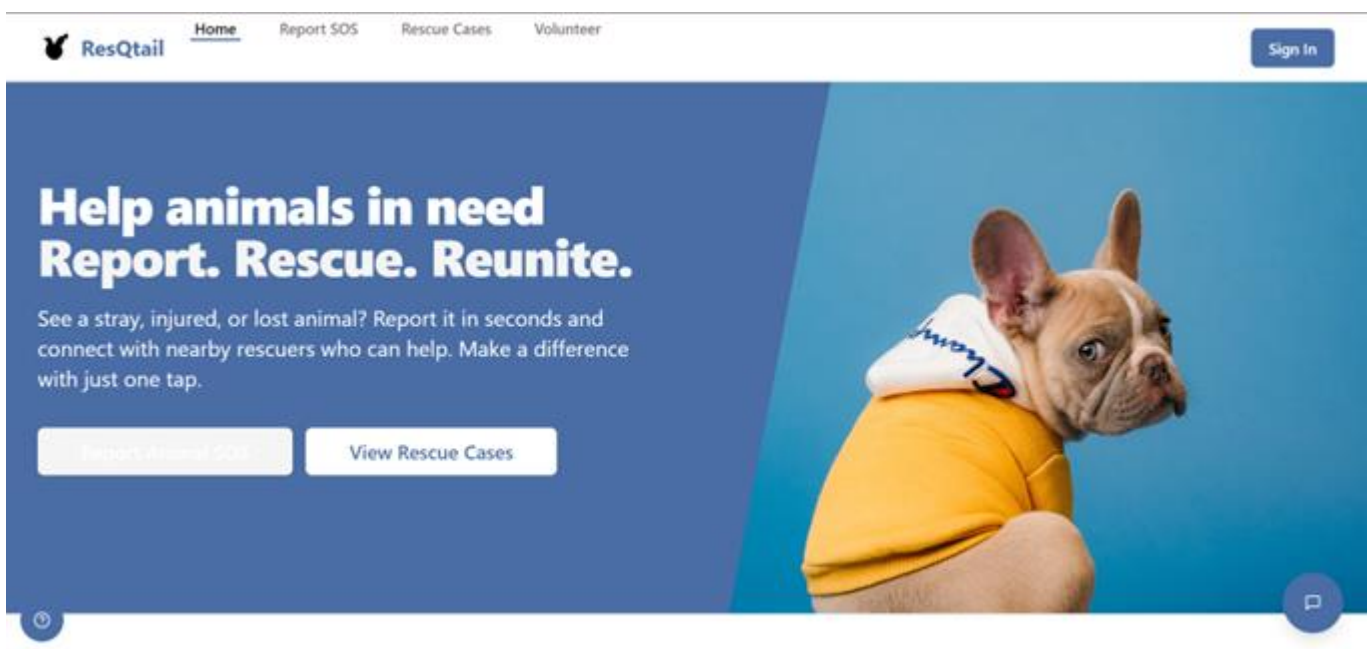
Or continue with

Email address

Password [Forgot password?](#)

Sign in

Figure 6.1 Login Page



The home page has a navigation bar with the ResQtail logo and links for Home, Report SOS, Rescue Cases, and Volunteer. A 'Sign In' button is in the top right. The main banner features the text 'Help animals in need Report. Rescue. Reunite.' followed by a description: 'See a stray, injured, or lost animal? Report it in seconds and connect with nearby rescuers who can help. Make a difference with just one tap.' Below this are two buttons: 'Report Animal SOS' and 'View Rescue Cases'. On the right side of the banner is a photo of a French Bulldog wearing a yellow hoodie. Floating action buttons for a camera and a chat icon are at the bottom left and right.


ResQtail Home Report SOS Rescue Cases Volunteer Sign In

Help animals in need Report. Rescue. Reunite.

See a stray, injured, or lost animal? Report it in seconds and connect with nearby rescuers who can help. Make a difference with just one tap.

Report Animal SOS View Rescue Cases

Figure 6.2 – Home Page



HomeReport SOSRescue CasesVolunteer

Report Animal SOS

Found an animal in need? Fill out this form to alert nearby rescuers.

Report an Animal in Need

Animal Type *

Select animal type


Emergency Type *

Select emergency type

Description *

Describe the animal and situation in detail. Include the animal's condition, behavior, and any landmarks to help rescuers find it.

Figure 6.3 – Report Page



HomeReport SOSRescue CasesVolunteer

Sign In


Become a Rescuer

Join our network of dedicated animal lovers who respond to rescue calls in their area. As a rescuer, you'll be notified when animals need help near you.

Benefits


Requirements

Application




Get Alerts

Receive notifications when animals are reported in your area, so you can respond quickly if you're available.



Join a Network

Connect with other animal rescuers in your community and coordinate efforts to help more animals in need.



Make a Difference

Track your impact with records of your rescues and receive recognition for your lifesaving work.

Apply Now

Figure 6.4 Volunteer Page

CHAPTER 8

CONCLUSION

CHAPTER 8

CONCLUSION

The **ResQtail** platform effectively addresses the critical challenges in animal rescue operations by providing an intelligent, centralized system for real-time reporting, injury assessment, and coordination. By integrating AI-based animal and injury recognition, geolocation services, and real-time communication tools, the platform empowers communities and responders to act swiftly and efficiently. Users benefit from a streamlined reporting process, instant updates, and clear rescue tracking, while responders gain data-driven insights for prioritizing actions and improving outcomes. The integration of modern web technologies and machine learning transforms traditional rescue efforts into a proactive, well-organized, and compassionate ecosystem. ResQtail demonstrates the powerful impact of technology in enhancing animal welfare, reducing response times, and fostering a more connected, humane society.

the platform empowers communities and responders to act swiftly and efficiently. Users benefit from a streamlined reporting process, instant updates, and clear rescue tracking, while responders gain data-driven insights for prioritizing actions and improving outcomes. The integration of modern web technologies and machine learning transforms traditional rescue efforts into a proactive, well-organized, and compassionate ecosystem. ResQtail demonstrates the powerful impact of technology in enhancing animal welfare, reducing response times, and fostering a more connected, humane society.

CHAPTER 9
REFERENCES

CHAPTER 9

REFERENCES

- [1] Srivastava, R., and Kumar, P. (2023). AI-Based Animal Welfare and Rescue System Using Deep Learning and IoT Integration. *Artificial Intelligence in Agriculture*, 11, 92–104.
<https://doi.org/10.1016/j.aiia.2023.02.006>
- [2] Liu, M., Zhao, X., and Nguyen, T. (2024). Real-Time Animal Detection in Urban Areas Using Deep Convolutional Networks. *Computer Vision and Image Understanding*, 230, 103777.
<https://doi.org/10.1016/j.cviu.2023.103777>
- [3] Rahman, M., Alam, S., and Das, T. (2023). Image-Based Injury Severity Assessment for Wildlife Using CNNs. *Ecological Informatics*, 74, 101874.
<https://doi.org/10.1016/j.ecoinf.2023.101874>
- [4] Chen, L., and Zhang, Y. (2022). Geolocation Technologies in Emergency Response Systems: A Survey and Outlook. *Journal of Location-Based Services*, 16(2), 105–120. <https://doi.org/10.1080/17489725.2022.2044891>
- [5] Ahmed, S., and Joseph, J. (2021). Real-Time Communication Technologies in Mobile Rescue Apps. *International Journal of Web and Grid Services*, 17(4), 289–307.
<https://doi.org/10.1504/IJWGS.2021.119379>
- [6] Pal, A., Kumar, R., and Verma, N. (2022). Firebase-Powered Mobile Platforms for Community Engagement and Real-Time Data Synchronization. *Procedia Computer Science*, 199, 243–251.
<https://doi.org/10.1016/j.procs.2022.01.028>