Full Stack Development Documentation - Poultry Health App

Project Title: Poultry Disease Detection Using Deep Learning (Poultry Health App)

Team ID: LTVIP2025TMID44725

Team Size: 4

Team Leader: Kandera Naga Prudhvi Sai

Project Duration: June 26–29, 2025

Location: Ongole, Andhra Pradesh

Institution: Rise Krishna Sai Prakasam Group of Institutions

Team Members

- Kandera Naga Prudhvi Sai – Full Stack Developer & AI Model Integrator
- Poondla Divya Lakshmi – Team Member
- Medida Gangothri – Team Member
- Pasupuleti Venkata Aneesha – Team Member

1. Project Overview

The Poultry Health App is a web-based system designed to detect poultry diseases from images using a trained deep learning model. It aims to help farmers and veterinary professionals quickly identify symptoms and take timely actions. The application classifies poultry conditions into four categories: Healthy, Salmonella, Newcastle Disease, and Coccidiosis.

- Key Features:
- Upload-based disease prediction using deep learning
- AI-powered real-time classification
- Educational pages for training and awareness
- Responsive, user-friendly interface
- Downloadable PDF reports of results

2. Architecture

Frontend:
- HTML5, CSS3, JavaScript
- Templates styled with Tailwind CSS and responsive design
- Animation support using Lottie

Backend:
- Flask (Python)
- TensorFlow/Keras model integration
- Secure file handling with werkzeug

Model:
- Trained `.h5` file using image classification via Transfer Learning
- Handles predictions based on uploaded images

File Storage:
- Uploaded images stored in `static/uploads`
- Reports generated in PDF format and saved in the same folder

3. Folder Structure

```
project-root/
├── app.py
├── Poultry_Disease.h5
├── requirements.txt
├── Procfile
├── static/
│   └── uploads/
├── templates/
│   ├── index.html
│   ├── about.html
│   ├── contact.html
│   ├── training.html
│   ├── result.html
│   ├── treatment_guidelines.html
│   ├── vet_resources.html
│   └── download_reports.html
├── poultry_env/ (ignored in Git)
└── .gitignore
```

4. Setup Instructions

Prerequisites:
- Python 3.10+
- pip

Installation:
```bash
# Clone repository
cd your-folder
python -m venv poultry_env
poultry_env\Scripts\activate
pip install -r requirements.txt
python app.py
```

5. API Routes

- `GET /` → Homepage with image upload form
- `GET /about` → Info page
- `GET /contact` → Contact page
- `GET /training` → Education content
- `POST /predict` → Accepts image and returns prediction result

6. Known Issues

- Model file (`.h5`) not included in repo due to size (>100MB)
- No authentication or database integration
- Only basic input validation

7. Future Enhancements

- Add login system for users and admins
- Store prediction history with timestamps
- Use Git LFS for model file
- Integrate with third-party veterinary APIs
- Mobile version of the application

8. Acknowledgements

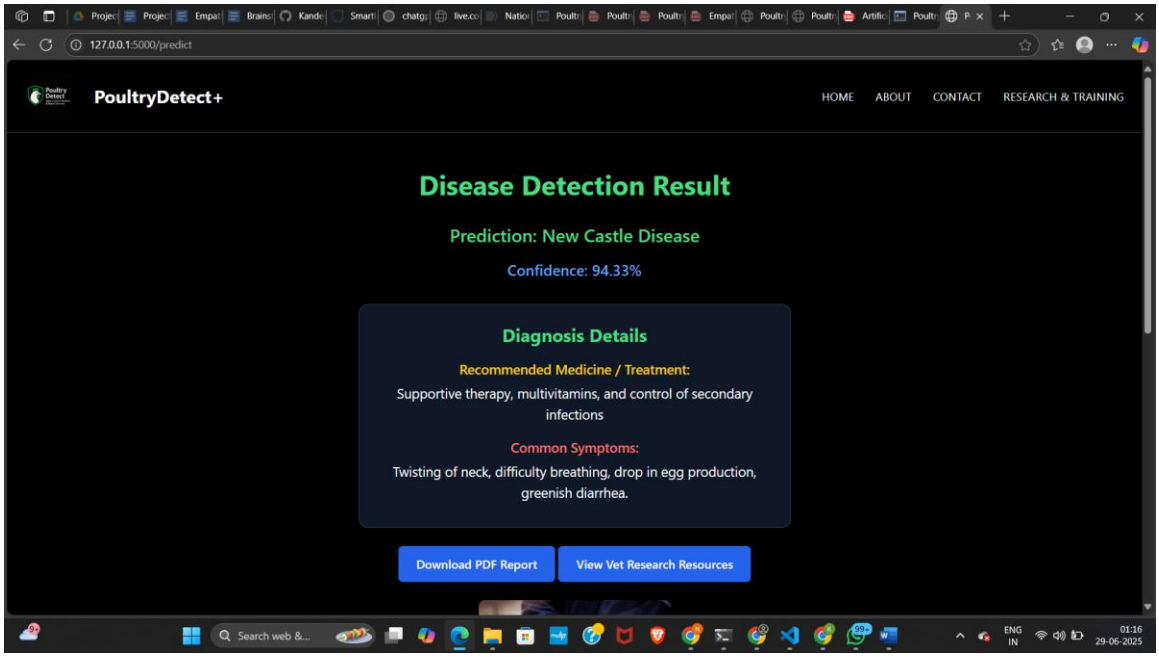
Thanks to Rise Krishna Sai Prakasam Group of Institutions for support, and to all team members for contributions.

GitHub Repo: https://github.com/KanderaNagaPrudhviSai/Poultry-Health-App-Final

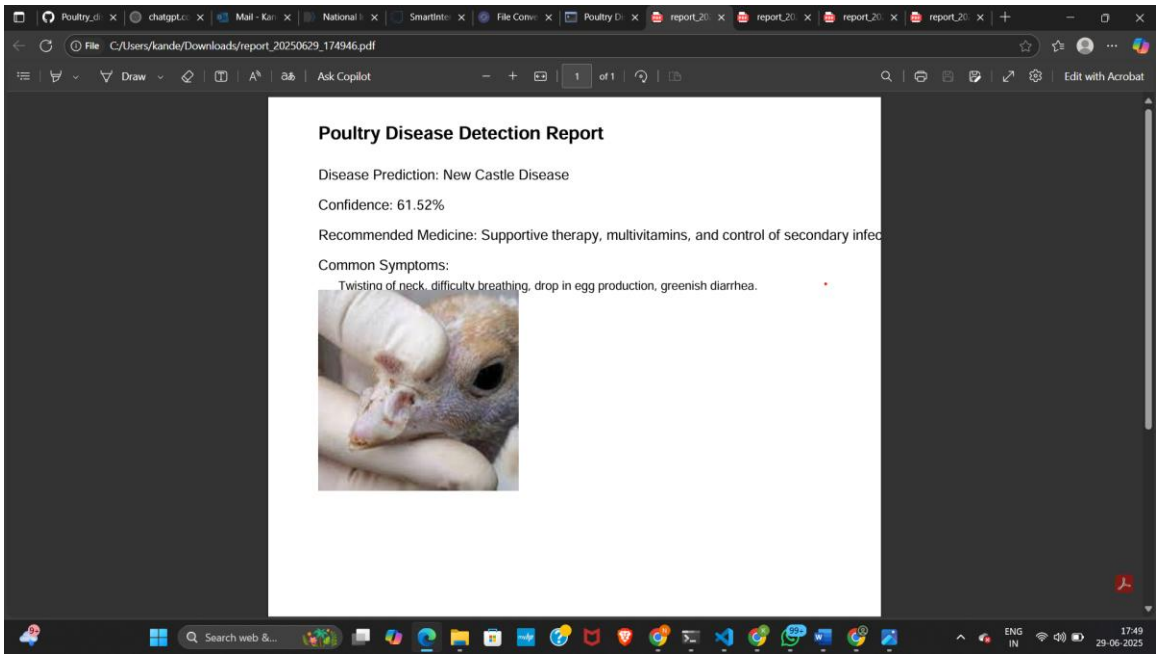
Homepage


127.0.0.1:5000
PoultryDetect+
HOME
ABOUT
CONTACT
RESEARCH & TRAINING
Poultry Disease Identifier & Report Generator
Choose File
download.jpg
Upload Image
Search web &...
ENG
IN
01:15
29-06-2025

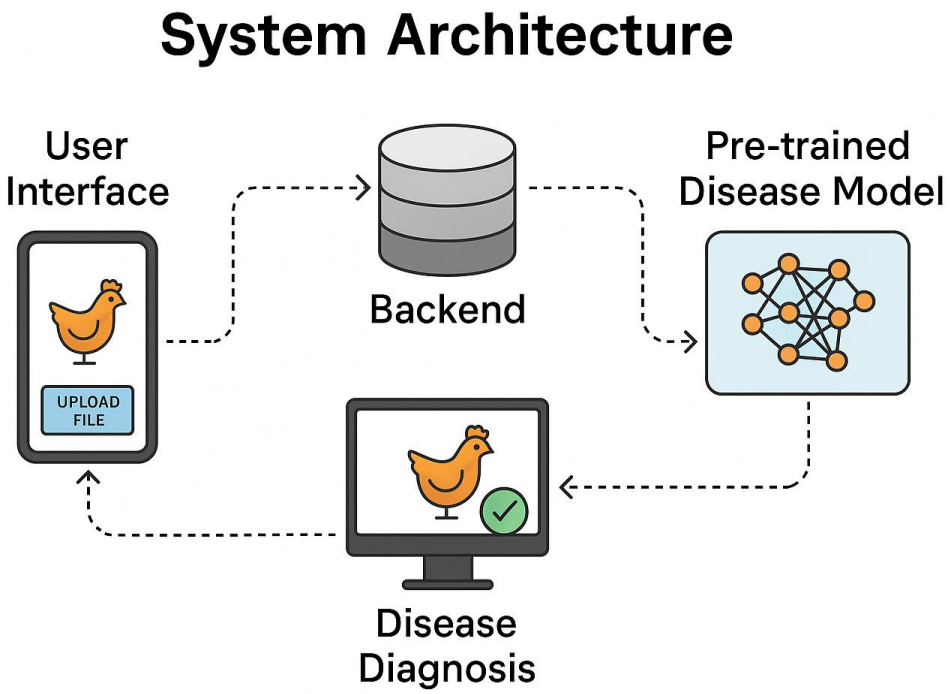
Prediction Result



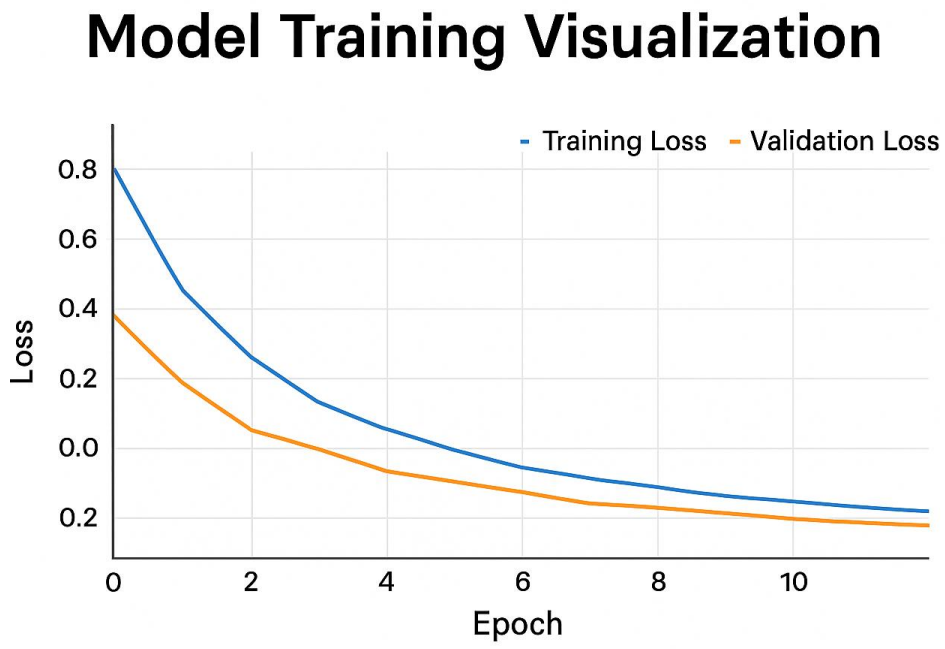
PDF Report


C:/Users/kande/Downloads/report_20250629_174946.pdf
Ask Copilot
Edit with Acrobat
Poultry Disease Detection Report
Disease Prediction: New Castle Disease
Confidence: 61.52%
Recommended Medicine: Supportive therapy, multivitamins, and control of secondary infec
Common Symptoms:
Twisting of neck, difficulty breathing, drop in egg production, greenish diarrhea.
Search web &...
ENG
IN
17:49
29-06-2025

System Architecture Diagram



Model Training Visualization



Mobile View / Responsive Design

9. Authentication

Authentication is not implemented in the current version of the application. Future updates may include login support for users and administrators to track prediction history, secure user data, and manage roles.

10. Testing

Manual testing was conducted throughout the development process. The application was tested for:

- Uploading and processing image files
- Displaying correct disease predictions
- Navigating between pages (About, Contact, Training, etc.)
- Generating and downloading PDF reports

No automated testing frameworks were used. All testing was performed on Windows using modern browsers like Chrome and Edge.