



International Journal of Research Publication and Reviews

Journal Homepage: www.ijrpr.com ISSN 2582-7421

CocoCare AI: AI-Powered Coconut Disease Detection and Farm Management Assistant

Mrs. Sabeetha M, Sadanasree C M, Rithika Sree U, Rooba K, Rakshika M A, Preethi Krishna B

Department of Computer Science and Engineering, Bachelor of Engineering, Sri Shakthi Institute of Engineering and Technology Coimbatore-641062

ABSTRACT

This project presents CocoCare AI, a web-based system that helps coconut farmers identify tree diseases from leaf images and manage routine farm activities in one place. The user uploads or captures a photo of a coconut leaf, and the system sends it to a pre-trained object detection model that returns the detected disease class and a confidence score. Each result is shown with a status badge, a confidence ring, a per-class confidence breakdown and a pre-written treatment recommendation.

The system includes a multilingual AI chatbot that answers farming questions in English, Tamil and Hindi. Disease descriptions and treatment advice are stored as fixed, reviewed text so that factual agricultural content is never generated live by a language model. Secure sign-up and login, a scan history that is private to each user and a summary dashboard complete the application. By making early disease identification and farm record keeping simple and available in regional languages, the system supports first-time and small-scale coconut farmers.

Keywords — Coconut Disease Detection, Object Detection, Plant Pathology, Multilingual Chatbot, Smart Agriculture, Farm Management

INTRODUCTION

- Coconut is a major plantation crop, and many families depend on it for their income, especially in southern India.
- Diseases such as bud rot, leaf blight, leaf rot, stem bleeding and lethal yellowing reduce yield, and some can kill a tree if they are not found early.
- Farmers usually identify diseases by eye. This depends on experience, and first-time growers often cannot tell one disease from another.
- Expert advice from agricultural officers is not always available at the moment it is needed.
- Most digital tools are in English only, which limits their use for farmers who are more comfortable in Tamil or Hindi.
- The proposed system uses artificial intelligence to give a quick first diagnosis from a photograph and pairs it with a conversational assistant and farm management tools.

OBJECTIVE:

- To design and develop a web application that detects coconut tree diseases from leaf images.
- To show the result in a form a non-technical farmer can understand, with the disease name, confidence score and recommended action.
- To provide an AI chatbot that answers coconut farming questions in English, Tamil and Hindi.
- To translate the whole interface, including disease information, into the language the user selects.
- To store each user's scan history securely and show summary statistics.
- To support basic farm management such as farm profile, planning and record keeping.

LITERATURE SURVEY

1. Deep Learning for Plant Disease Detection

Research on image based plant disease detection shows that convolutional neural networks can classify leaf diseases with high accuracy when trained on

labelled images. Mohanty et al. demonstrated this on a large public leaf dataset, which made image based diagnosis a standard approach.

2. Coconut-Specific Detection Studies

Several studies have applied deep learning to coconut trees, including work on coconut leaf spot classification, disease and pest detection, and mobile friendly models such as DeepSeqCoco. These studies show that coconut diseases can be recognised from images, but most of them stop at a research prototype and do not provide a complete farmer facing application.

3. Object Detection Models

Object detection models such as the YOLO family locate and classify affected regions in a single pass. This suits leaf images where symptoms appear as patches, because the output shows both what the disease is and where it appears.

4. Language Models in Agriculture Advisory

Large language models can answer farming questions conversationally and in regional languages. The literature and practice also warn that they can produce confident but wrong numbers, such as dosages. This project therefore limits the language model to open conversation and keeps all factual treatment content as fixed text.

METHODOLOGY

Input Processing

The user logs in and uploads a leaf image or takes a photo with the device camera. Accepted formats are PNG, JPG and JPEG.

Image Submission

The frontend sends the image to the FastAPI backend as multipart form data, together with the selected language.

Disease Detection

The backend forwards the image to a hosted pre trained coconut tree disease detection model through the Roboflow inference SDK.

Result Processing

The backend selects the detection with the highest confidence as the predicted class, sorts all detections for the confidence breakdown and maps the class to a status

Recommendation Lookup

The description and treatment text for the predicted class is read from a fixed dictionary available in English, Tamil and Hindi. No language model is used for this step.

Storage

Each scan is saved in SQLite with the user, timestamp, predicted class, confidence and status.

Chatbot Interaction

The user's message, recent conversation and latest scan result are sent to the chat endpoint. A Llama 3.3 model served through Groq replies in the selected language, and the system prompt keeps it on coconut and farming topics.

Display

The frontend shows the result card, history and dashboard statistics in the selected language.

EXISTING SYSTEM

1. Visual Inspection by Farmers How it works:

Farmers look at leaves and trunks and compare symptoms with their experience.

Drawbacks:

- Depends on personal experience
- Early symptoms are easily missed
- Similar diseases are confused with each other

2. Agricultural Extension Visits How It Works:

Officers or scientists inspect the plantation and advise on treatment.

Drawbacks:

- Not available immediately
- Limited coverage in rural areas
- Advice is not recorded for later reference

3. General Plant Identification Applications How It Works:

A photo is matched against a broad database of plants and diseases.

Drawbacks:

- Not specific to coconut diseases
- Mostly English-only
- No farm records or history

4. Laboratory Testing How It Works:

Samples are tested, for example by PCR for phytoplasma diseases.

Drawbacks:

- Slow and costly
- Needs trained staff and equipment
- Impractical for routine screening

PROPOSED SYSTEM

1. Image Input and Disease Detection Objective:

Image Input and Disease Detection

How It Works:

- The user uploads an image or uses the camera.
- The image is checked and sent to the backend.
- The hosted detection model returns the disease class and confidence.
- Low or empty detections are reported as uncertain.

2. Result and Recommendation Interface Objective:

Present the result in a way a farmer understands.

How It Works:

- A coloured status badge shows healthy or disease.
- A circular ring shows the confidence percentage.
- A bar breakdown shows confidence for every class.
- A fixed, reviewed treatment recommendation is displayed with a disclaimer.

3. Multilingual AI Chatbot Objective:

Answer farming questions in the farmer's own language. **How It Works:**

- A floating chat button is available on every screen after login.
- The reply language follows the selected language: English, Tamil or Hindi.
- The latest scan result is passed as context so the answer refers to the user's actual case.

- The assistant redirects unrelated questions and avoids giving exact dosages.

4. Whole-Interface Language Objective:

Make the entire application usable in Tamil and Hindi.

How It Works:

- A language selector in the header changes all labels and messages.
- Disease names, descriptions and treatments come from a fixed, human-written translation table.
- The chosen language is saved to the user's account.

5. Authentication and User History Objective:

Keep each farmer's data private

How It Works:

- Users create an account and log in with email and password.
- Passwords are stored as bcrypt hashes and sessions use JWT tokens.
- Scan history and statistics are filtered per user.

6. Give an overview of crop health Objective:

Give an overview of crop health.

How It Works:

- Cards show total scans, healthy percentage and recent alerts.
- A history screen lists past scans with date, status and confidence.
- Filters separate healthy results from disease results.

7. Farm Management Modules Objective:

Extend the application beyond diagnosis.

How It Works:

- A farm profile stores area, location, soil type, water source and tree count.
- A planner estimates the number of trees for a given area and spacing, clearly labelled as an estimate.
- Task, irrigation, soil, expense and growth records are stored for later review.
- Recommendations that need agricultural data use fixed reference tables and always show a disclaimer to confirm with local agricultural guidance.

SYSTEM REQUIREMENT

Hardware Requirements

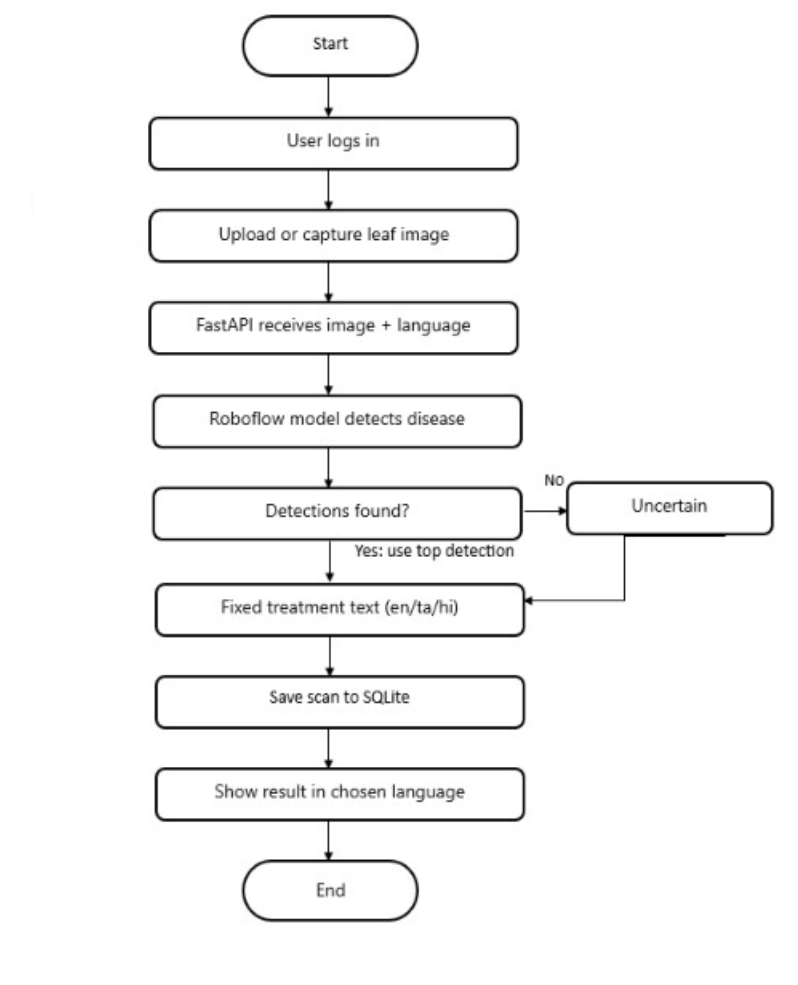
- Processor: Intel i3 or above
- RAM: Minimum 4 GB (8 GB recommended)
- Storage: Minimum 5 GB free space
- Internet connection required
- Smartphone or webcam for capturing leaf images

Software Requirements

- Frontend: Streamlit (Python) with custom CSS, HTML and SVG components
- Backend: FastAPI with Uvicorn
- Database: SQLite

- AI Detection: Roboflow hosted object detection model (coconut-tree-disease/1) through the inference SDK
- Chatbot: Llama 3.3 served through the Groq API
- Authentication: JWT with bcrypt password hashing
- Design Tool: Google Stitch
- Development Tools: Antigravity IDE, Git, Python virtual environment
- Deployment: Streamlit Community Cloud (<https://cococare.streamlit.app/>)
- Browser: Google Chrome or any modern browser

Flow Chart:



MODE OF DESCRIPTION:

- Sign up page
- User login
- Home dashboard
- Leaf image upload
- Analyzing screen
- Scan result
- Scan history

- AI chatbot
- Language selection
- Farm profile and planner

SIGN UP PAGE:

Inspect FastAPI backend server health, manage user authentication profile, and SQLite database storage.

Profile

☐ Log In ☒ Create Account

Username	Farm Name
Email	Password
Full Name	Confirm Password

Create Account

A simple form creates an account with name, email, farm name and password, with inline validation for invalid email and mismatched passwords.

USER LOGIN:

Inspect FastAPI backend server health, manage user authentication profile, and SQLite database storage.

Settings & Profile

Profile

☒ Log In ☐ Create Account

Username
Password

Log In

Users sign in with email and password. Invalid credentials show a clear error message.

HOME DASHBOARD:

Farm Profile Incomplete

Configure your farm profile in the Farm Profile & Planner tab to enable tailored irrigation, nutrient, and weather advisories.

Start Scan

Irrigation, Soil & Calendar

View Plantation Analytics

Live Farm Weather & Field Advisories — Pollachi

28.5°C

TEMPERATURE

70%

HUMIDITY

20%

RAIN CHANCE

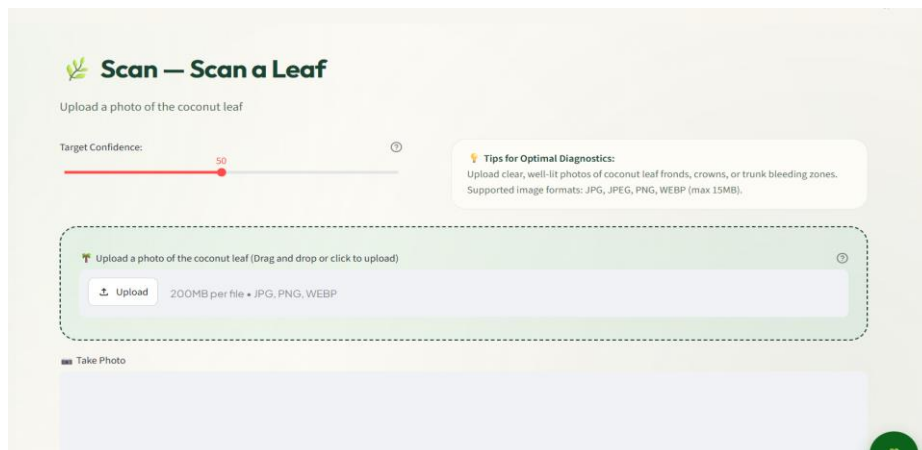
Partly Cloudy

CONDITION

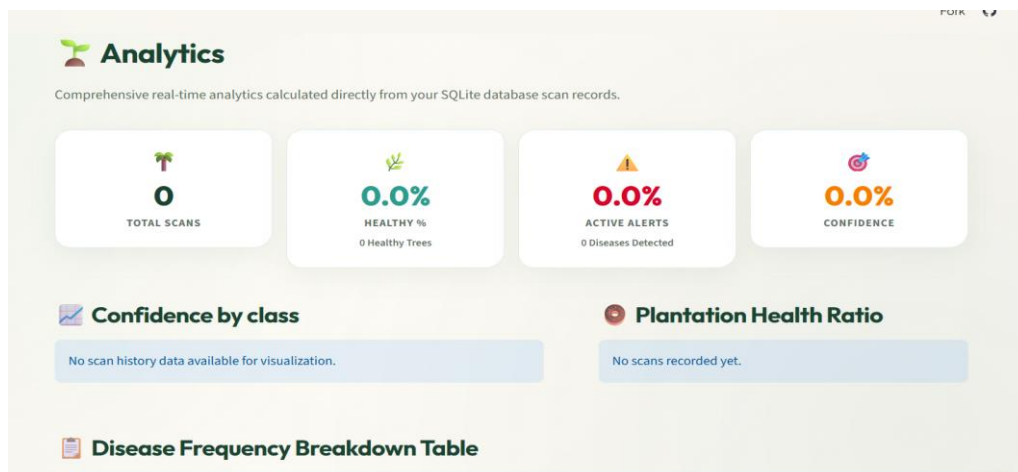
Field Advisory:

Favorable weather conditions for routine plantation operations.

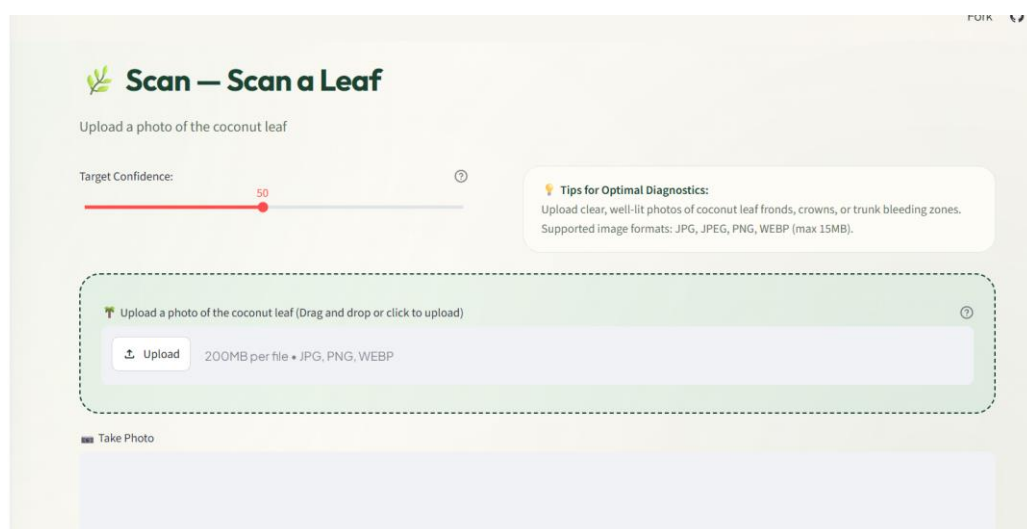
Summary cards show total scans, healthy percentage and active alerts, with a large call-to-action to scan a leaf

LEAF IMAGE UPLOAD:

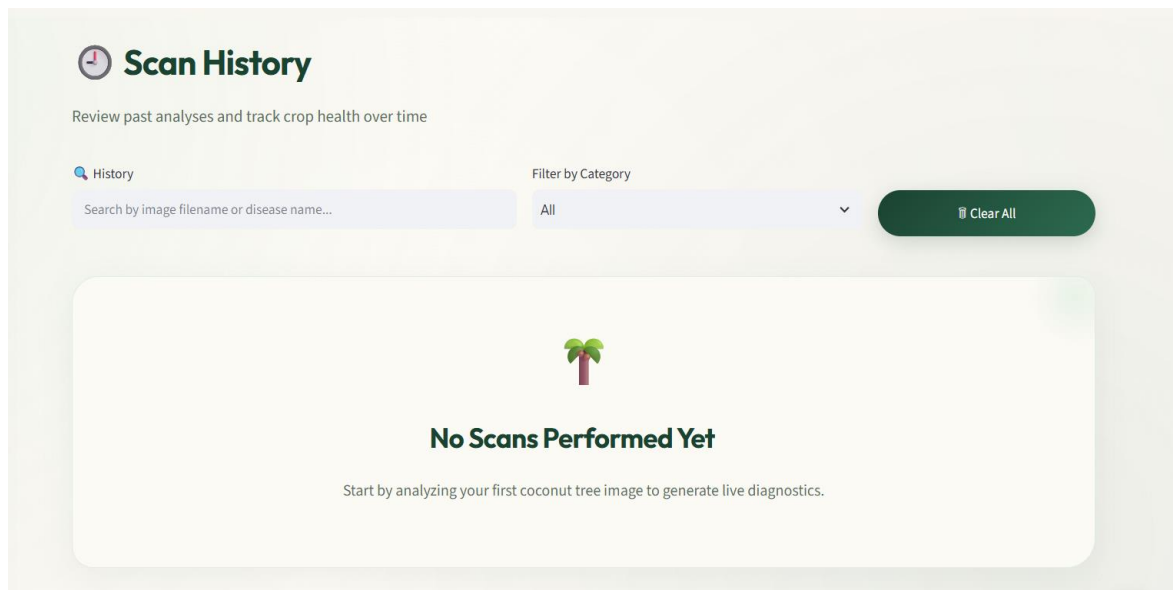
The user drags and drops an image or uses the camera.

ANALYSING SCREEN:

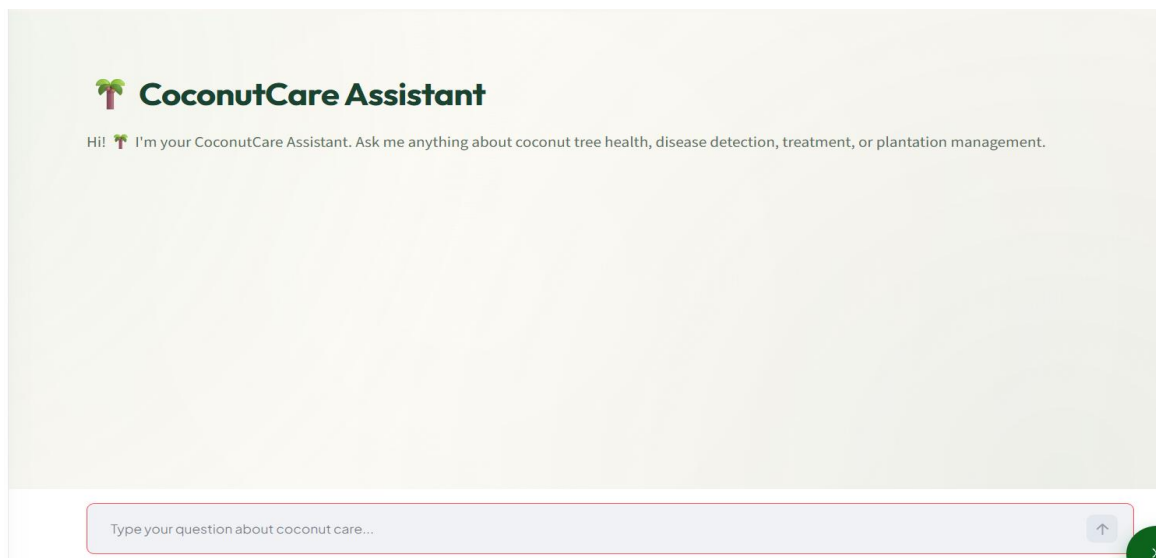
Analyze data and visualize key insights, trends, and perform

SCAN RESULT:

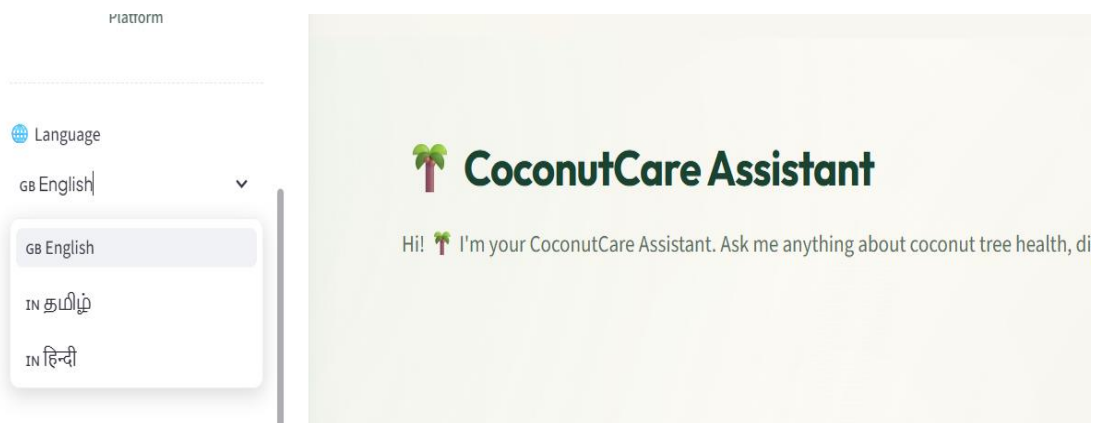
The result shows a status badge, a confidence ring, the per class breakdown and the treatment recommendation.

SCAN HISTORY:

Past scans are listed with date, status badge and confidence, and can be filtered by category.

AI CHATBOT:

A floating chat button opens a panel where the farmer asks questions and receives replies in the chosen language.

LANGUAGE SELECTION:

The language selector changes the interface and chatbot language to English, Tamil or Hindi.

FARM PROFILE AND PLANNER:

The farmer enters farm details and the planner shows an estimated tree layout.

CONCLUSION:

CocoCare AI provides a simple and accessible way for coconut farmers to check tree health from a photograph. By combining a hosted object detection model with fixed, reviewed treatment information, a multilingual chatbot and per-user history, the system gives quick guidance while avoiding the risk of invented agricultural advice. Support for English, Tamil and Hindi makes it usable for farmers who are not comfortable with English. Compared with manual inspection and general plant applications, the proposed system offers a unified, coconut focused and user-friendly platform. Overall, it improves the speed and accessibility of early disease identification and supports better farm record keeping.

REFERENCES

- Coconut Disease Prediction System Using Image Processing and Deep Learning Techniques. IEEE Conference Publication, IEEE Xplore.
- DeepSeqCoco: A Robust Mobile Friendly Deep Learning Model for Detection of Diseases in Cocos nucifera. arXiv:2505.10030.
- FastAPI Documentation. <https://fastapi.tiangolo.com>
- Groq API Documentation. <https://console.groq.com/docs>
- Mayhoc. Coconut Tree Disease Object Detection Model. Roboflow Universe. <https://universe.roboflow.com/mayhoc/coconut-tree-disease>
- Mohanty, S. P., Hughes, D. P., & Salathé, M. (2016). Using Deep Learning for Image-Based Plant Disease Detection. *Frontiers in Plant Science*.
- Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You Only Look Once: Unified, Real-Time Object Detection. *IEEE CVPR*.
- Selvaraju, R. R., et al. (2017). Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization. *IEEE ICCV*.
- Streamlit Documentation. <https://docs.streamlit.io>