Smart Plant Monitoring and Intruder Alert System using IoT

Overview

This project combines IoT-based plant monitoring with a basic home security feature using facial recognition. It ensures your plant receives the right conditions for growth while also detecting intruders using face detection. The system uses ESP8266 for sensor control and data logging, Blynk IoT Cloud for live monitoring, and Python + OpenCV for webcam-based intruder alert.

Features

- Real-time soil moisture, temperature, and humidity monitoring.
- Automatic watering via relay when soil is dry.
- Local LED indicators to reflect soil status.
- Live data dashboard using Blynk Cloud.
- Intruder detection via laptop webcam using OpenCV.
- Instant buzzer alert and notification via Blynk if face is detected.
- Live camera streaming via a Flask web app.

Components Used

Component	Quantity	Description
ESP8266 NodeMCU	1	Microcontroller with Wi-Fi support
Soil Moisture Sensor	1	Analog moisture detection
DHT11/DHT22	1	Temperature and humidity sensor
Relay Module	1	To control water pump
LED	2	For dry and sufficient moisture status
Buzzer	1	Sounds alert on intruder detection
Laptop Webcam	1	Used for live streaming and detection
Flask + OpenCV (Python)	Software	Used for video stream and face detection
Blynk IoT Cloud	Platform	IoT dashboard for control and monitoring

---

Wiring Diagram

- Soil Moisture: A0 of NodeMCU
- DHT11: D5 (GPIO14)
- Relay: D6 (GPIO12)
- LED1 (Dry): D1
- LED2 (Sufficient Moisture): D2
- Buzzer: D7 (GPIO13)

Blynk Setup

Virtual Pins:

Virtual Pin	Function	Units
V0	Temperature	°C
V1	Humidity	%
V2	Soil Moisture	0–1000 (analog)
V3	Intruder Alert Flag	0 or 1

Arduino Code (ESP8266)

- Reads DHT11 and Soil sensor
- Sends data to Blynk virtual pins
- Controls relay based on soil level
- Lights LEDs depending on moisture status

Thresholds

- Moisture < 400: Soil is dry -> Relay ON, LED1 ON
- Moisture > 700: Soil is okay -> Relay OFF, LED2 ON

Python Code

File 1: `live_stream.py`

- Flask-based live stream of webcam on `localhost:5000`

File 2: `intruder_detection.py`

- Uses OpenCV to detect faces

- Sends alert to Blynk via HTTP API when face is detected
- Also stops alert when face is gone

Python Libraries

Include the following in `requirements.txt`:

```
flask
opencv-python
requests
```

Install using:

```
pip install -r requirements.txt
```

Folder Structure

```
SmartPlantIntruderProject/
├── esp8266_plant_monitor.ino
├── esp8266_buzzer_control.ino
├── live_stream.py
├── intruder_detection.py
├── README.md
├── requirements.txt
├── wiring_diagram.png
├── blynk_dashboard.png
```

How to Run

Step 1: Setup ESP8266

- Upload `esp8266_plant_monitor.ino` to NodeMCU
- Ensure correct Wi-Fi and Blynk token

Step 2: Run Python Scripts

- Start webcam stream:

```
python live_stream.py
```

- In another terminal, start face detection:

```
python intruder_detection.py
```

Step 3: Open Blynk App

- View temperature, humidity, moisture
- Check alert status on V3
- Remotely monitor system

Future Improvements

- Add image capture and logging for intruder faces
- Use ESP32-CAM to eliminate need for PC webcam
- Add water level monitoring
- Enable notifications via email/WhatsApp

Credits

Developed by Aaryan Gupta and Anmol Mishra as a full-stack smart IoT project combining plant health monitoring with basic home security.

License

This project is open-source and free to use for educational and non-commercial purposes.