System Monitor with Telegram Alerts

A Streamlit Application for Monitoring System Status and Sending Alerts via Telegram

Introduction

- Overview of the system monitor script
- Purpose: To monitor system status and send alerts for anomalies
- Integration with Telegram for real-time notifications

Configuration Management

- Load configuration from config.json
- Default settings if the configuration file is not found
- Configuration parameters: USB port, baud rate, Telegram bot token, Telegram chat ID, check interval, and anomaly threshold

Initializing Session State

- Load configuration into session state
- Initialize alert history and IsolationForest model for anomaly detection

Sending Telegram Messages

- Function to send messages via Telegram
- Use of Telegram bot API to send messages
- Error handling for failed message sends

User Interface (UI)

- Streamlit sidebar for configuration
- Input fields for USB port, baud rate, Telegram bot token, Telegram chat ID, and check interval
- Button to save configuration
- Button to test Telegram alert

Checking Device Status

- Function to check device status and read data
- Detect anomalies in data readings
- Send alerts if anomalies are detected

Detecting Anomalies

- Function to detect anomalies using IsolationForest model
- Handling value errors for invalid data

Alert History

- Display alert history in the Streamlit app
- Store and display the last 10 alerts

Main Monitoring Loop

○ **Continuously check device status at specified intervals**
○ **Use st.experimental_rerun to refresh the app**

Conclusion

- Summary of the system monitor script
- Benefits of real-time monitoring and alerting
- Future improvements and potential features