🛢️ Oil Rig Raspberry Pi Control System – Quick Start Guide

---

1. Project Overview

A Raspberry Pi-based dashboard for **monitoring and controlling oil rig sensors** and systems via GPIO and WiFi. Built with **Flask** and **Tailwind CSS**, it provides:

- Real-time sensor readings (Pressure, Temperature, Gas)
- Control of critical equipment (BOP, Mud Pump, Emergency Alarm)
- Active systems display with color-coded badges
- Operator-friendly guidance

---

2. Hardware Setup

Required:

- Raspberry Pi 4 (or Pi with GPIO)
- Pressure, Temperature, Gas sensors
- Relay modules for: BOP, Mud Pump, Emergency Alarm
- WiFi connectivity

Connections:

Device	GPIO Pin
Pressure Sensor	17
Temperature Sensor	27
Gas Sensor	22
BOP Relay	configurable
Mud Pump Relay	configurable
Alarm Relay	configurable

⚠️ Always power off Pi when wiring hardware.

3. Software Setup

1. Install Python 3.9+ and Flask

```
sudo apt update
sudo apt install python3 python3-pip
pip3 install flask
```

2. Clone the repository

```
git clone https://github.com/yourusername/oilrig-pi-dashboard.git
cd oilrig-pi-dashboard
```

3. Run Flask app

```
export FLASK_APP=app.py
export FLASK_ENV=development
flask run --host=0.0.0.0
```

4. Access Dashboard:

```
http://<raspberry_pi_ip>:5000/
```

4. Dashboard Overview

Sensor Readings

Displays live values for:

- Well Pressure
- Drill Temperature
- Gas Level (Safe / Danger)

Control Systems

- BOP, Mud Pump, Emergency Alarm
- Toggle buttons with confirmation prompts
- Alarm button dynamically changes Activate / Deactivate

Active Systems

- Shows which devices are currently ON
- Color-coded to match the buttons:
    - BOP → Red
    - Mud Pump → Yellow
    - Alarm → Orange

Status Indicators

- Green dot = Online / Connected
- Red dot = Offline / Disconnected

5. Using Controls

1. Click toggle button
2. Confirm action in dialog box
3. Button updates automatically if device is active
4. Active Systems section refreshes every 5 seconds

⚠️ Emergency Alarm should only be used in real emergencies.

6. Flask API Endpoints

Endpoint	Method	Description
/api/toggle/<device>	POST	Toggle device (bop, pump, alarm)
/api/toggle-states	GET	Return current ON/OFF states
/api/status	GET	Return WiFi and server status

7. Safety Guidelines

- Always confirm before toggling equipment
- Follow operational protocols for BOP and Mud Pump
- Use simulation mode when testing

8. Contact

- **Support**: techsupport@pason.com
- **Emergency Hotline**: +1 (877) 255-3158

9. Notes

- System supports hybrid operation (simulated vs live GPIO)
- Dashboard is responsive for tablet or desktop use
- Active Systems section and Alarm button provide visual feedback for critical systems