IOT : PATIENT MONITORING SYSTEM

Shashwat Singh (B20230)
Prakash Mandloi (B20222)

OBJECTIVE:

The objective of this project is to develop a patient monitoring system for an ICU with five beds. Each bed is equipped with a sensor suite that measures vital health parameters such as heart rate (BPM), blood pressure (BP), and blood oxygen saturation (spO2). The system aims to monitor and summarize the information of each patient in real-time, displaying it at a nurse's station. Additionally, in case of emergencies, the system must provide a warning for the correct patient. The project considers two test cases: ICU during a pandemic and ICU during a normal period.

COMPONENT REQUIRED:

a) 5 NodeMCU Boards: These boards are used to collect sensor data from the beds and transmit it to the Raspberry Pi where the NodeRED server is running.

b) Raspberry Pi (Rpi): The Raspberry Pi acts as the central server for data collection of all the patients and after that is sent to NodeRED.

SOLUTION DETAIL:

a) NodeMCU Boards: Each NodeMCU board is connected to a specific bed's sensor suite. It collects sensor data periodically and sends it to the Raspberry Pi using the MQTT protocol. The data includes BPM, BP, and spO2 readings for each patient. As we are not using real sensors, we are generating sensor data using multivariate normal distributions based on mean values specific to the test case (normal or COVID).

b) Raspberry Pi (Rpi): The Raspberry Pi acts as the central server. It runs the Node-RED platform, which receives the sensor data from the NodeMCU boards. The Rpi processes the incoming data, analyzes it, and triggers appropriate actions based on the specified thresholds for each parameter. It maintains a database of patient information and updates it in real-time.

c) Node-RED: Node-RED provides a graphical interface for creating flows and handling the received sensor data. It receives MQTT messages from the NodeMCU boards and extracts the BPM, BP, and spO2 values for each patient. The data is then visualized using gauges for each parameter.

d) Thresholds and Emergency Warnings: The system compares the received sensor data with predefined threshold values for BPM and BP. If any of the parameters exceed the threshold, an emergency warning is

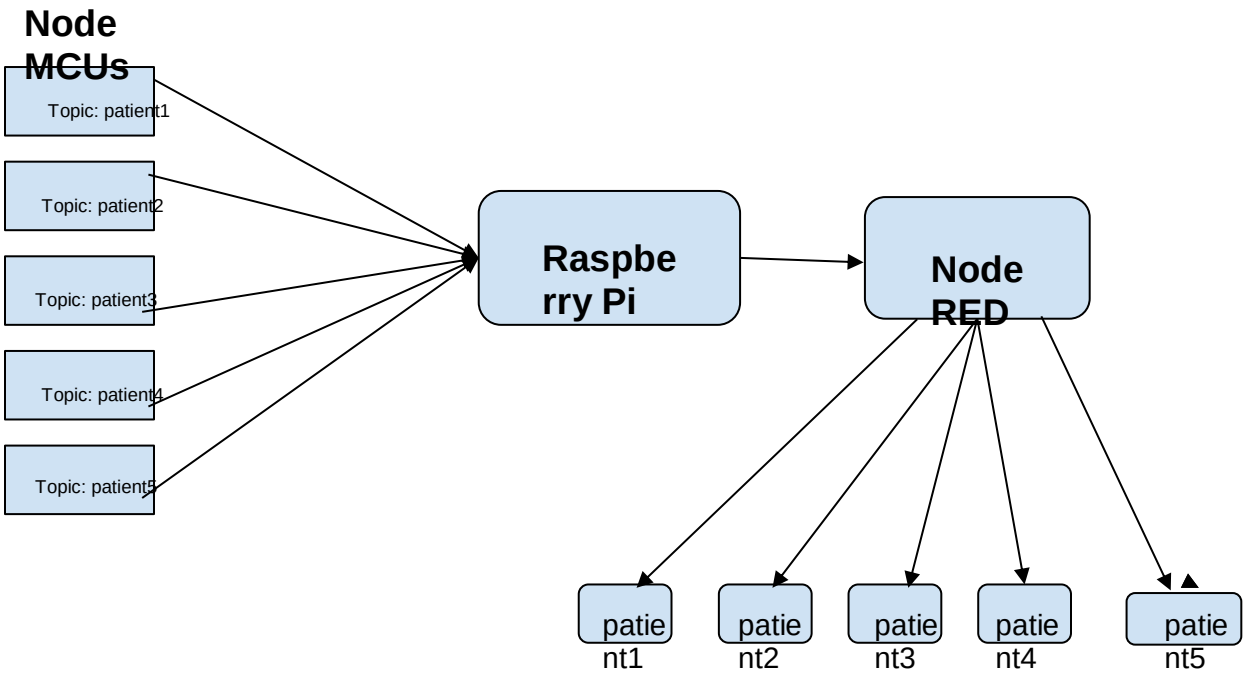
triggered in Node-RED for the corresponding patient, alerting the nurses at the station.

CONTROL FLOW:

1. Initialization:
    - Set up the NodeMCU boards, Raspberry Pi, and Node-RED.
    - Configure MQTT communication between the NodeMCU boards and the Raspberry Pi.
2. Data Collection:
    - The NodeMCU boards collect sensor data (BPM, BP, and spO2) from each bed's sensor suite.
    - Generate sensor data using multivariate normal distributions based on the mean values, variance & standard deviation specific to the test case.
3. Data Transmission:
    - The NodeMCU boards send the sensor data to the Raspberry Pi using the MQTT protocol.
    - Each patient's data is published to a specific MQTT topic. For example, the topic for the first patient is "patient1," the topic for the second patient can be "patient2," and so on.
    - We are sent data in the form of a string containing three data BPM, BP & spO2 values with comma separated.
4. Data Processing and Visualization:
    - The Raspberry Pi receives the MQTT messages and passes the data to Node-RED.
    - Node-RED subscribes to the MQTT topics for each patient and receives the sensor data.
    - Node-RED processes the data and extracts the vital health parameters (BPM, BP, and spO2) for each patient.
    - The extracted data is visualized using gauges on the Node-RED dashboard.
5. Threshold Comparison and Emergency Alert:
    - For deciding Emergency cases, the JavaScript code compares the received data for each patient with pre-defined threshold values for BPM and BP.
    - If any of the parameters exceed the thresholds, an emergency warning is triggered in Node-RED for the corresponding patient.
    - The emergency warning can be displayed in the summary section of the Node-RED dashboard or sent as a notification to the nurses' station.
6. Nurse's Station Monitoring:
    - Nurses at the station can access the Node-RED dashboard to monitor the patient's information.
    - The dashboard displays the vital health parameters (BPM, BP, and spO2) for each patient in real-time.
    - In case of an emergency, the corresponding patient's information is highlighted or a separate emergency alert is displayed.
7. Continuous Monitoring:
    - The system continues to collect sensor data from the NodeMCU boards at regular intervals. i.e after every 10 seconds data is sended.
    - The data is transmitted to the Raspberry Pi via MQTT and processed in Node-RED for visualization and threshold comparison.

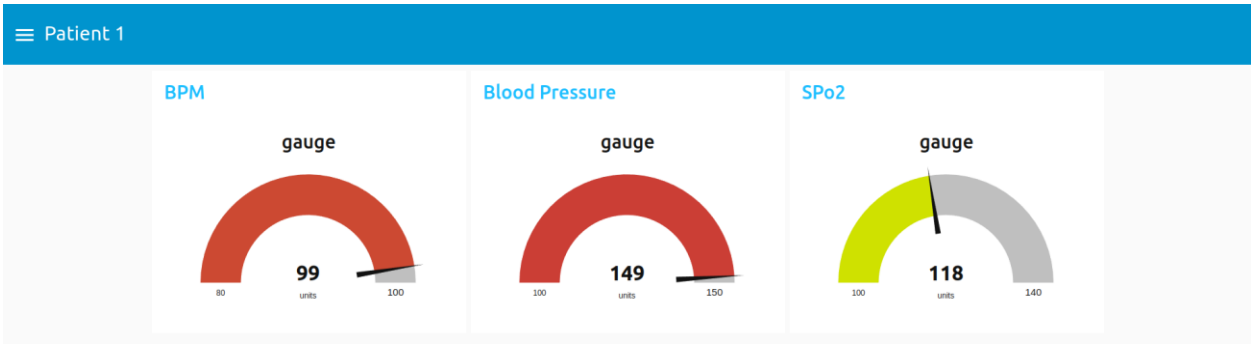
- The system provides continuous monitoring of the patients' vital health parameters and alerts the nurses in case of emergencies.

By utilizing the MQTT protocol and appropriate topic names for each patient, the system can differentiate and process the sensor data from multiple beds in the ICU. This allows for efficient monitoring and accurate emergency alerts for individual patients.

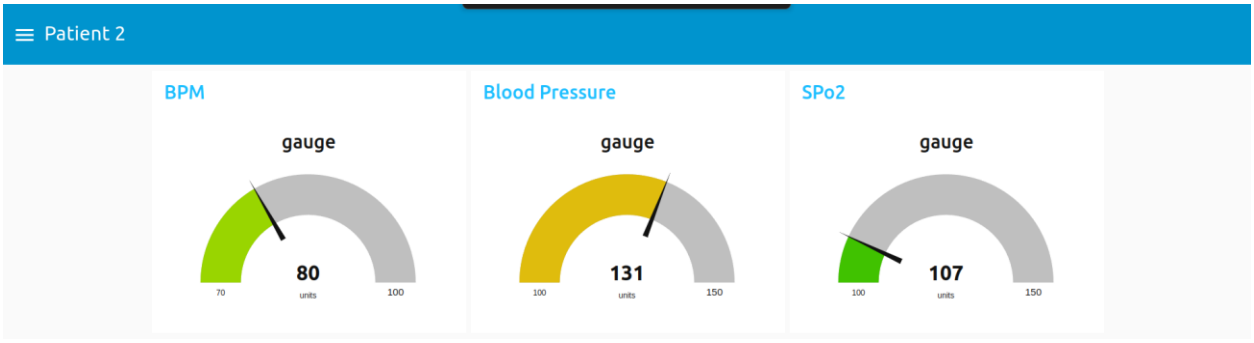


OUTPUTS:

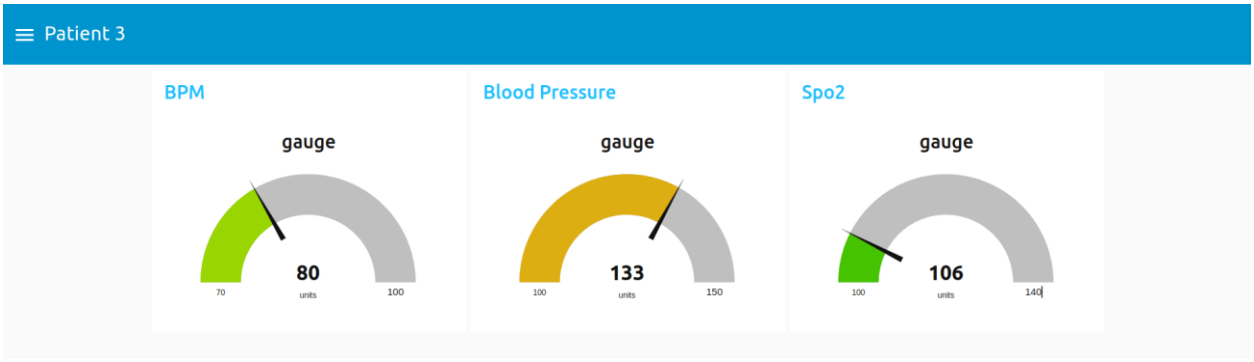
Patient 1



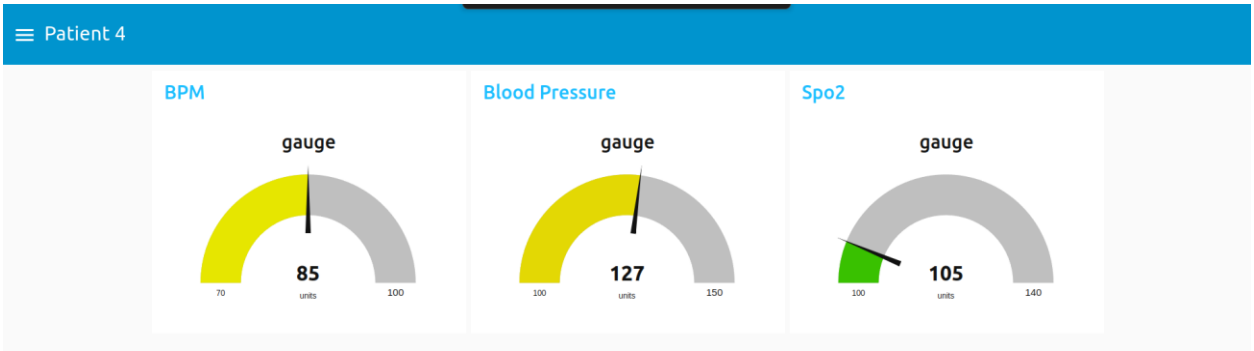
Patient 2



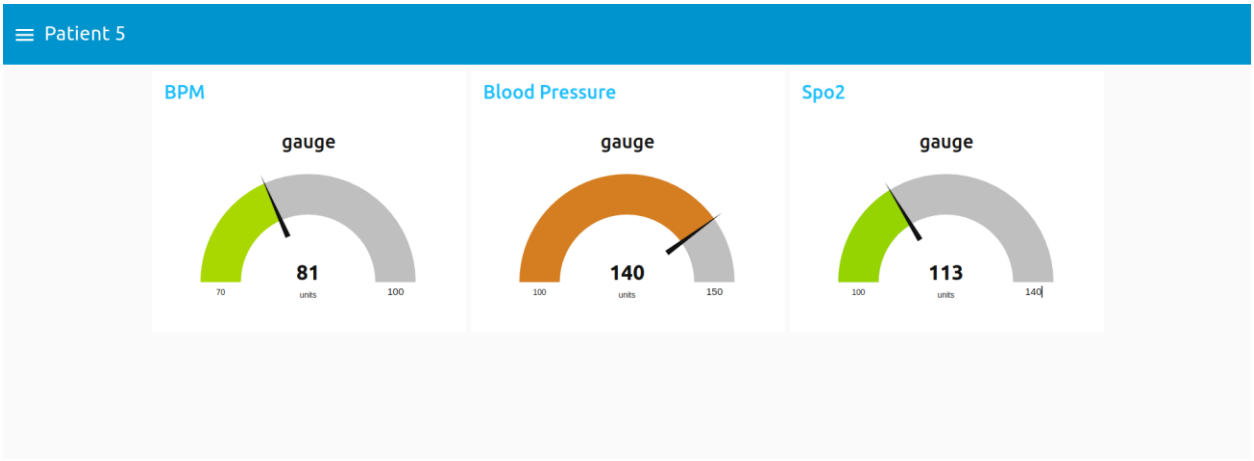
Patient 3



Patient 4



Patient 5


Patient 5
BPM
gauge
81
units
70
100
Blood Pressure
gauge
140
units
100
150
Spo2
gauge
113
units
100
140

Summary


Summary
Patient 5
text
Emergency
Patient 4
text
Emergency
Patient 3
text
Normal
Patient 2
text
Normal
Patient 1
text
Emergency

END