Jawahar Education Societys Annasaheb Chudaman Patil College of Engineering, Kharghar, Navi Mumbai

EXPERMINT: 03

● **Title**: IOT NodeRed simulator

● **Aim:** Real time data acquisition and transmission using NodeRed simulator.

●Theory:

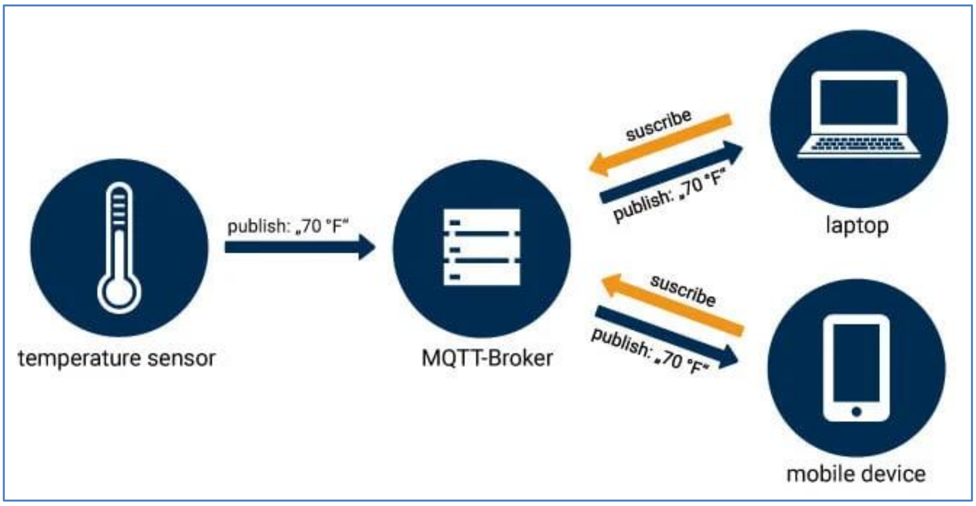
●NodeRed:

Node-RED is a flow-based, low-code development tool for visual programming developed originally by IBM for wiring together hardware devices, APIs and online services as part of the Internet of Things.

Node-RED provides a web browser-based flow editor, which can be used to create JavaScript functions. Elements of applications can be saved or shared for re-use. The runtime is built on Node.js. The flows created in Node-RED are stored using JSON. Since version 0.14, MQTT nodes can make properly configured TLS connections.

●MQTT stands for Message Queuing Telemetry Transport.

MQTT stands for Message Queuing Telemetry Transport. It is an extremely simple and lightweight messaging protocol (subscribe and publish) designed for limited devices and networks with high latency, low bandwidth or unreliable networks. Its design principles are designed to reduce the network bandwidth and resource requirements of devices and ensure security of supply. In addition, these principles are advantageous for M2M (machine-to-machine) or IoT devices because battery performance and bandwidth are very important.



The MQTT broker is the center of every Publish / Subscribe protocol. Depending on the implementation, a broker can manage up to thousands of simultaneously connected MQTT clients. The broker is responsible for receiving all messages, filtering the messages, determining who subscribed to each message and sending the message to those subscribed clients. The Broker also holds the sessions of all persistent clients, including subscriptions and missed messages. Another task of the Broker is the authentication and authorization of clients. Usually the broker is extensible, which facilitates custom authentication, authorization and integration with backend systems. Integration is especially important, because the Broker is often the component directly exposed on the Internet, serves many clients and has to forward messages to downstream analysis and processing systems.

●Features of MQTT

There are various features of MQTT which are as follows:

- It is a machine-to-machine protocol, i.e., it provides communication among the devices.
- It is produced as a simple and lightweight messaging protocol that supports a publish/subscribe system to transfer the data between the user and the server.
- It does not require that both the user and the server create a connection at an equal time.
- It provides quicker data transmission, like how Whatsapp/messenger supports a quicker delivery.
- It enables the users to subscribe to the definite selection of topics to receive the data they are viewing for.
- It can distribute data more effectively.
- It can increase scalability.
- It is used to lower network bandwidth consumption dramatically

●Advantages Of MQTT Protocol

Following are the benefits or advantages of MQTT protocol:

- The MQTT protocol payload can carry any type of data such as binary, ascii text etc. The receiver need to interpret and decode as per format used by the transmitter. Hence MQTT is packet agnostic.
- It uses packet of low size and hence can be used for low bandwidth applications.
- It offers lower battery power consumption.
- It is reliable protocol as it uses QoS options to provide guaranteed delivery.
- Due to its publish/subscribe model, It is scalable.
- It offers de-coupled design as it is easy to decouple the device and server.

●*Here's a general guide on how to simulate data or interact with simulated devices using Node-RED:*

1. Install Node-RED:

- Ensure Node-RED is installed on your system. You can install it globally using npm:

```
bashCopy code

npm install -g node-red
```

2. Start Node-RED:

- Start Node-RED by running the following command:

```
bashCopy code

node-red
```

- Access the Node-RED editor by opening your web browser and navigating to **http://localhost:1880**.

3. Install Necessary Nodes:

- Depending on your simulation requirements, you might need to install additional nodes. For example, for MQTT communication or other IoT protocols.

4. Create a Flow:

- Design a flow in Node-RED by dragging and dropping nodes onto the workspace.

5. Simulate Data:

- Use inject nodes or function nodes to simulate data. Inject nodes allow you to inject messages at specific intervals.

6. Connect to External Simulators:

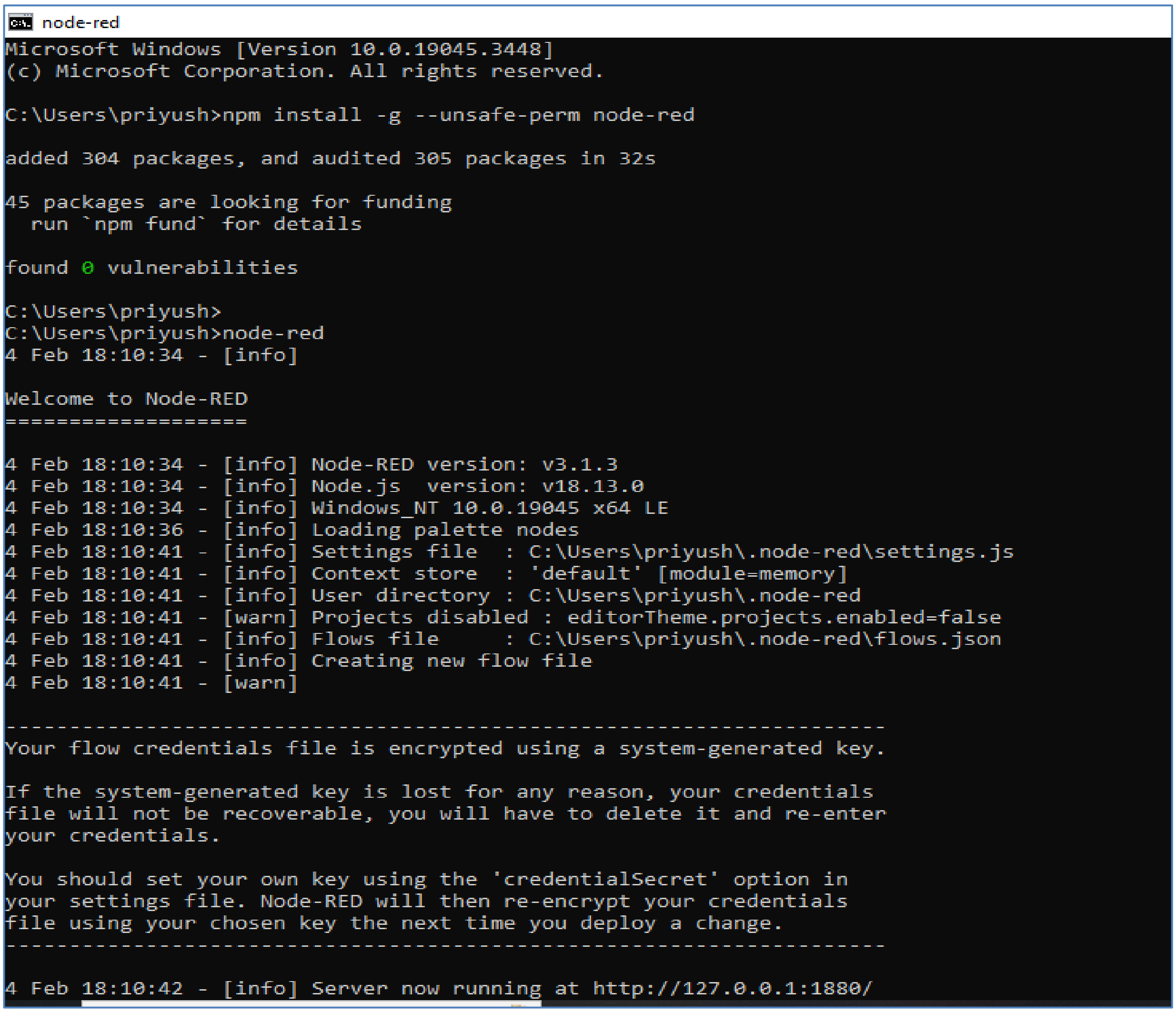
- If you're working with external simulators, connect Node-RED to them using appropriate nodes. For instance, MQTT nodes can connect to an MQTT broker that is part of your simulation.

7. Visualize Data (Optional):

- Use Dashboard nodes to create visualizations and dashboards for the simulated data.

8. Deploy and Monitor:

- Deploy your flow, and monitor the Node-RED debug console for any messages or errors.



```
Microsoft Windows [Version 10.0.19045.3448]
(c) Microsoft Corporation. All rights reserved.

C:\Users\priyush>npm install -g --unsafe-perm node-red

added 304 packages, and audited 305 packages in 32s

45 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

C:\Users\priyush>
C:\Users\priyush>node-red
4 Feb 18:10:34 - [info]

Welcome to Node-RED
===================

4 Feb 18:10:34 - [info] Node-RED version: v3.1.3
4 Feb 18:10:34 - [info] Node.js  version: v18.13.0
4 Feb 18:10:34 - [info] Windows_NT 10.0.19045 x64 LE
4 Feb 18:10:36 - [info] Loading palette nodes
4 Feb 18:10:41 - [info] Settings file  : C:\Users\priyush\.node-red\settings.js
4 Feb 18:10:41 - [info] Context store  : 'default' [module=memory]
4 Feb 18:10:41 - [info] User directory : C:\Users\priyush\.node-red
4 Feb 18:10:41 - [warn] Projects disabled : editorTheme.projects.enabled=false
4 Feb 18:10:41 - [info] Flows file     : C:\Users\priyush\.node-red\flows.json
4 Feb 18:10:41 - [info] Creating new flow file
4 Feb 18:10:41 - [warn]

---------------------------------------------------------------------
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
---------------------------------------------------------------------

4 Feb 18:10:42 - [info] Server now running at http://127.0.0.1:1880/
```

●**Step Real time data acquisition and transmission using NodeRed simulator.**

1. **Install Necessary Nodes:**

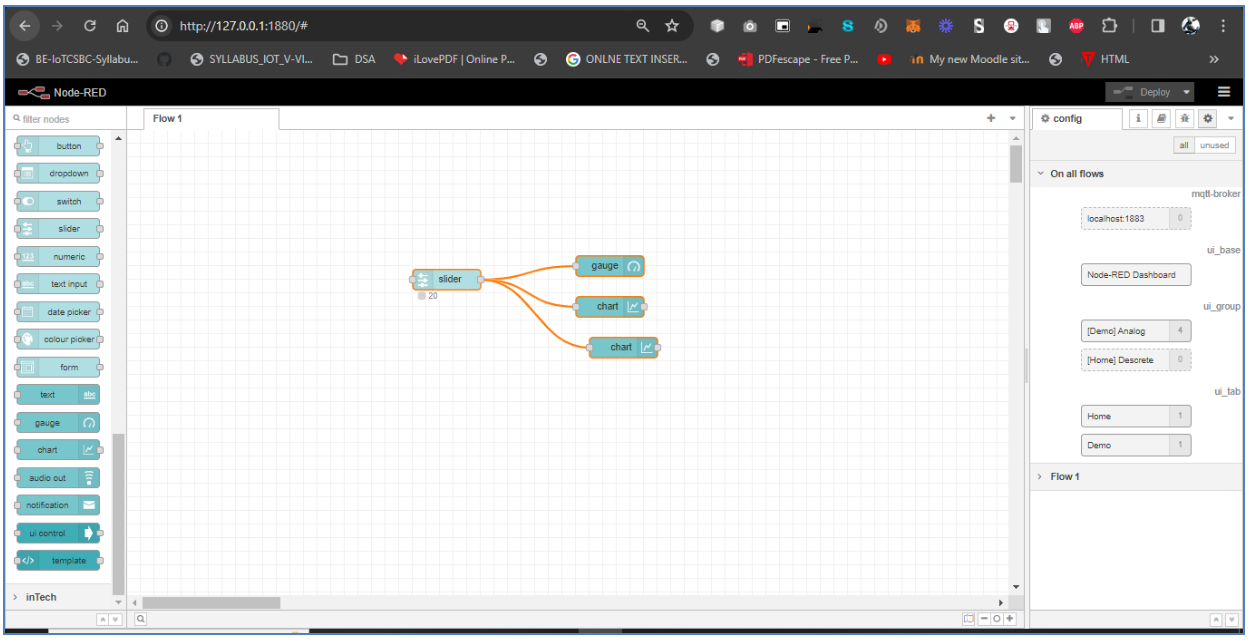
- In Node-RED, install nodes that are necessary for your project, such as MQTT nodes for communication.

2. **Create Data Source:**

- Simulate or connect a data source. This could be a simulated sensor generating data or an actual IoT device.

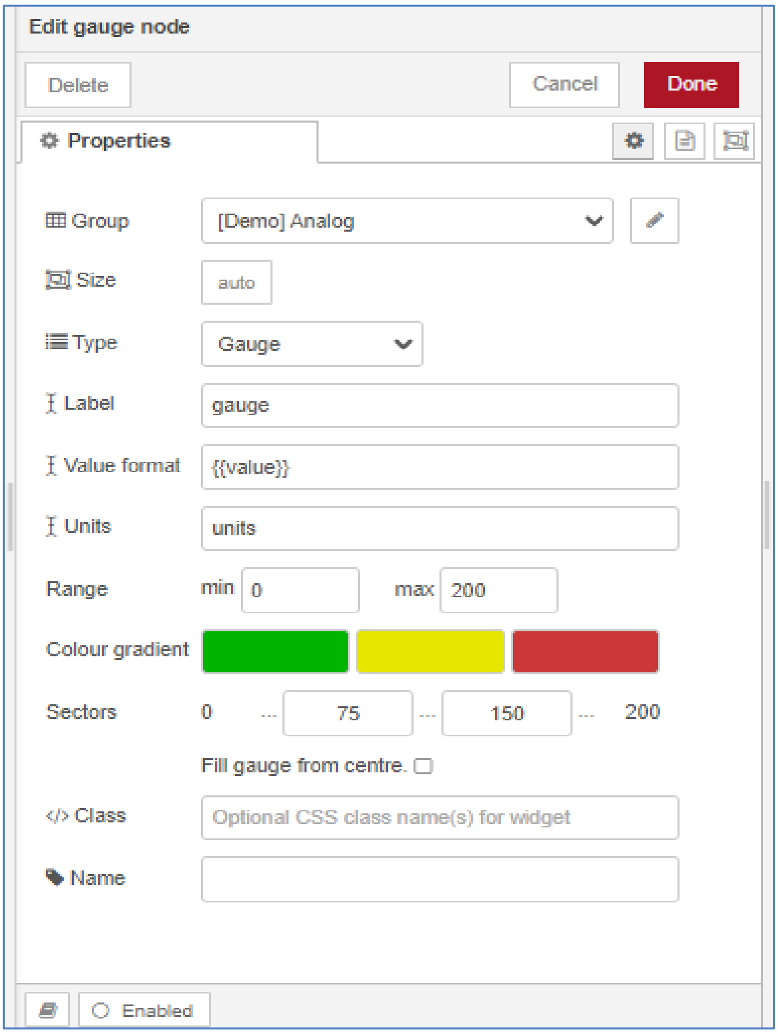
3. **Node-RED Flow Design:**

- Design a flow in Node-RED that includes nodes for data acquisition, processing, and transmission.



4. **Data Acquisition:**

- Use appropriate nodes (e.g., inject nodes, function nodes) to acquire real-time data from the source.



Group: [Demo] Analog

Size: auto

Type: Gauge

Label: gauge

Value format: {{value}}

Units: units

Range: min 0 max 200

Colour gradient: 🟩 🟨 🟥

Sectors: 0 ... 75 ... 150 ... 200

Fill gauge from centre. □

Class: Optional CSS class name(s) for widget

Name:

5. **Data Processing:**

- Implement any required data processing within Node-RED. This could include filtering, formatting, or aggregating data.

6. **Real-time Transmission:**

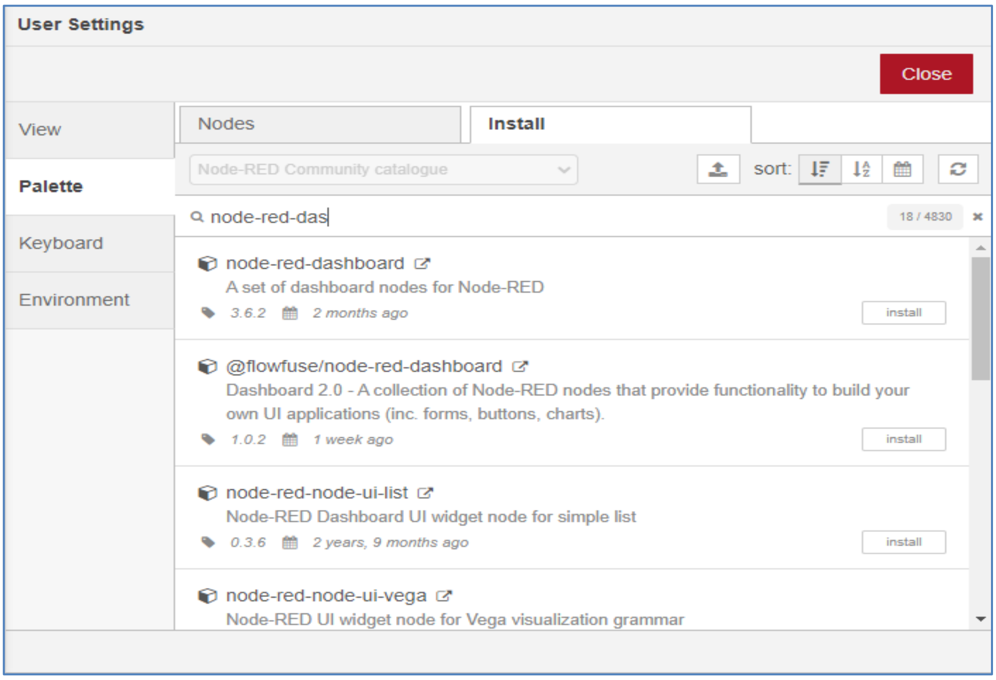
- Use nodes like MQTT or WebSocket nodes to transmit real-time data to a destination.

7. **Integration with External Services:**

- If necessary, integrate Node-RED with external services or databases to store or further process the data.

8. **Dashboard (Optional):**

- Create a real-time dashboard using Node-RED Dashboard nodes to visualize the acquired data.



User Settings

Close

View

Palette

Keyboard

Environment

Nodes | Install

Node-RED Community catalogue

sort:

node-red-das

18 / 4830

node-red-dashboard
A set of dashboard nodes for Node-RED
3.6.2 2 months ago
install

@flowfuse/node-red-dashboard
Dashboard 2.0 - A collection of Node-RED nodes that provide functionality to build your own UI applications (inc. forms, buttons, charts).
1.0.2 1 week ago
install

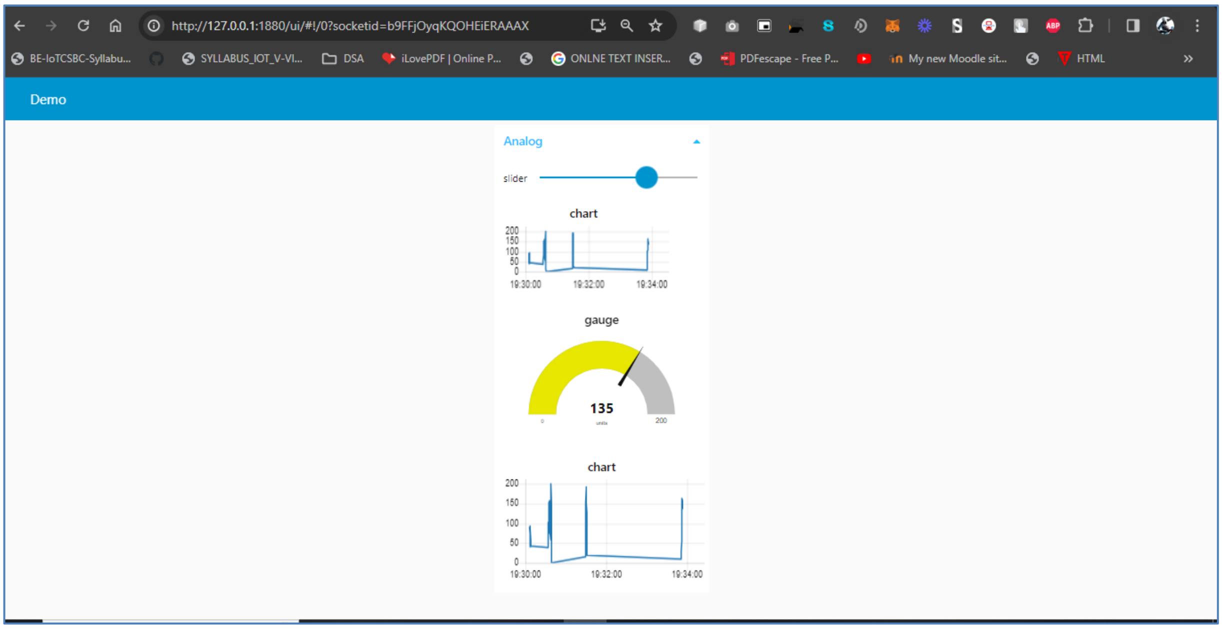
node-red-node-ui-list
Node-RED Dashboard UI widget node for simple list
0.3.6 2 years, 9 months ago
install

node-red-node-ui-vega
Node-RED UI widget node for Vega visualization grammar

9. **Testing:**

- Test the entire flow by deploying it in Node-RED and observing the real-time data acquisition and transmission.

●**Result**:



●**Conclusion:**

Implementing real-time data acquisition and transmission using Node-RED provides a flexible and visual approach to building IoT applications. It allows you to quickly prototype and iterate on your project, making it easier to adapt to changing requirements.