Sprint - 2

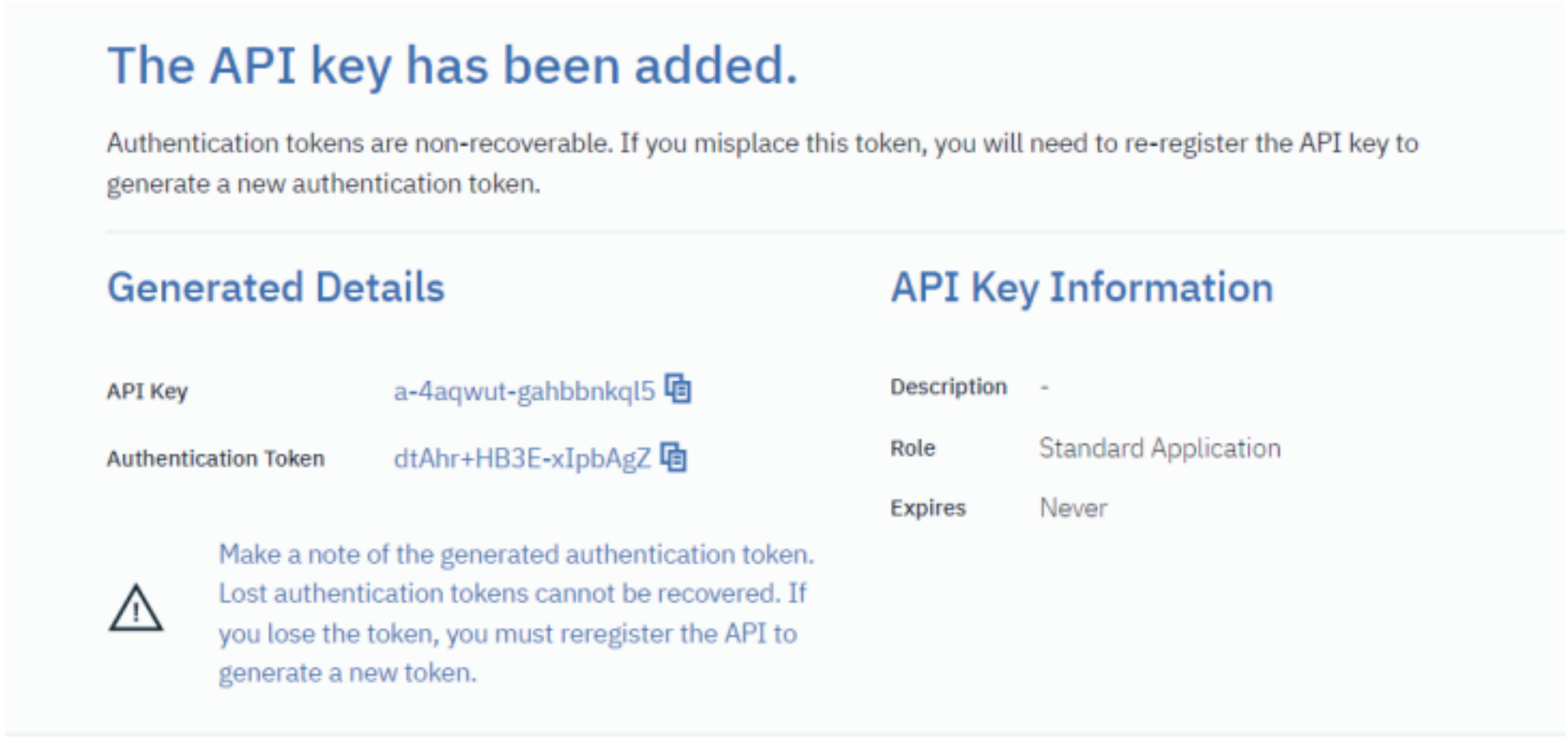
Date	04 November 2022
Team ID	PNT2022TMID50096
Project Name	Project - Industry-specific intelligent fire management system
Maximum Marks	20 marks

Sprint-2	US-1	Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.	10	High	Sneha K, Thanalakshmi M
Sprint-2	US-2	Create a Node-RED service.	10	High	Ramya R,Santhiya V

US-1 Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.

US-2 Create a Node-RED service.

US-1 Configure the connection security and create API keys that are used in the Node-RED service for accessing the IBM IoT Platform.



US-2 Create a Node-RED service

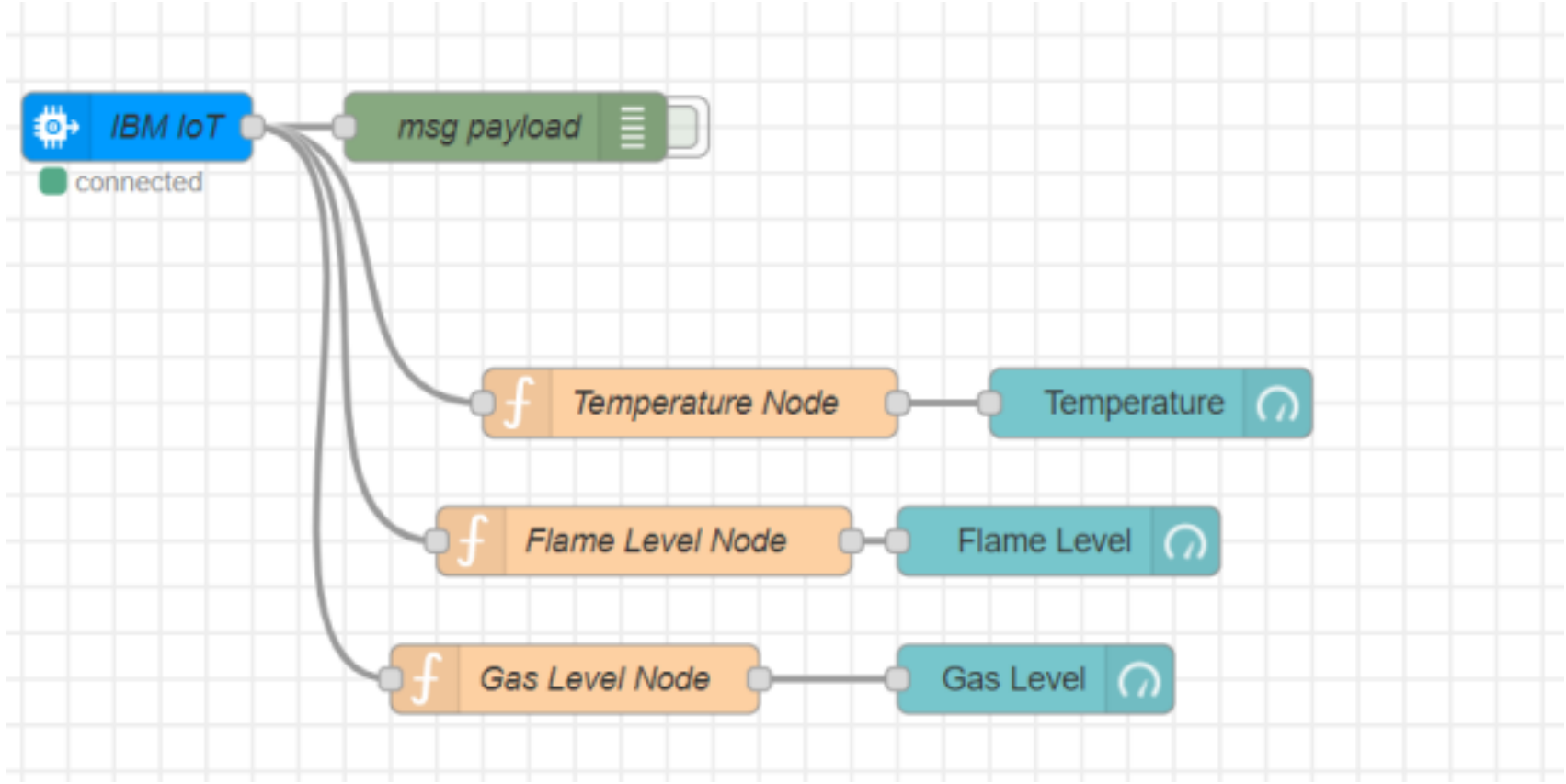


fig1 - Monitoring the sensor values - Temperature, Flame Level, Gas Level. These values are randomly generated by IBM WATSON IOT PLATFORM.

Fig 2 - Temperature, Flame_Level, Gas_Level values displayed in deploy tab in node-red

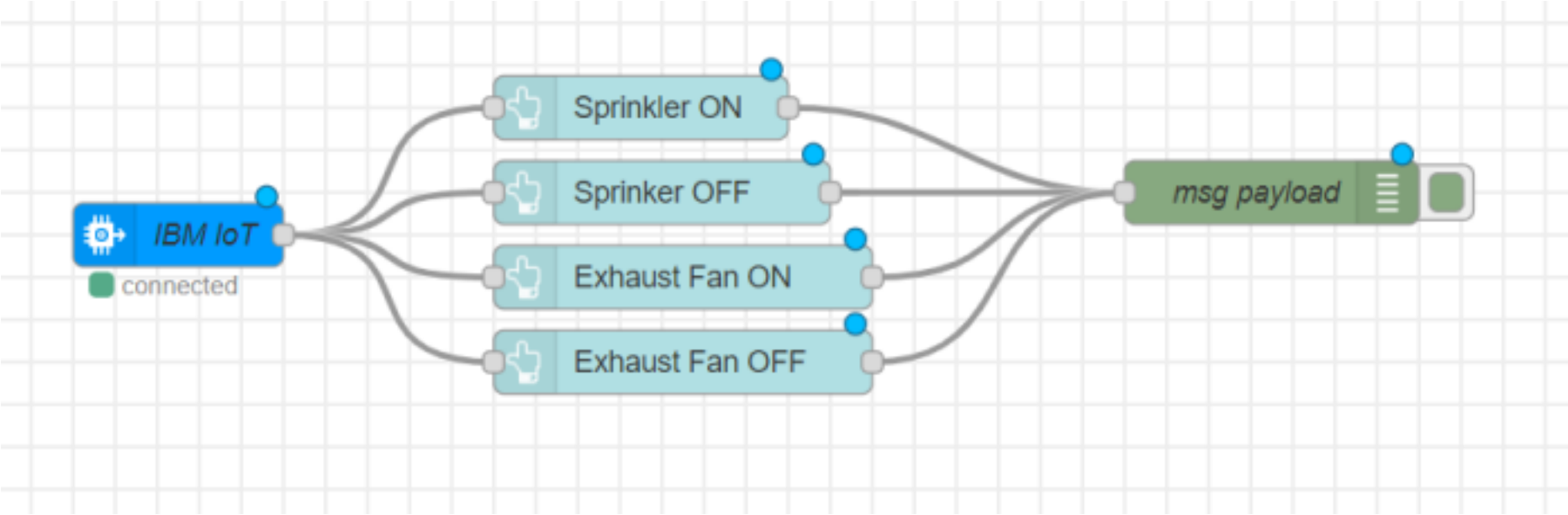


fig 3 - Control buttons (Sprinkler ON, Sprinkler OFF, Exhaust Fan ON, Exhaust Fan OFF)

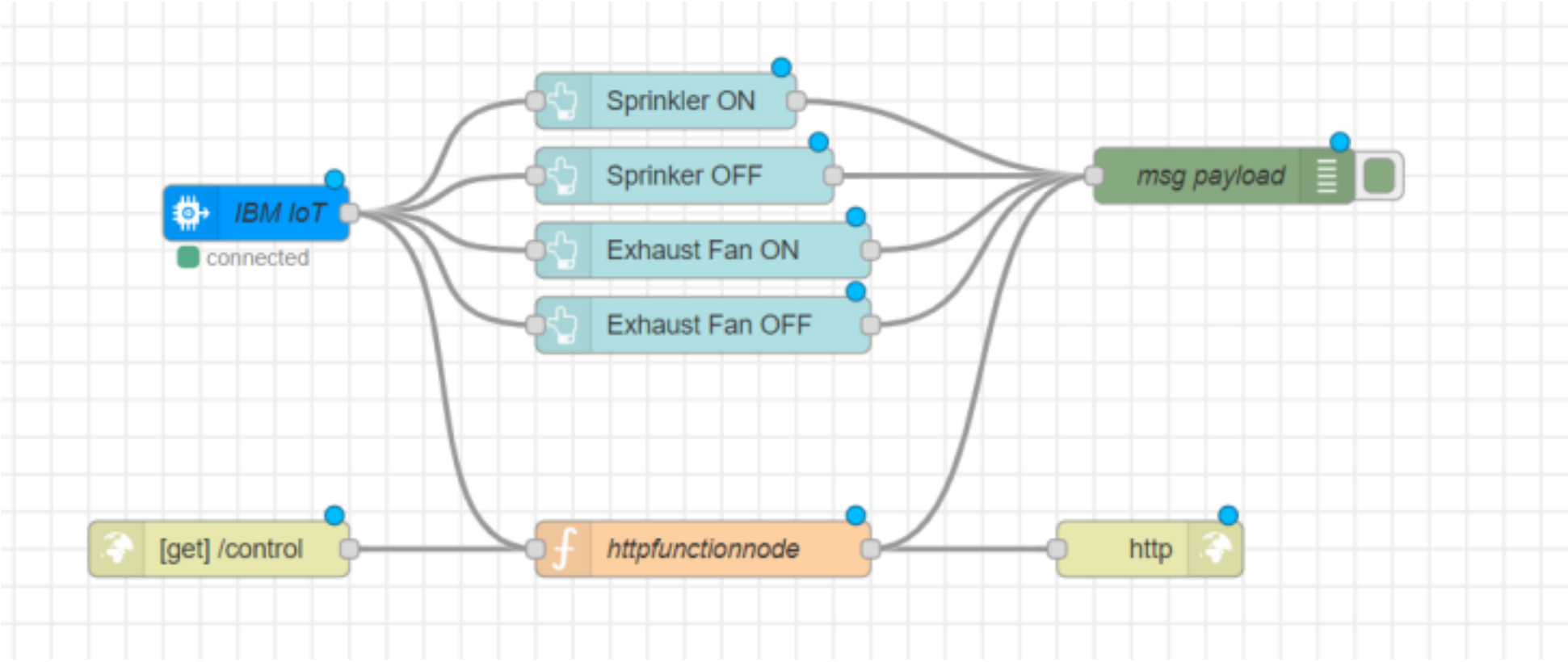


Fig 4 - Using HTTP in and HTTP response in network option, http://127.0.0.1:1880/#flow/f74f1b96473dc208/control will display the control options

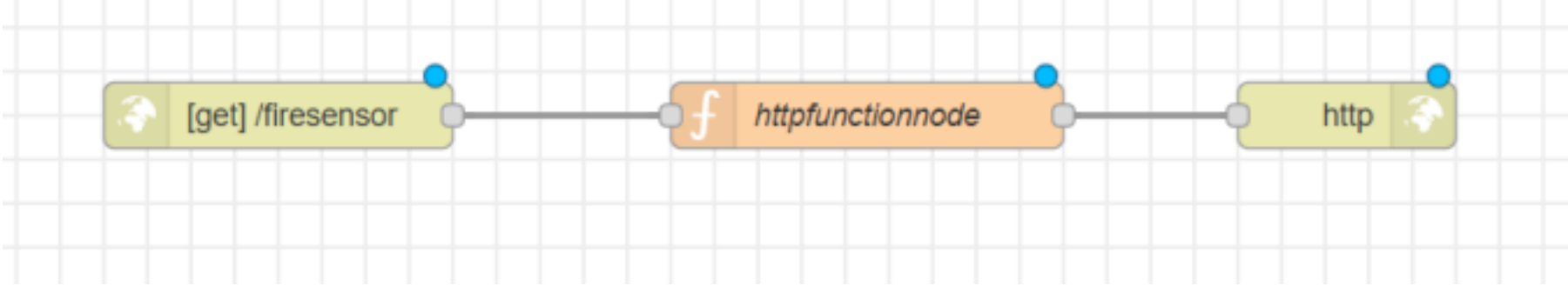


Fig 5 - Using HTTP in and HTTP response in network option, http://127.0.0.1:1880/#flow/f74f1b96473dc208/firesensor will display the sensor values

like Temperature, Gas_Level and Flame_Level from the IBM WATSON IOT PLATFORM.

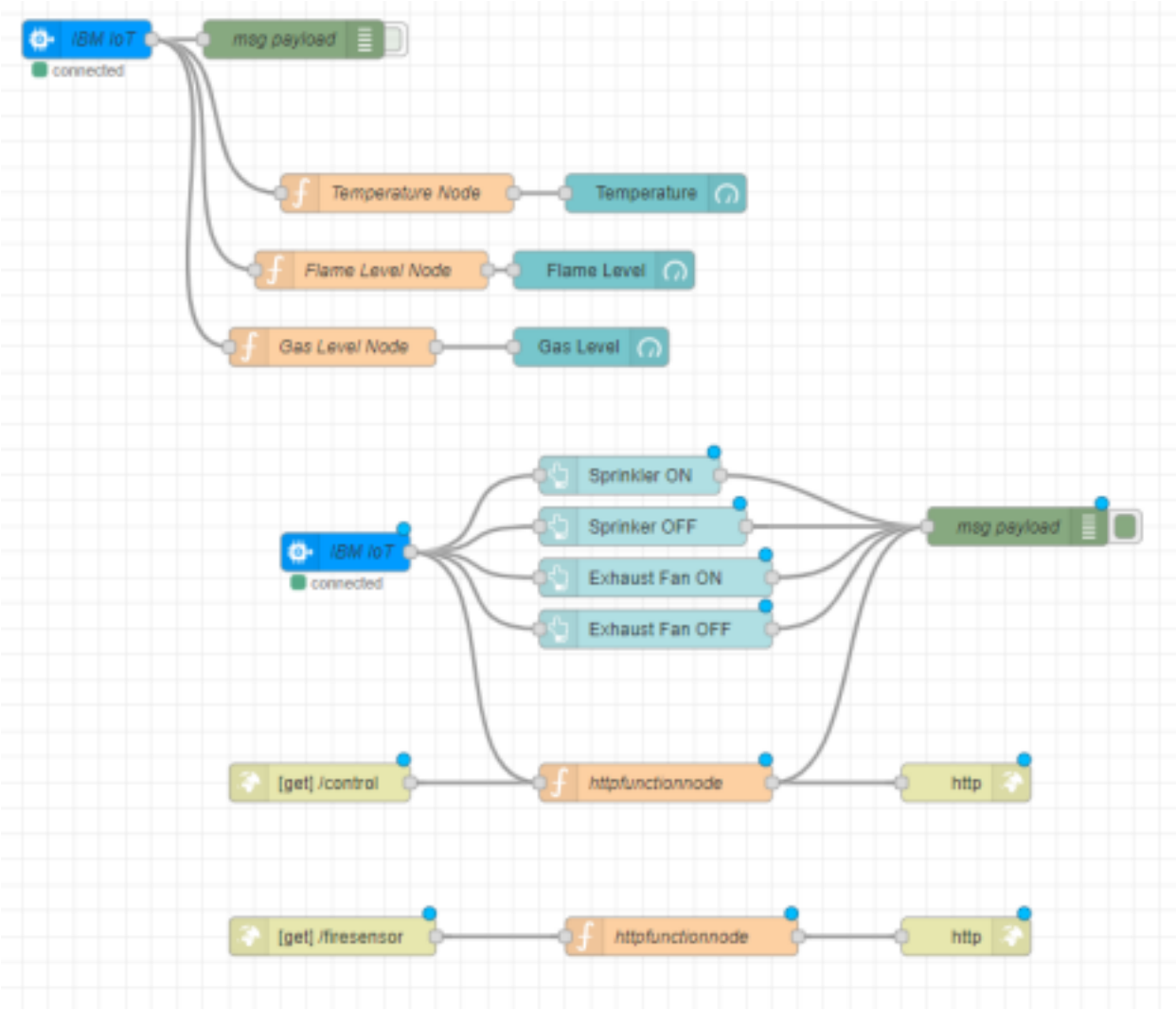


Fig 6 - Entire Node-Red connection for our project

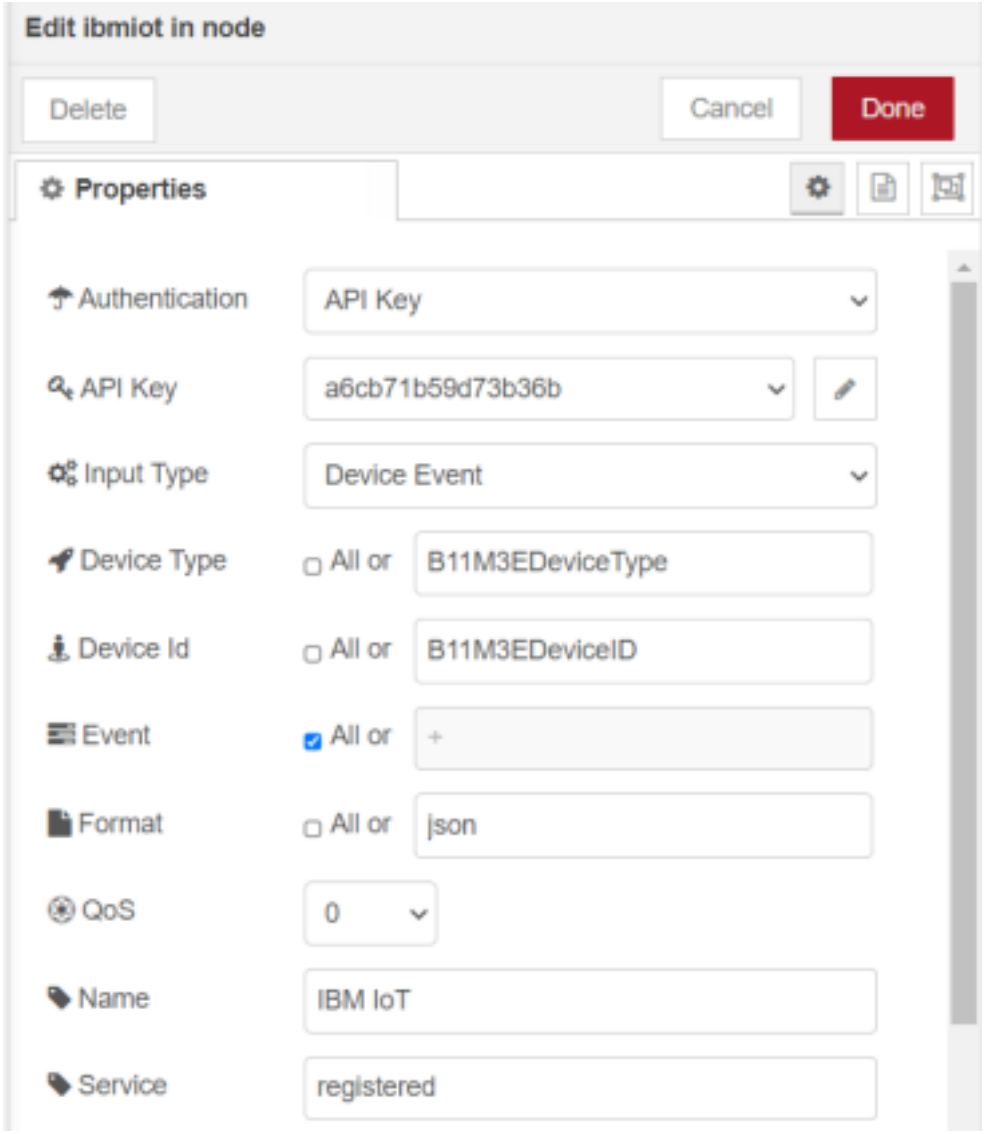
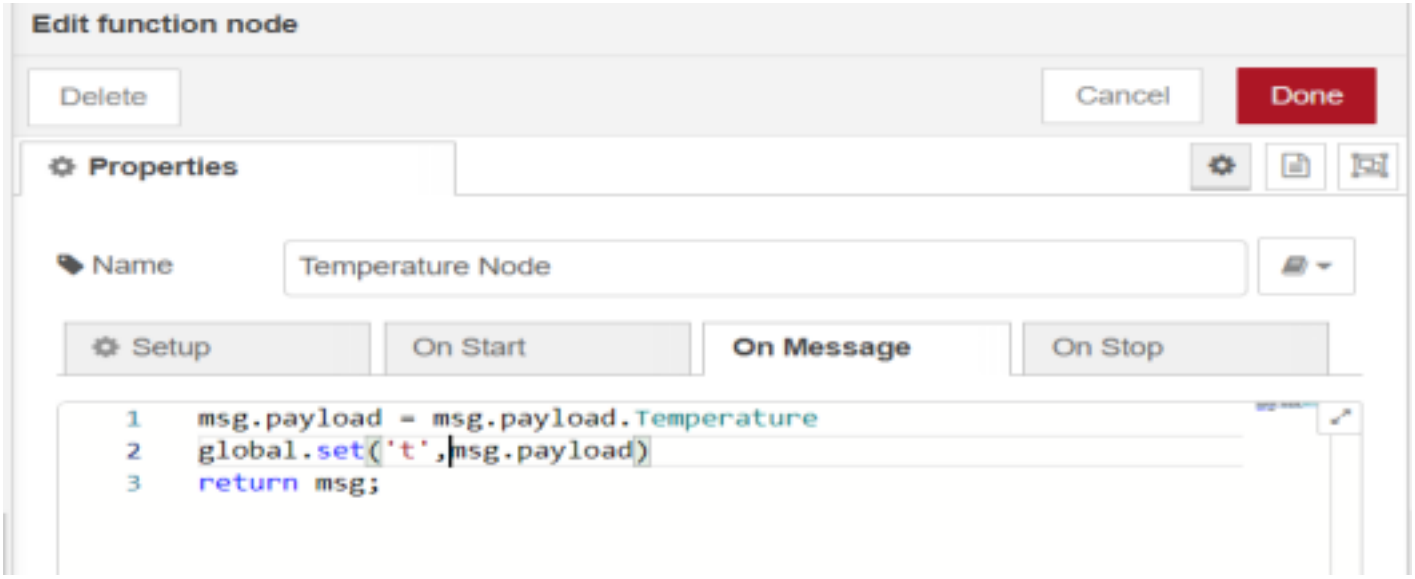
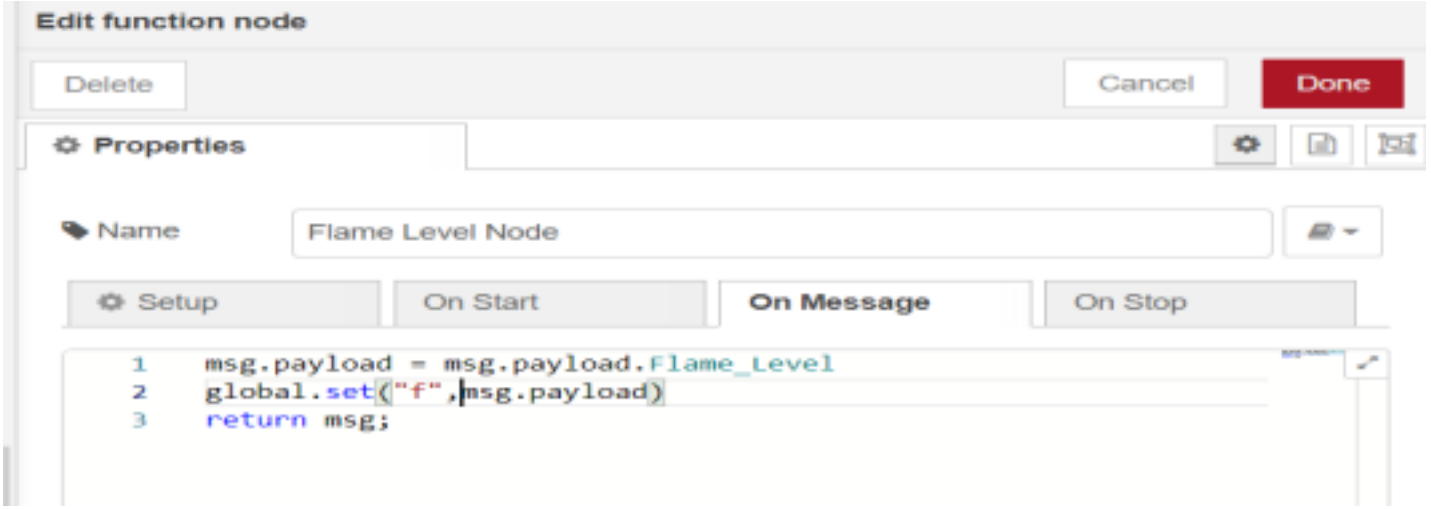


Fig 7 - Properties of IBM IOT are shown. The API key, Device Type, Device ID are taken from IBM IOT WATSON


Edit function node
Delete
Cancel
Done
Properties
Name
Temperature Node
Setup
On Start
On Message
On Stop
1 msg.payload = msg.payload.Temperature
2 global.set('t',msg.payload)
3 return msg;

PLATFORM.


Edit function node
Delete
Cancel
Done
Properties
Name
Flame Level Node
Setup
On Start
On Message
On Stop
1 msg.payload = msg.payload.Flame_Level
2 global.set("f",msg.payload)
3 return msg;

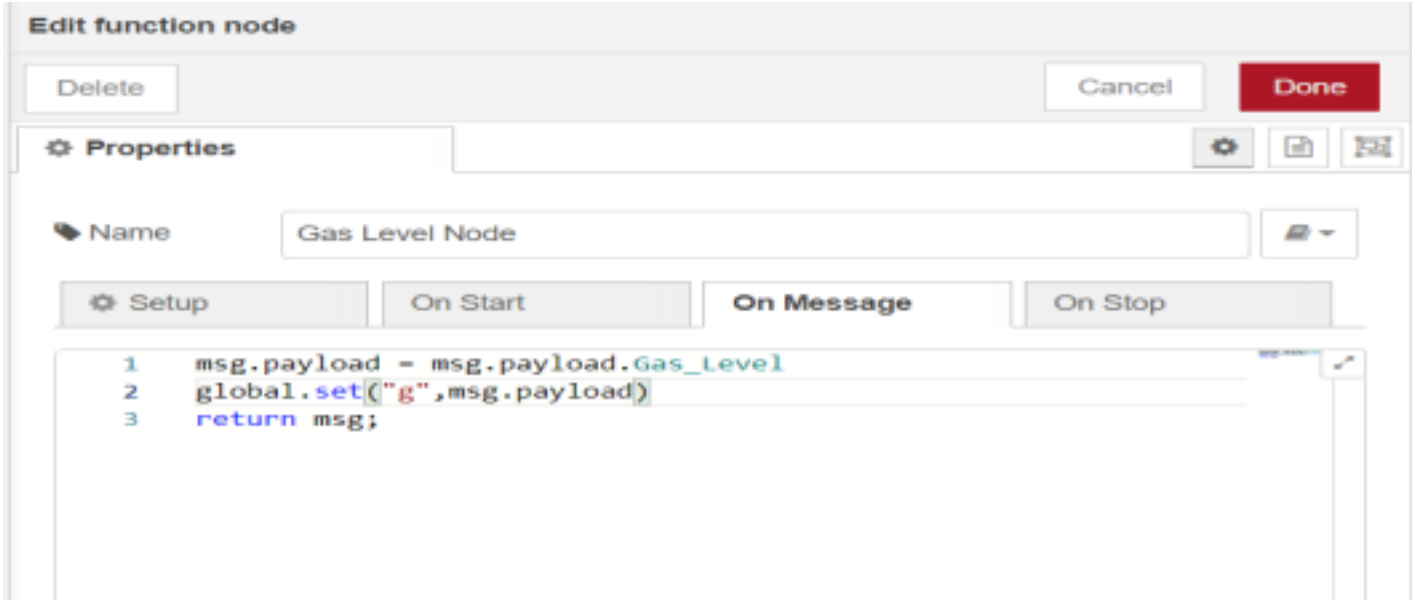


Fig 8 - Properties of Function Node -Temperature Node, Flame_Level Node, Gas_Level Node.

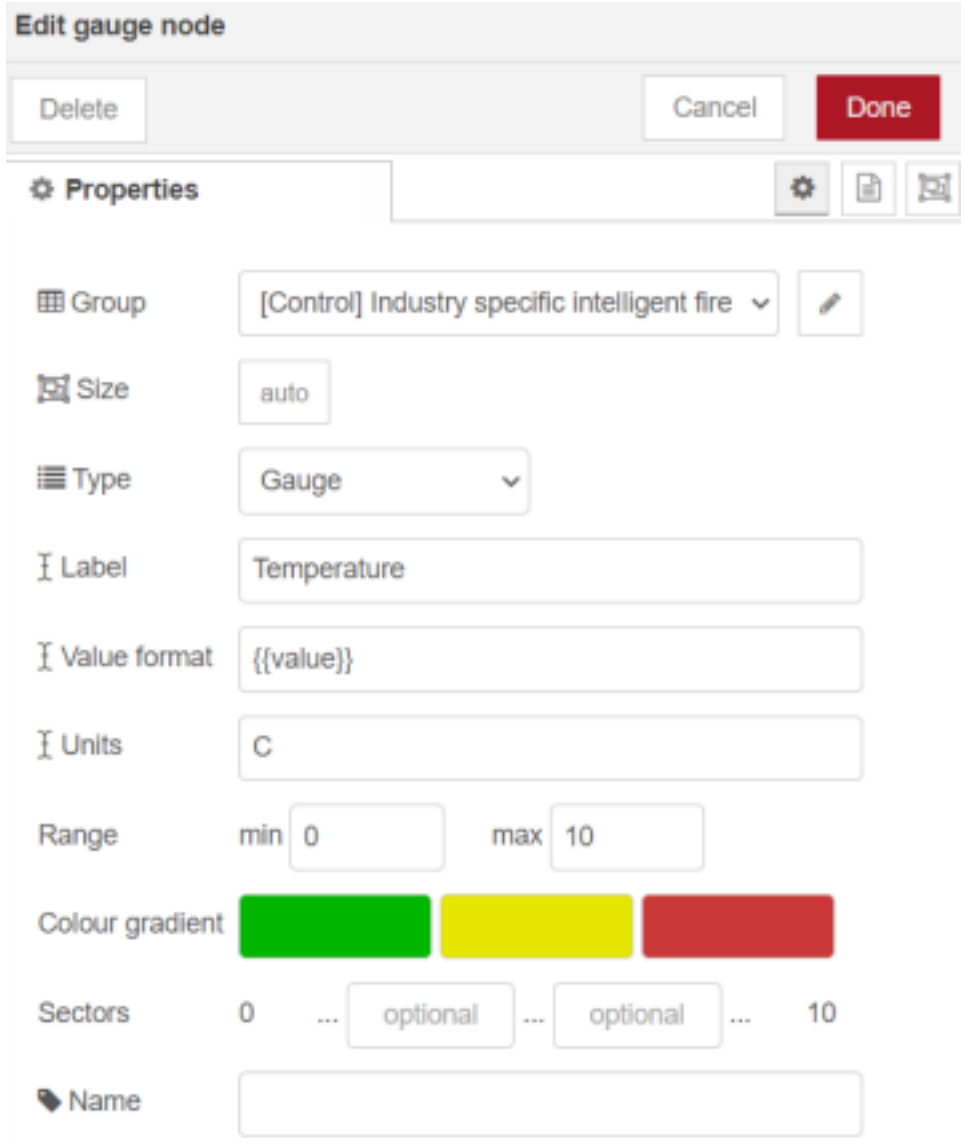


Fig 9 - Properties of Temperature Gauge.

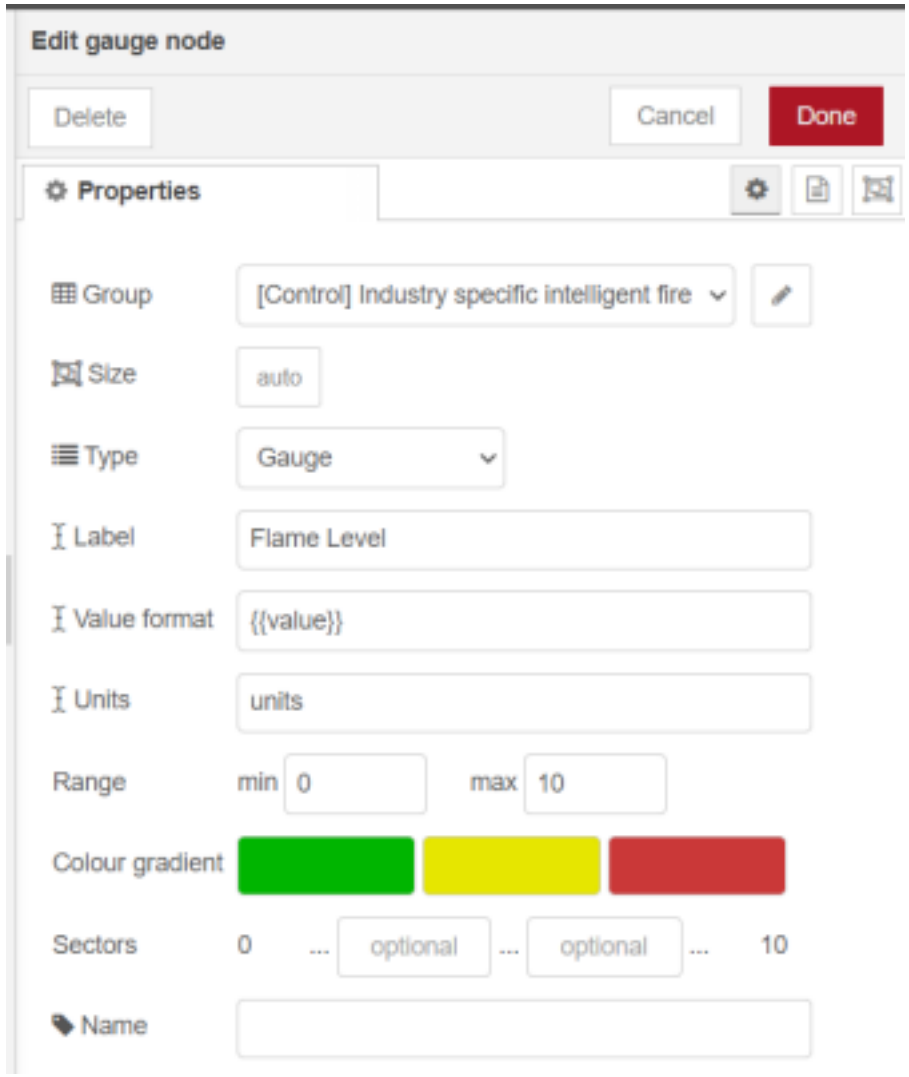


Fig 9 - Properties of Flame_Level Gauge.

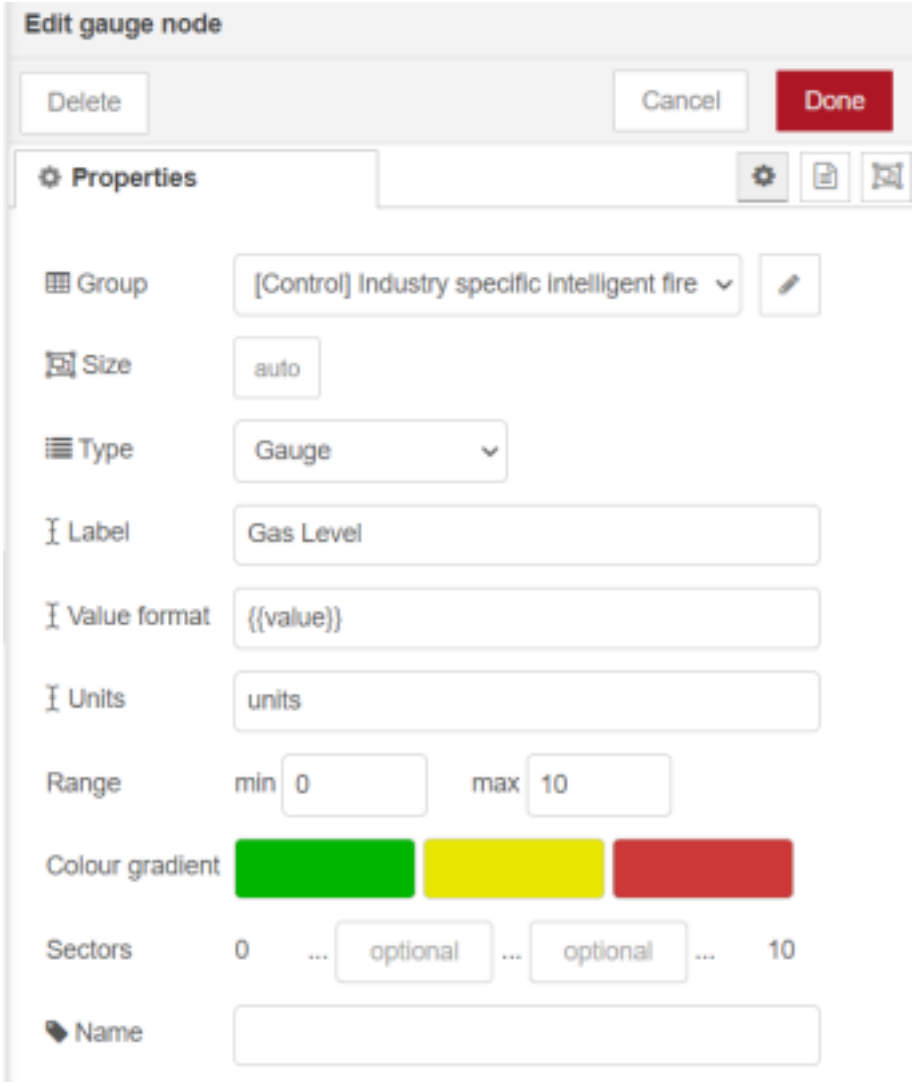


Fig 9 - Properties of Gas_Level Gauge.

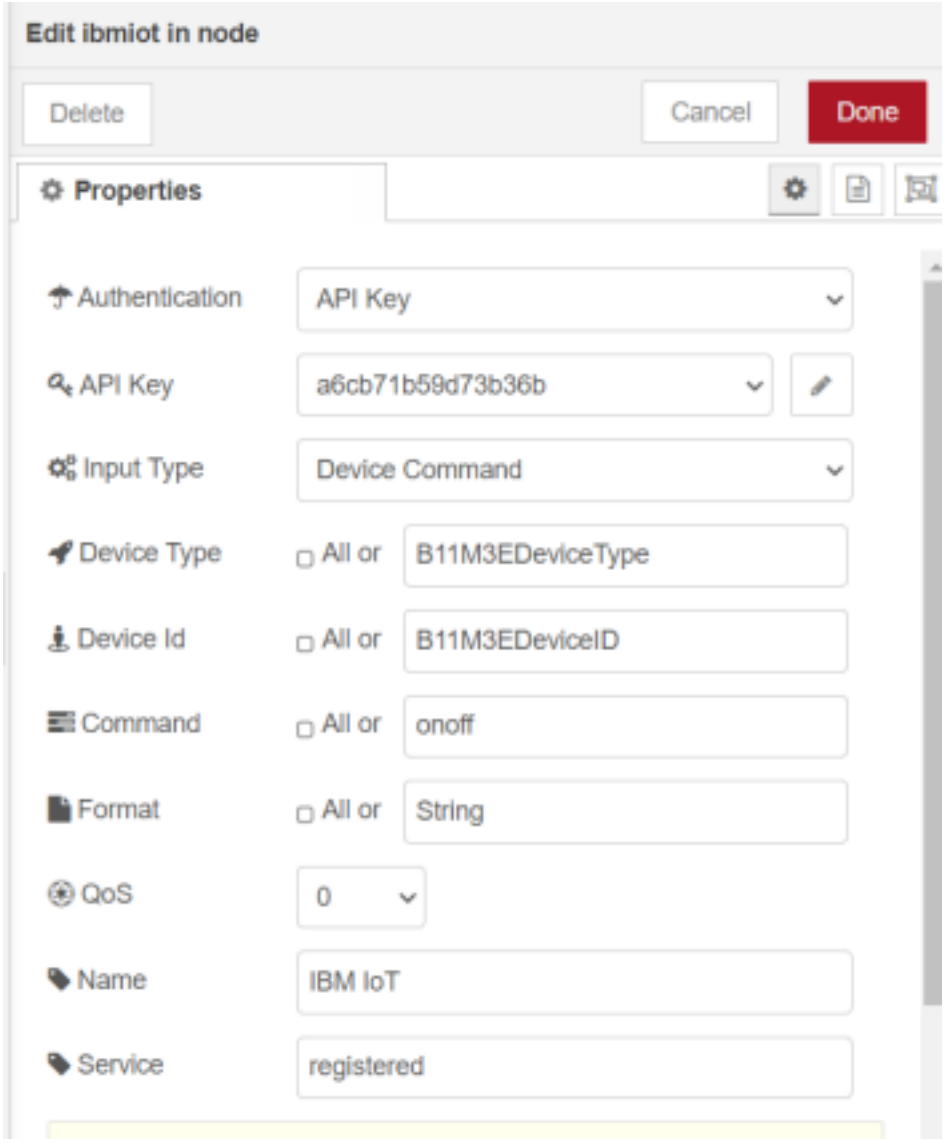


Fig 9 - Properties of IBM IOT Node.

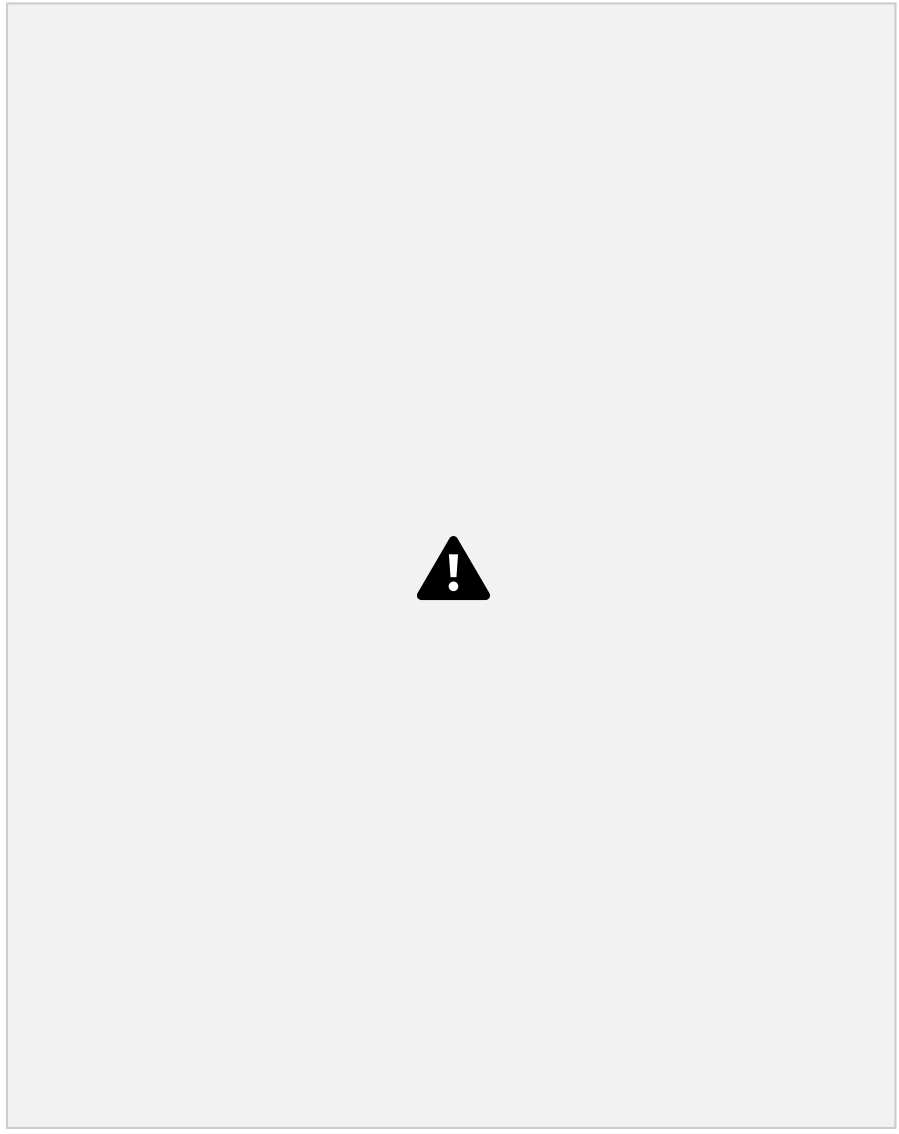


Fig 10 - Properties of Sprinkler ON button node.

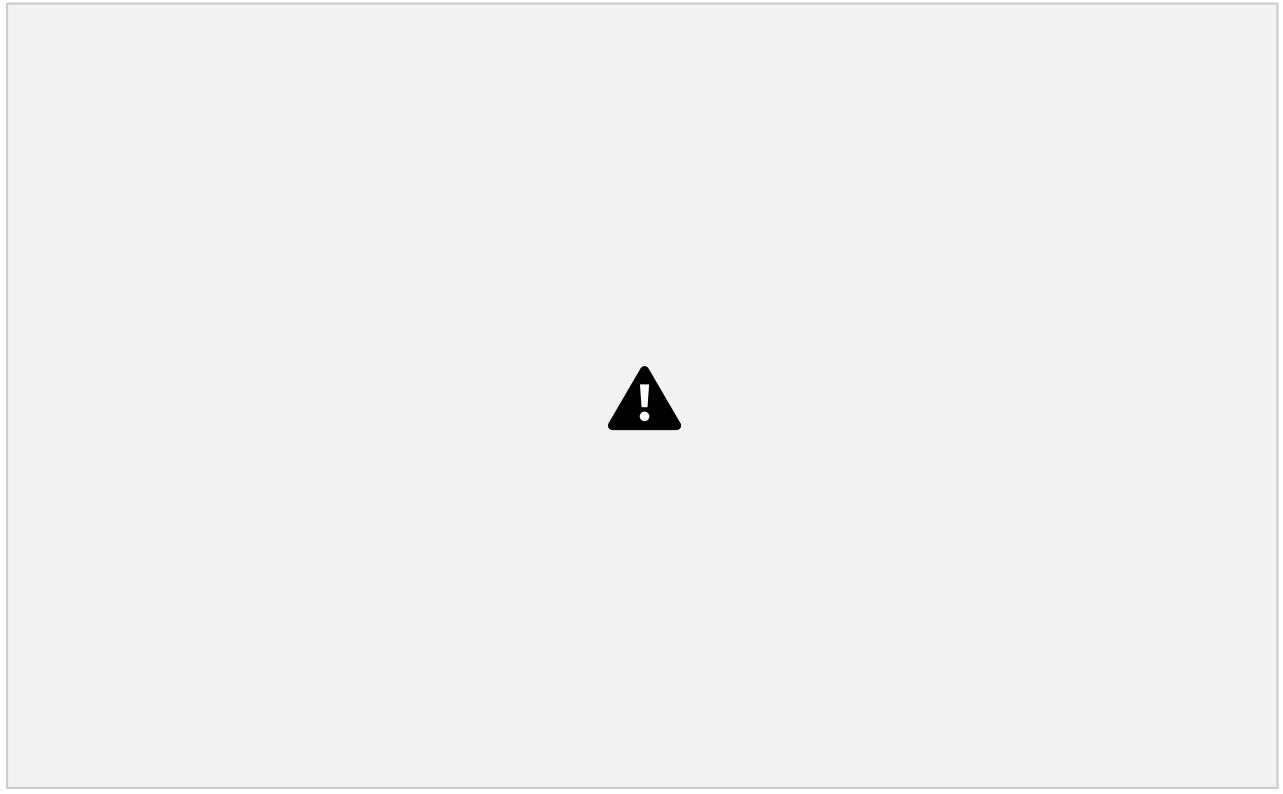


Fig 10 - Properties of HTTP Node with method GET and URL /control,

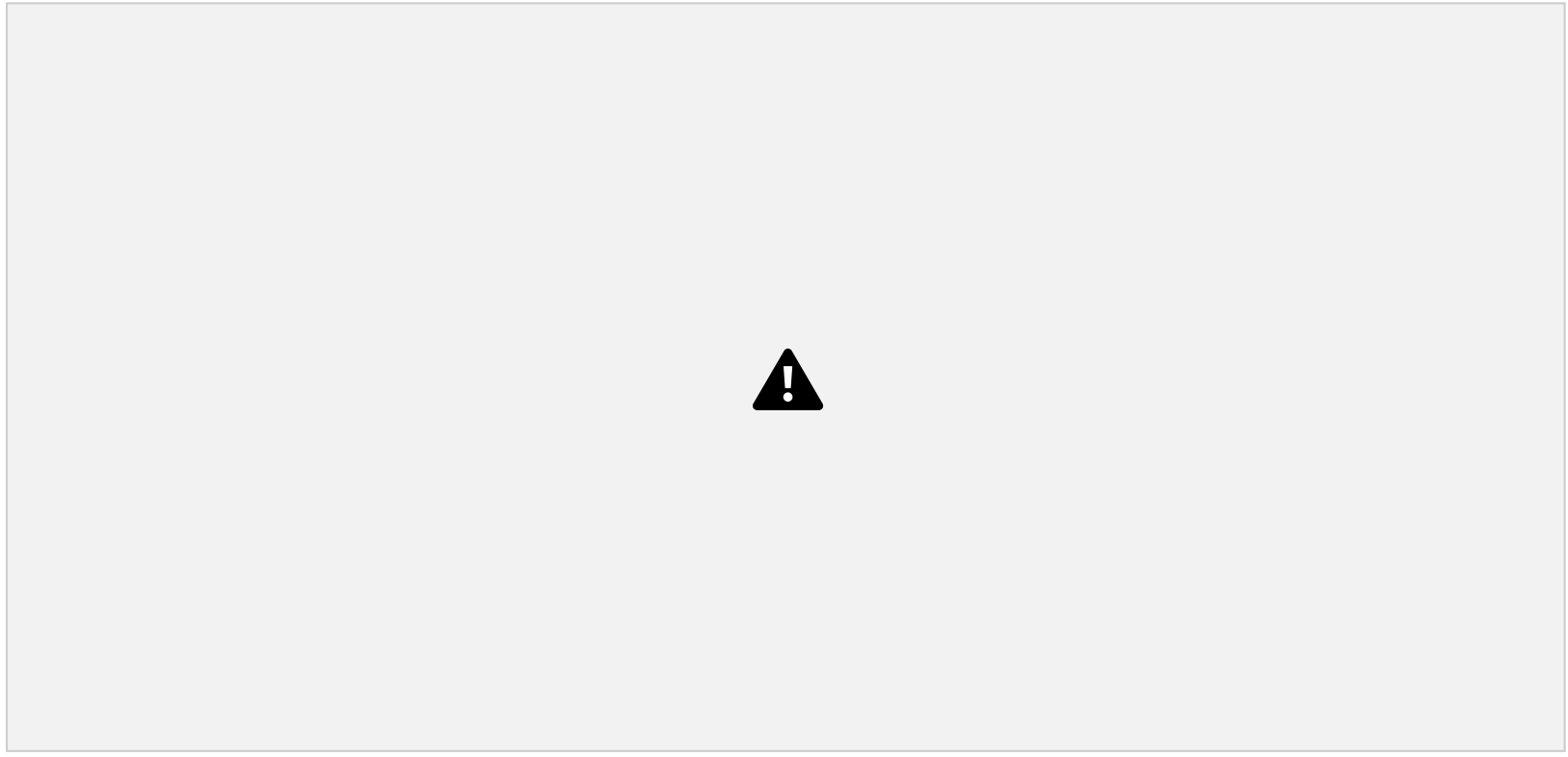
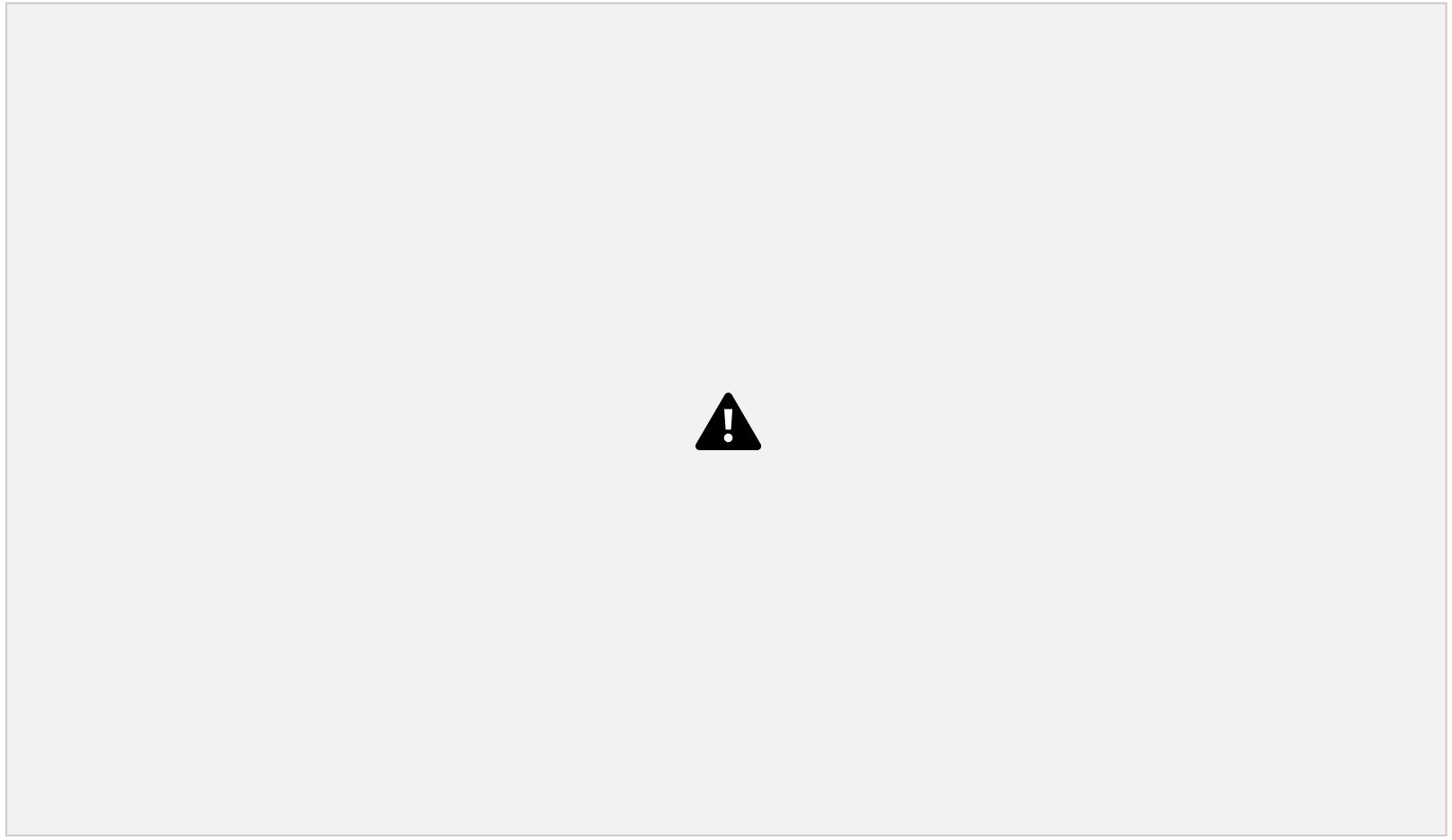


Fig 11 - Properties of Control HTTP Function Node.



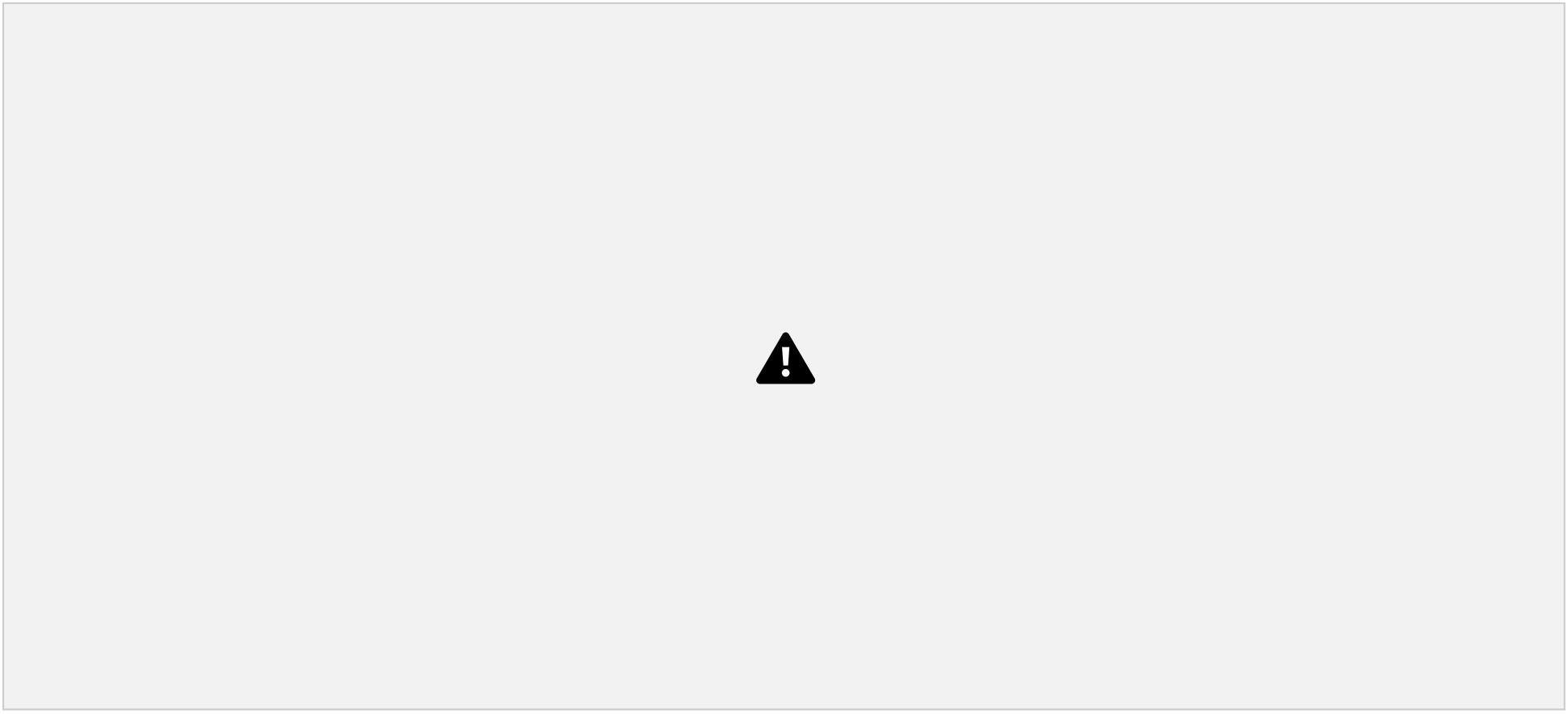


Fig 12 - Properties of Monitor HTTP Function Node

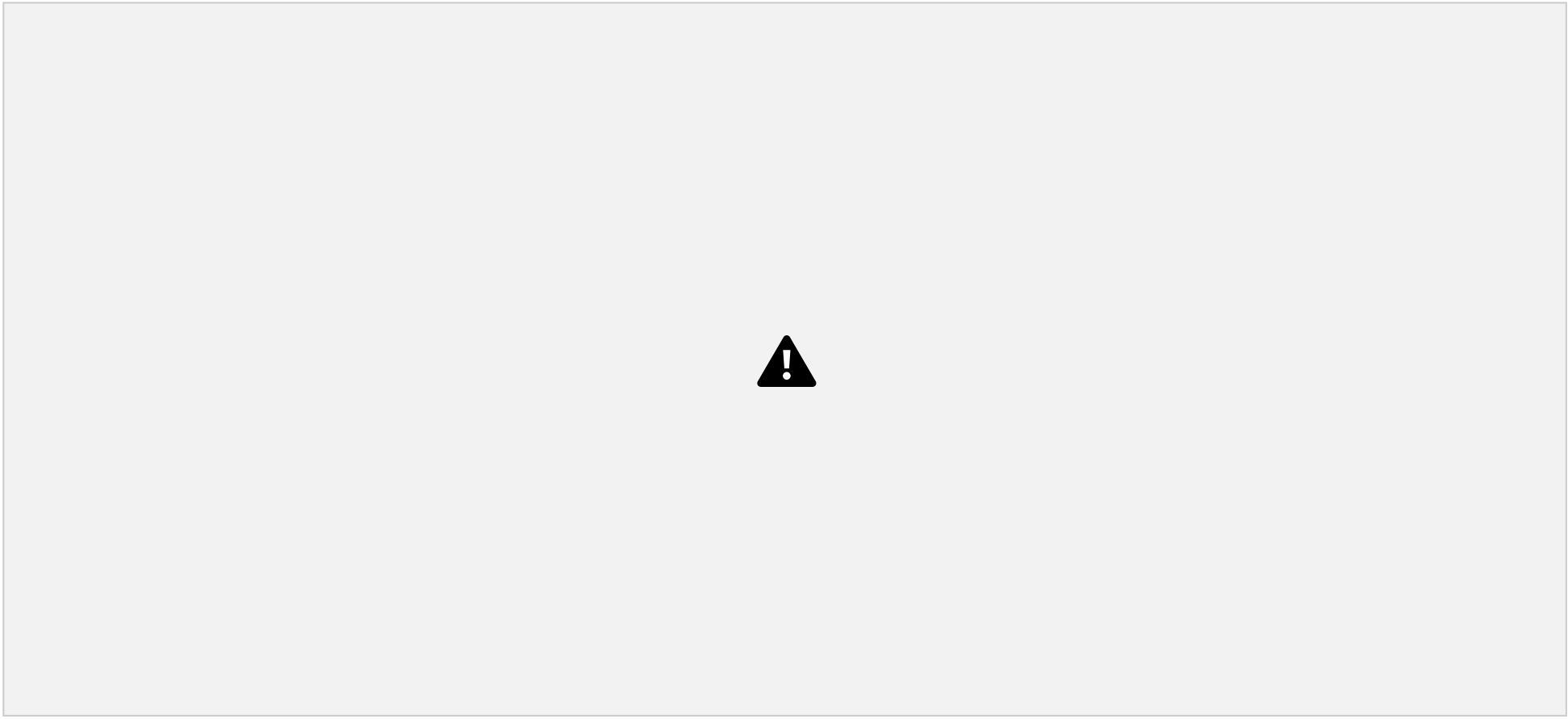


Fig 13 - Front-end APP for our project, to display the Temperature Level, Smoke Level and Flame Level with control buttons like Sprinkler ON and OFF and Exhaust Fan ON and OFF