

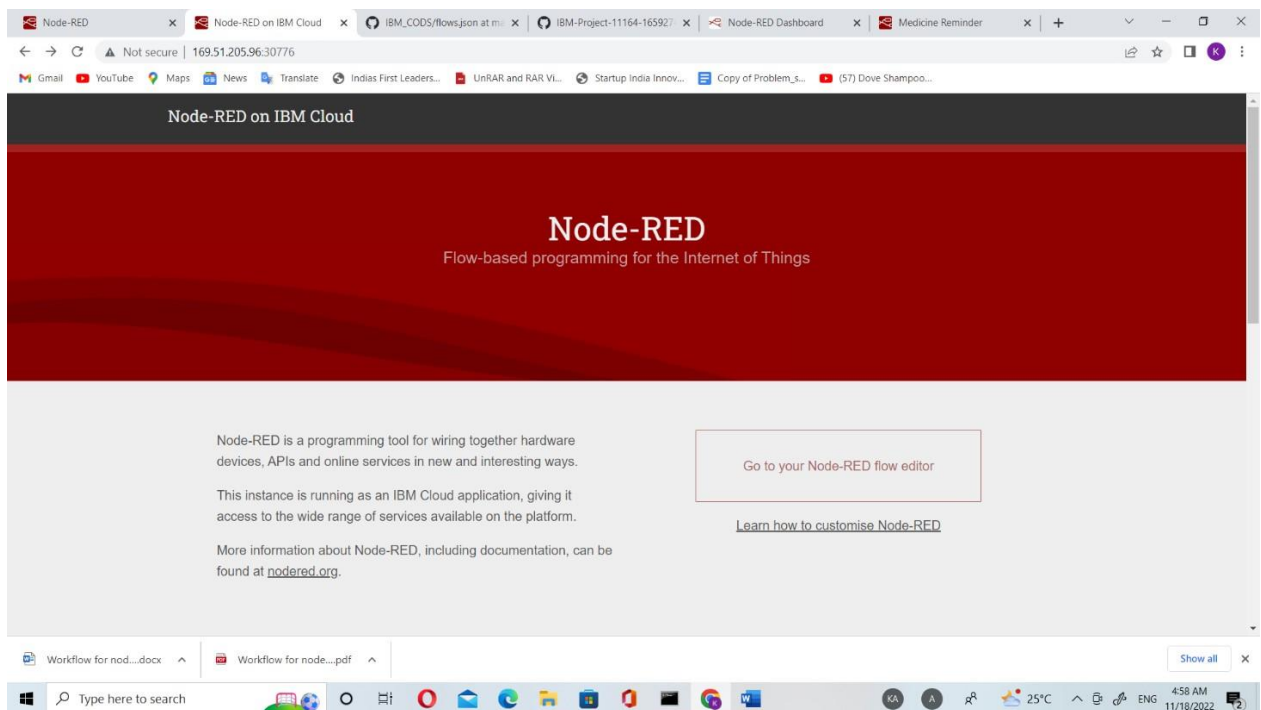
SPRINT DELIVERY – 3

TEAM ID : PNT2022TMID19314

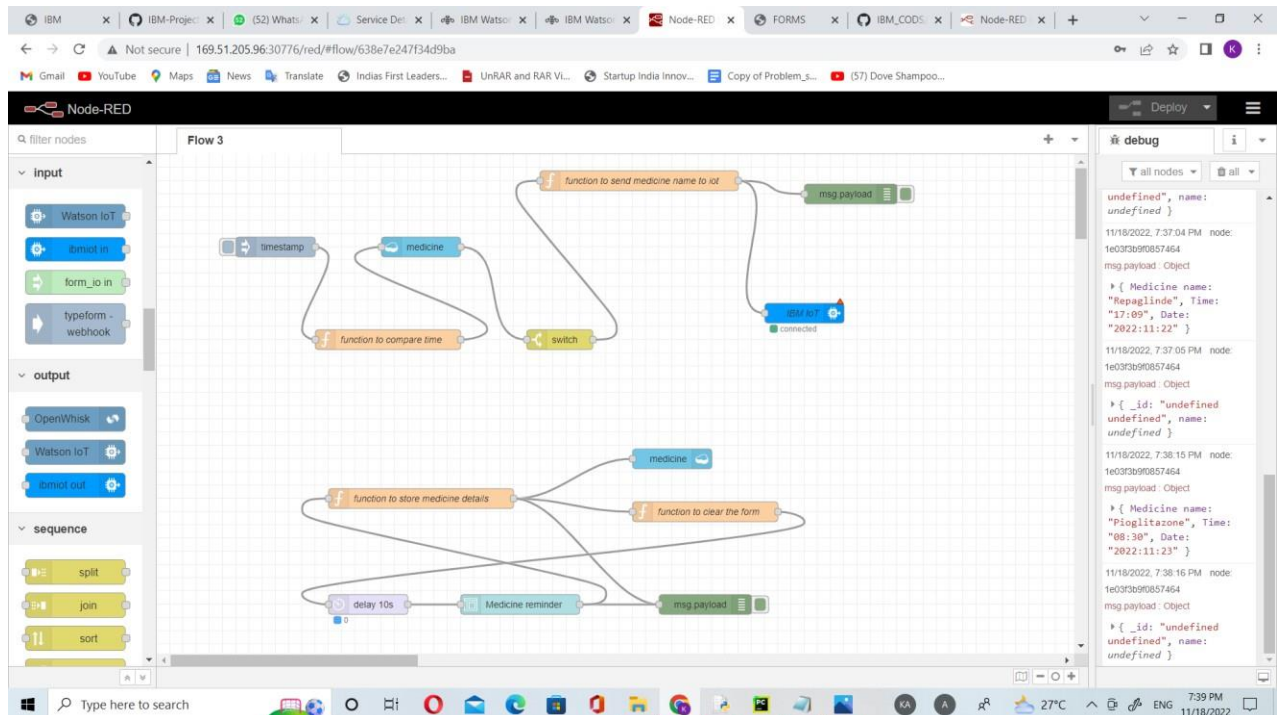
PROJECT NAME: PERSONAL ASSISTANCE FOR SENIORS WHO ARE SELF RELIANT

WORKFLOW FOR IOT SCENARIOS USING NODE RED (CREATING WEB UI)

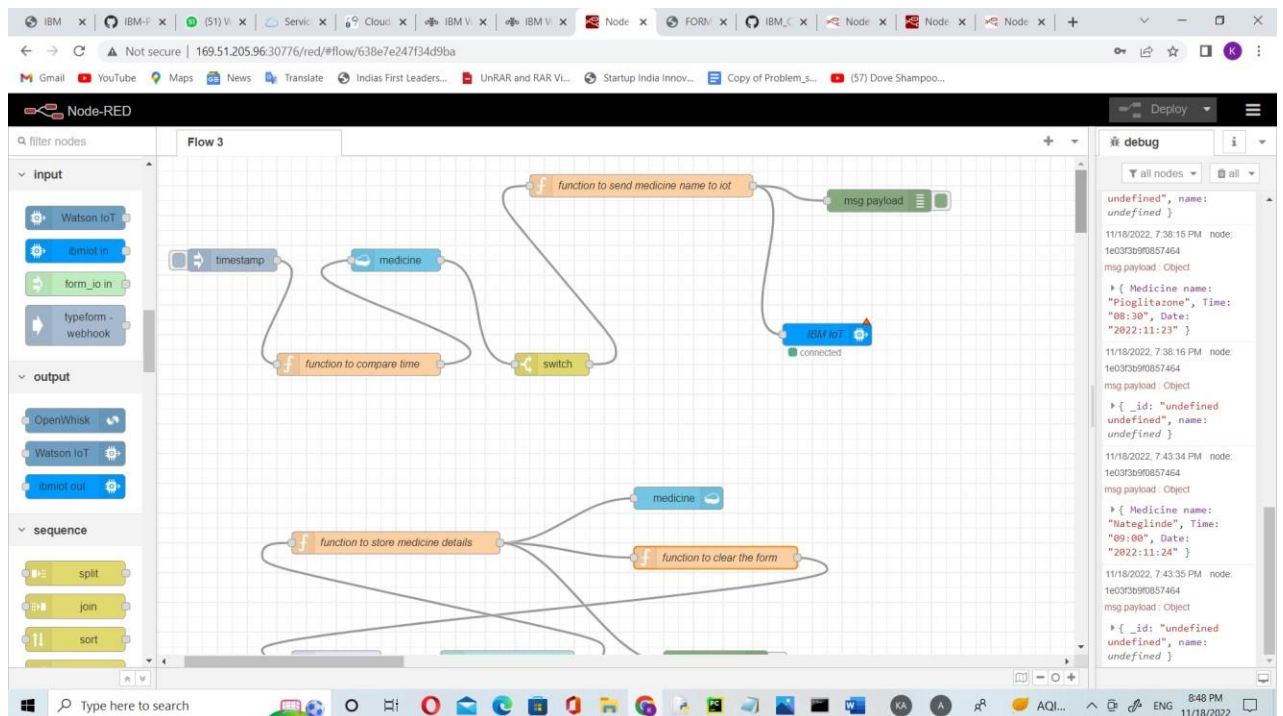
NODE-RED INSTALLATION:



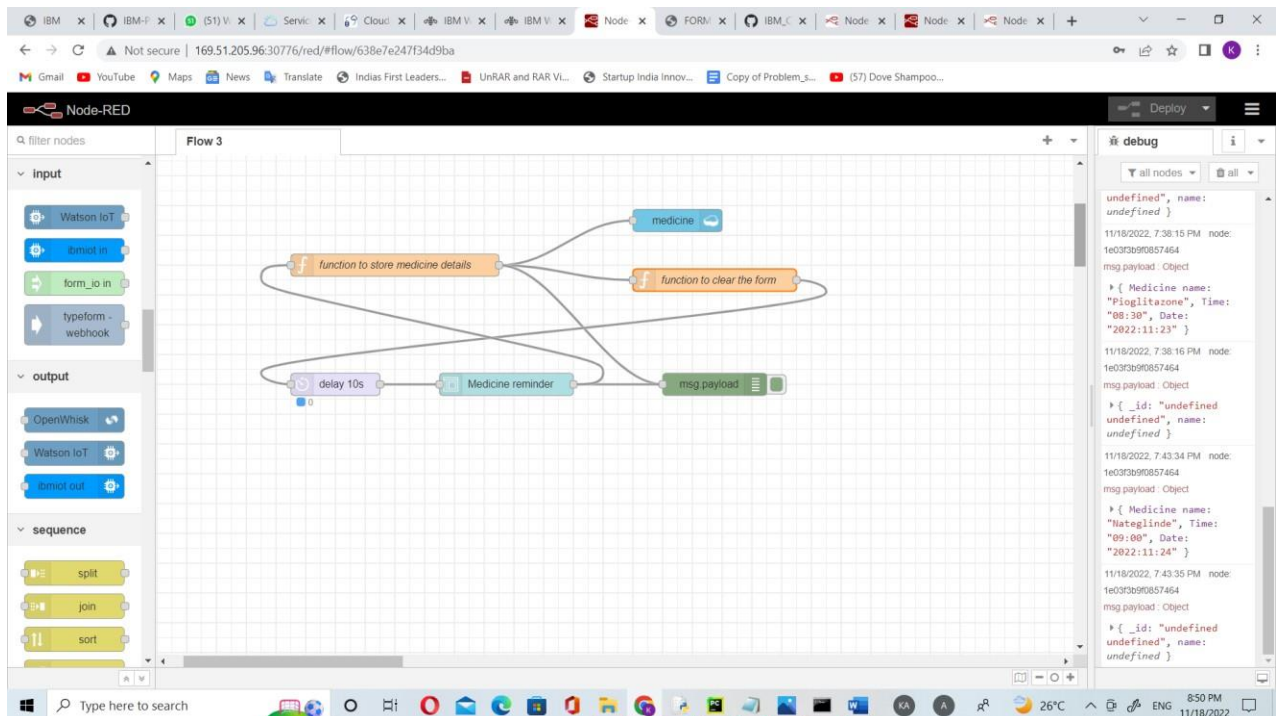
CREATING MEDICINE REMINDER FORM: (COMPLETE)



SENDING MEDICINE NAME AT APPROPRIATE TIME:

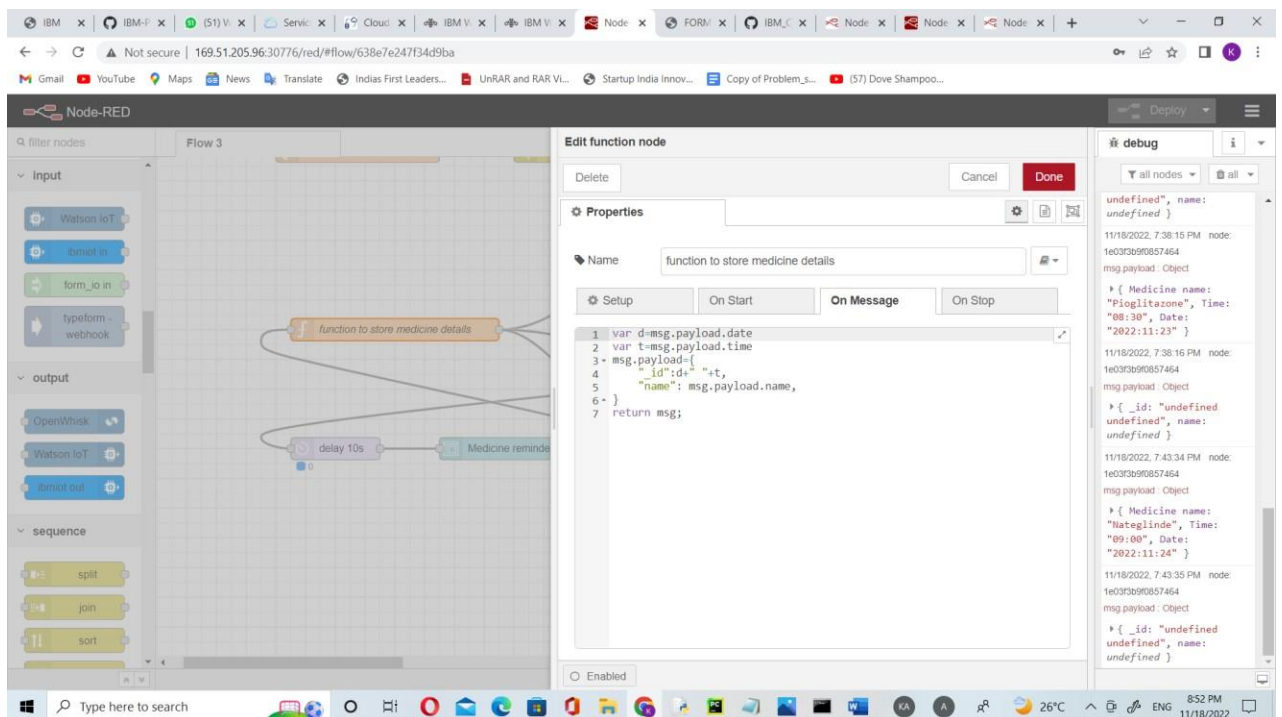


TO ENTER MEDICINE DETAILS:



FUNCTIONS:

1. FUNCTION TO STORE MEDICINE DETAILS



2. FUNCTION TO CLEAR THE FORM

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow with several nodes: 'form_io_in' (input), 'typeform-webhook' (input), 'OpenWhisk' (output), 'Watson IoT' (output), 'delay 10s' (delay), and 'Medicine reminder' (output). A function node is highlighted, and its configuration panel is open on the right. The function node is named 'function to clear the form' and is set to trigger 'On Message'. The code in the function node is as follows:

```
1 msg.payload={
2   "date": "",
3   "name": "",
4   "time": ""
5 }
6 return msg;
```

The debug console on the right shows the message payload being sent to the function node, which is an object with 'date', 'name', and 'time' properties, all set to empty strings.

3. FUNCTION TO SEND MEDICINE NAME TO IOT

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow with several nodes: 'timestamp' (input), 'medicine' (input), 'function to compare time' (function), 'function to store medicine details' (function), and 'OpenWhisk' (output). A function node is highlighted, and its configuration panel is open on the right. The function node is named 'function to send medicine name to iot' and is set to trigger 'On Message'. The code in the function node is as follows:

```
1 msg.payload={"command":msg.payload.name}
2 return msg;
```

The debug console on the right shows the message payload being sent to the function node, which is an object with 'command' and 'name' properties, both set to empty strings.

4.FUNCTION TO COMPARE TIME

The screenshot shows the Node-RED web interface. A flow named 'Flow 3' is visible, containing a 'timestamp' node, a 'medicine' node, and a 'function to compare time' node. The 'function to compare time' node is selected, and its configuration panel is open. The configuration panel shows the node name 'function to compare time' and the following JavaScript code:

```
1 var d= new Date();
2 var utc= d.getTime()+ (d.getTimezoneOffset() *60000);
3 var offset=5.5;
4 newDate = new Date(utc + (3600000*offset));
5 var n=newDate.toISOString()
6 var date= n.slice(0,10)
7 var time=n.slice(11,16)
8 global.set('time', time)
9 msg.payload=date+" "+time
10 return msg;
```

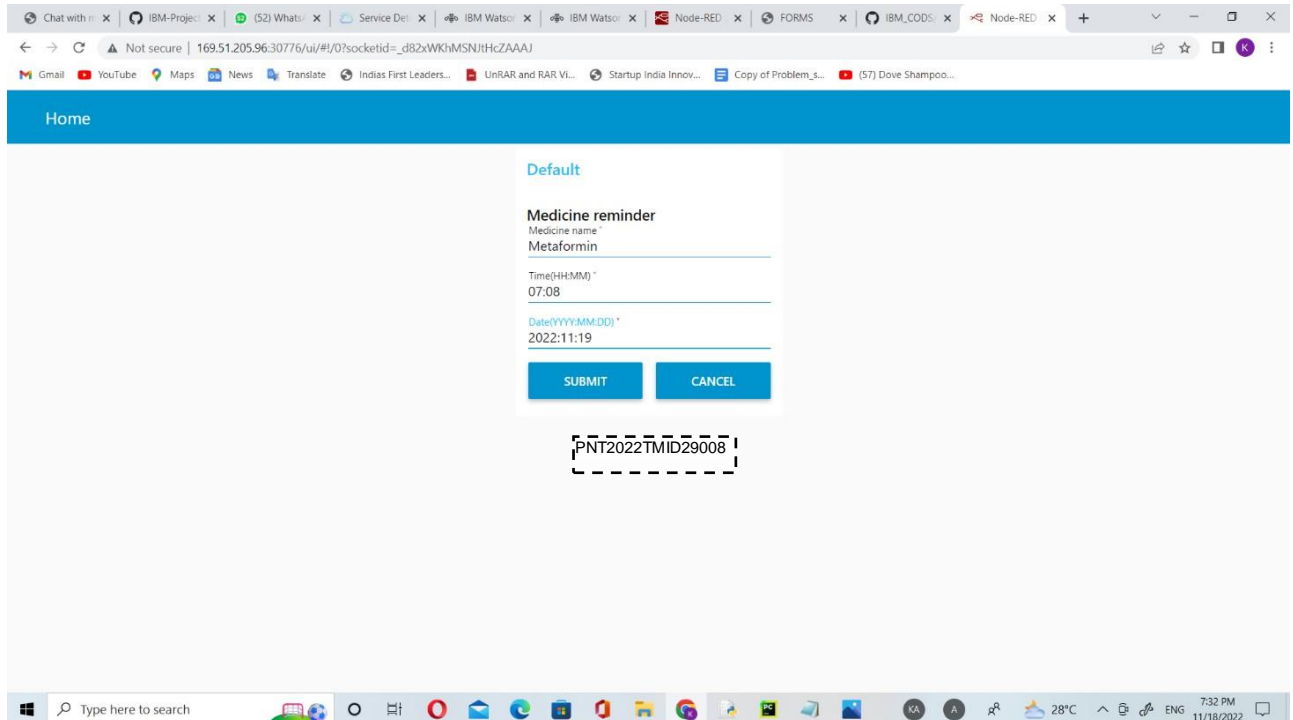
The debug console on the right shows the output of the function, displaying the date and time in ISO format.

WEB UI(MEDICINE REMINDER PAGE):

The screenshot shows the 'Medicine reminder' web page. The page has a blue header with the text 'Home'. Below the header, there is a form titled 'Medicine reminder' with the following fields:

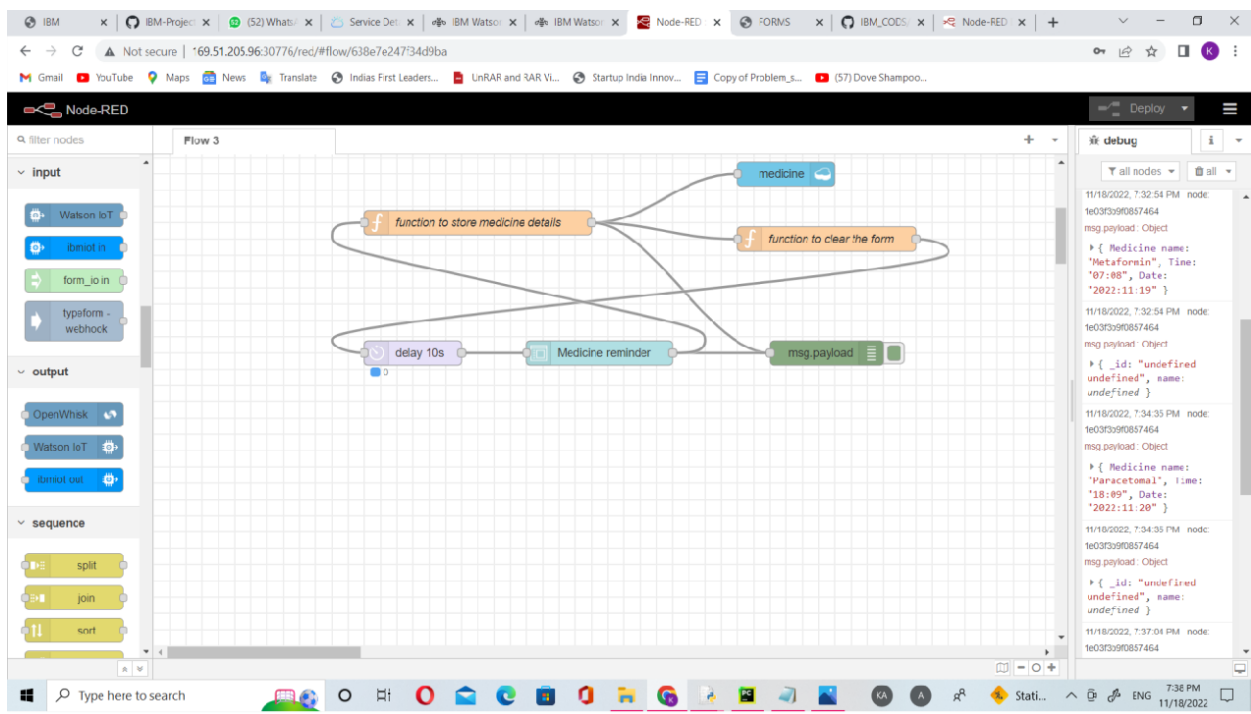
- Medicine name *
- Time(HH:MM) *
- Date(YYYY:MM:DD) *

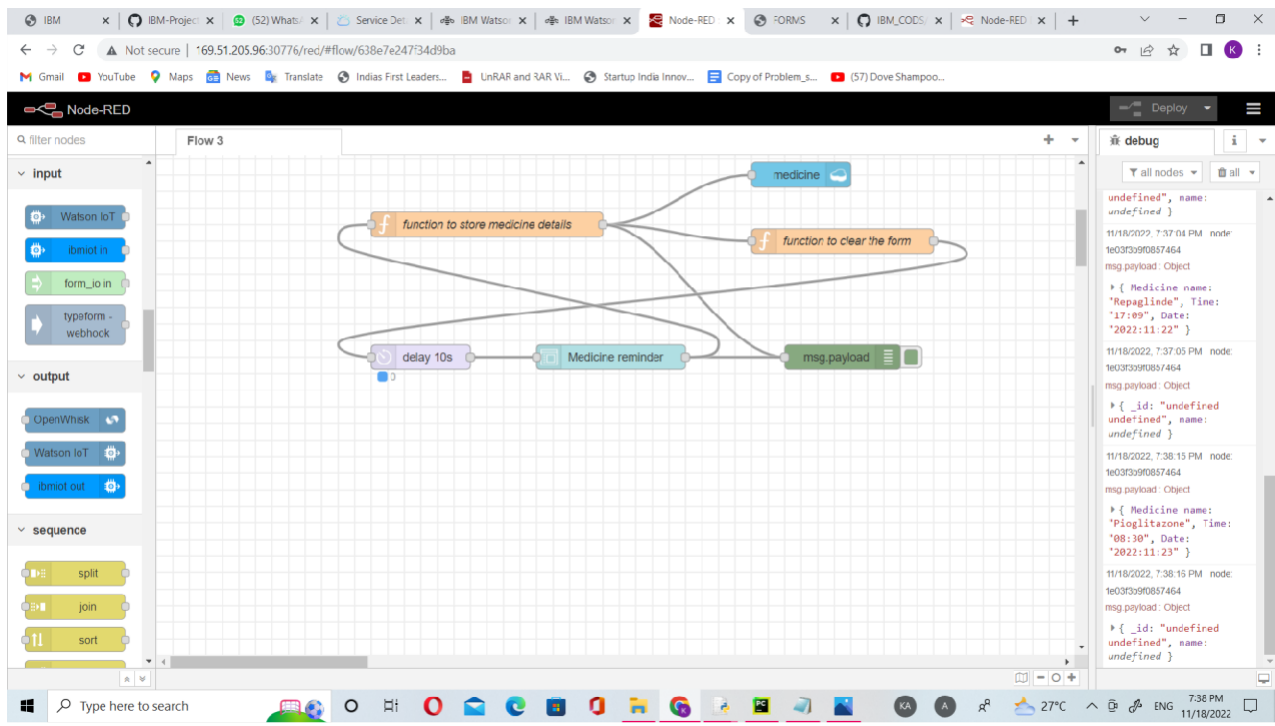
At the bottom of the form, there are two buttons: 'SUBMIT' and 'CANCEL'.



DEBUG WINDOW:

The details of medicine name,date and time that are entered are shown in debug window.





MEDICINE DATABASE IN CLOUDANT DB:

The Cloudant dashboard screenshot displays a database named 'medicine'. The left sidebar contains navigation options: 'All Documents', 'Query', 'Permissions', 'Changes', and 'Design Documents'. The main area shows a table view of the database contents. The table has two columns: '_id' and 'name'. The data is as follows:

_id	name
Time:07:08 Date:2022:11:19	{ "name": "metformin" }
Time:08:30 Date:2022:11:23	{ "name": "Pioglitazone" }
Time:09:00 Date:2022:11:24	{ "name": "Natalinde" }
Time:17:09 Date:2022:11:22	{ "name": "Repaglinde" }
Time:18:09 Date:2022:11:18	{ "name": "paracetamol" }

The bottom of the dashboard shows a status bar indicating 'Showing 2 of 3 columns' and 'Showing document 1 of 5'.

The screenshot shows the IBM Watson IoT Platform dashboard for a device named 'medicine'. The left sidebar contains navigation options: All Documents, Query, Permissions, Changes, and Design Documents. The main area displays a list of documents in JSON format. The first document has an ID of 'Time:07:08 Date:2022:11:19' and contains a JSON object with 'id', 'key', 'value', 'rev', and 'doc' fields. The second document has an ID of 'Time:08:30 Date:2022:11:23' and contains a similar JSON object. The bottom status bar indicates 'Showing document 1 of 5' and 'Documents per page: 20'.

IBM WATSON IOT PLATFORM:

The screenshot shows the IBM Watson IoT Platform dashboard for a device named '12345'. The left sidebar contains navigation options: Browse, Action, Device types, and Interfaces. The main area displays a table of devices with columns: Device ID, Status, Device Type, Class ID, Date Added, and Descriptive Location. The device '12345' is listed as 'Disconnected' and is of type 'Iotsensor'. Below the table, the 'Recent Events' tab is selected, showing a list of events with columns: Event, Value, Format, and Last Received. The events are listed as 'event_1' with values like 'Medicine name: "Nateglindol"', 'Medicine name: "Pioglitazone"', 'Medicine name: "Repaglinide"', 'Medicine name: "Paracetamol"', and 'Medicine name: "Metformin"'. The bottom status bar indicates '1 Simulation running'.

SIGNUP FORM IN NODE RED:

In debug window the details are stored.

The screenshot shows the Node-RED web interface in a browser. The main workspace displays a flow with the following nodes: a `[get]/signup` node, a `function 2` node, a `function 3` node, a `delay 5s` node, a `signup` node, and a `debug 1` node. The flow starts with the `[get]/signup` node, which connects to `function 2`. `function 2` connects to `function 3`, which then connects to the `debug 1` node. A `delay 5s` node is also connected to the `signup` node, which also connects to the `debug 1` node. The left sidebar shows the 'filter nodes' and 'network' categories. The right sidebar shows the 'debug' console with a list of messages. The messages are as follows:

Time	Node	Message
11/18/2022, 8:37:27 PM	node: debug 1	msg payload: Object
11/18/2022, 8:39:18 PM	node: debug 1	{ empty: }
11/18/2022, 8:39:18 PM	node: debug 1	msg payload: Object
11/18/2022, 8:39:18 PM	node: debug 1	{ Username: "keerthana", Password: "12345" }
11/18/2022, 8:39:18 PM	node: debug 1	msg payload: Object
11/18/2022, 8:37:43 PM	node: debug 1	{ Username: "keerthana", Password: "12345" }
11/18/2022, 8:37:43 PM	node: debug 1	msg payload: Object
11/18/2022, 8:37:43 PM	node: debug 1	{ Username: "keerthana", Password: "12345", email: "keerthanaandh4@gmail.com", age: "20" }
11/18/2022, 8:37:43 PM	node: debug 1	msg payload: Object
11/18/2022, 8:37:43 PM	node: debug 1	{ Username: "keerthana", Password: "12345", email: "keerthanaandh4@gmail.com", age: "20" }

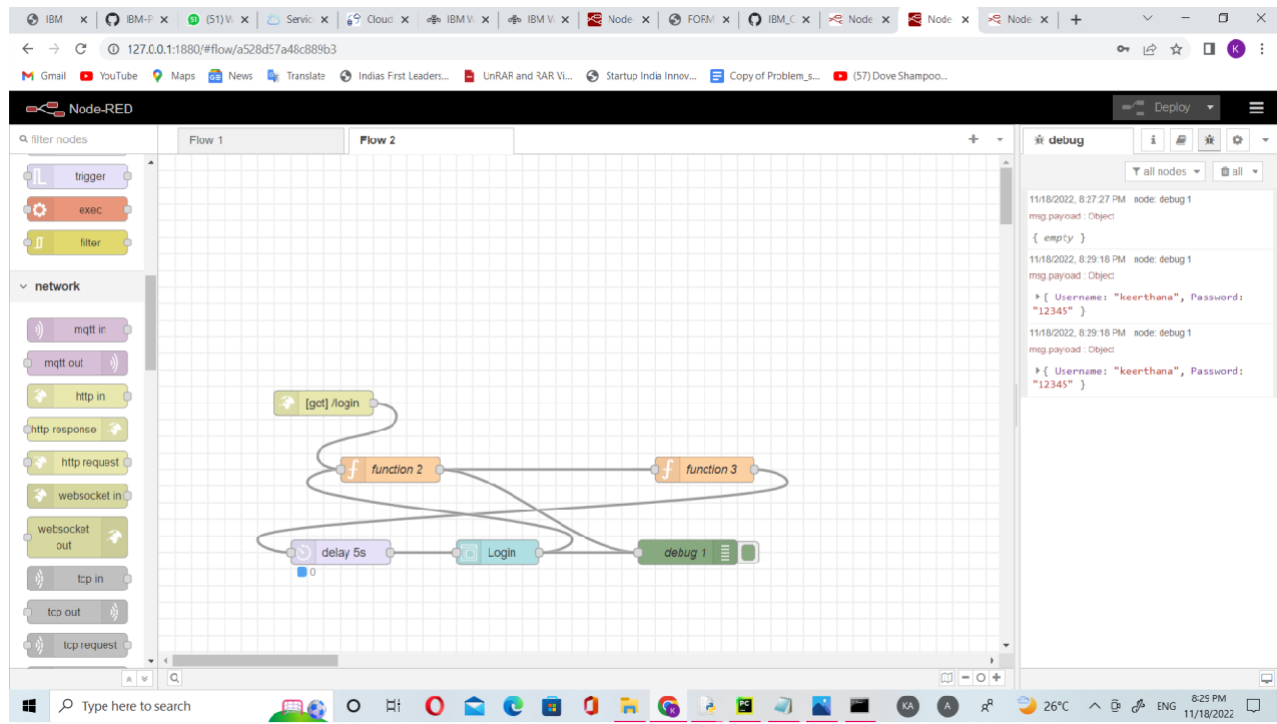
WEBUI(SIGNUP):

The screenshot shows a web browser displaying a 'Signup' form. The form has the following fields:

- Username:
- Password:
- Email:
- Age:

Below the form are two buttons: 'SUBMIT' and 'CANCEL'. Below the buttons is a dashed box containing the text 'PNT2022TMID29008'.

LOGIN FORM USING NODE RED:



The screenshot shows a web browser displaying a login form. The form is titled "Login" and has a "Default" tab. The form fields are:

- Username:** A text input field containing the value "keerthana".
- Password:** A password input field containing the value "12345".
- Buttons:** Two buttons labeled "SUBMIT" and "CANCEL".

Below the form, there is a text box containing the string "PNT2022TMID29008".

MEDICINE REMINDER PAGE:

[http://169.51.205.96:30776/ui/#!/0?socketid= d82xWKhMSNJtHcZAAAJ](http://169.51.205.96:30776/ui/#!/0?socketid=d82xWKhMSNJtHcZAAAJ)

SIGNUP PAGE:

<http://127.0.0.1:1880/ui/#!/0?socketid=5aWJRQVNq8uphtMhAAAA>

