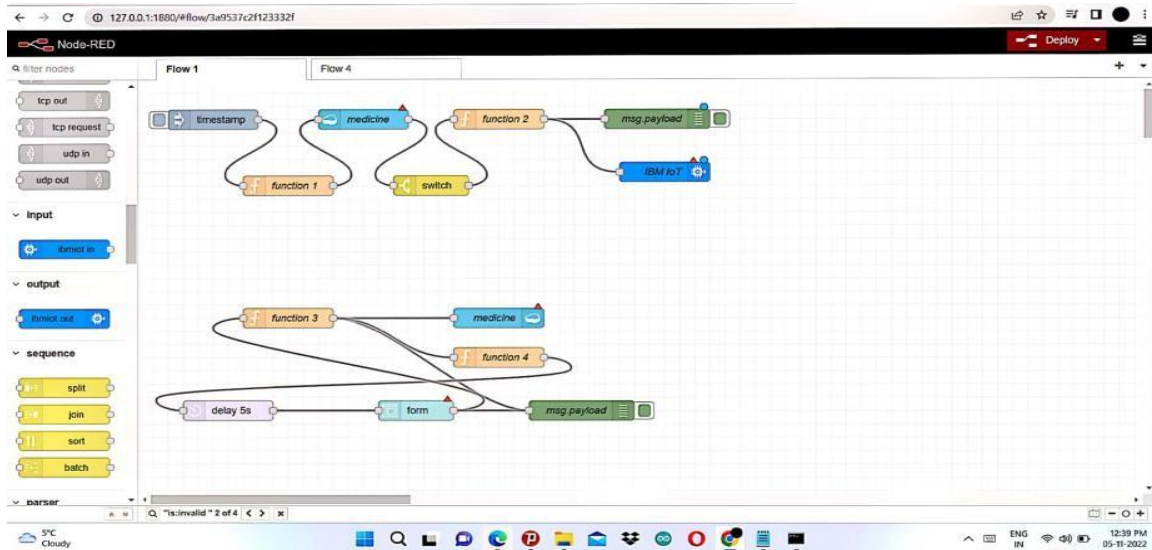


## SPRINT DELIVERY – 2

### CREATING WEB APPLICATION FOR USER

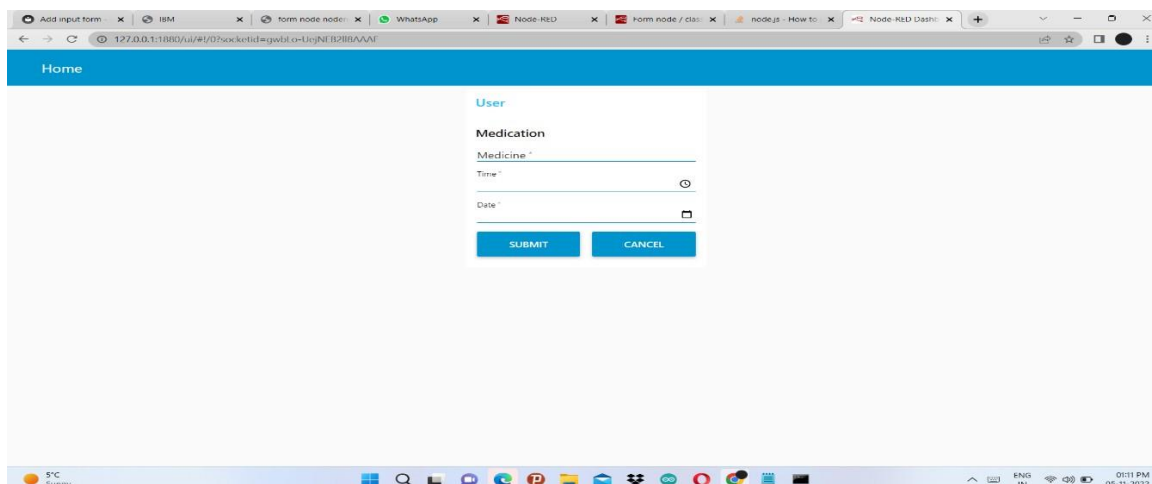
#### NODE-RED DESIGN WITH FORM NODE:



#### STEPS TO ADD INPUT FORM TO UI:

- 1) Configure the UI input form
- 2) Add a group with the name user
- 3) Add a tab with the name Home and press add
- 4) Press add
- 5) Give the UI from a name Insert a new User and add the input fields for medicine name, time and date
- 6) Add a function node and insert code and wire the nodes
- 7) Open in a new browser tab [https://\[your-host\]/ui/](https://[your-host]/ui/)
- 8) Insert required details and press submit
- 9) Now wire the function as input for the database as shown in about layout
- 10) Press Deploy.

#### RUN AND CLICK URL:



## WEB UI INTERFACE FOR USERS:

In the below image, depicts the web UI interface or web application for users to enter their medication details like name, date and time.

The screenshot shows a web browser window with a blue header bar labeled "Home". Below the header, there is a form titled "User" with a sub-header "Medication". The form contains three input fields: "Medicine" with the value "Crocin", "Time" with the value "01:25 PM", and "Date" with the value "05-11-2022". Below these fields are two buttons: "SUBMIT" and "CANCEL". The browser's address bar shows the URL "127.0.0.1:1880/ui/#/?0?socketid=gwblO-UejNEB2I8BAAAF". The Windows taskbar at the bottom shows the system clock as 01:24 PM on 05-11-2022, and the weather as 5°C Sunny.

## NOTE:

On clicking submit the details will be saved in the Cloudant database and will be accessed by function nodes used in the node red design.

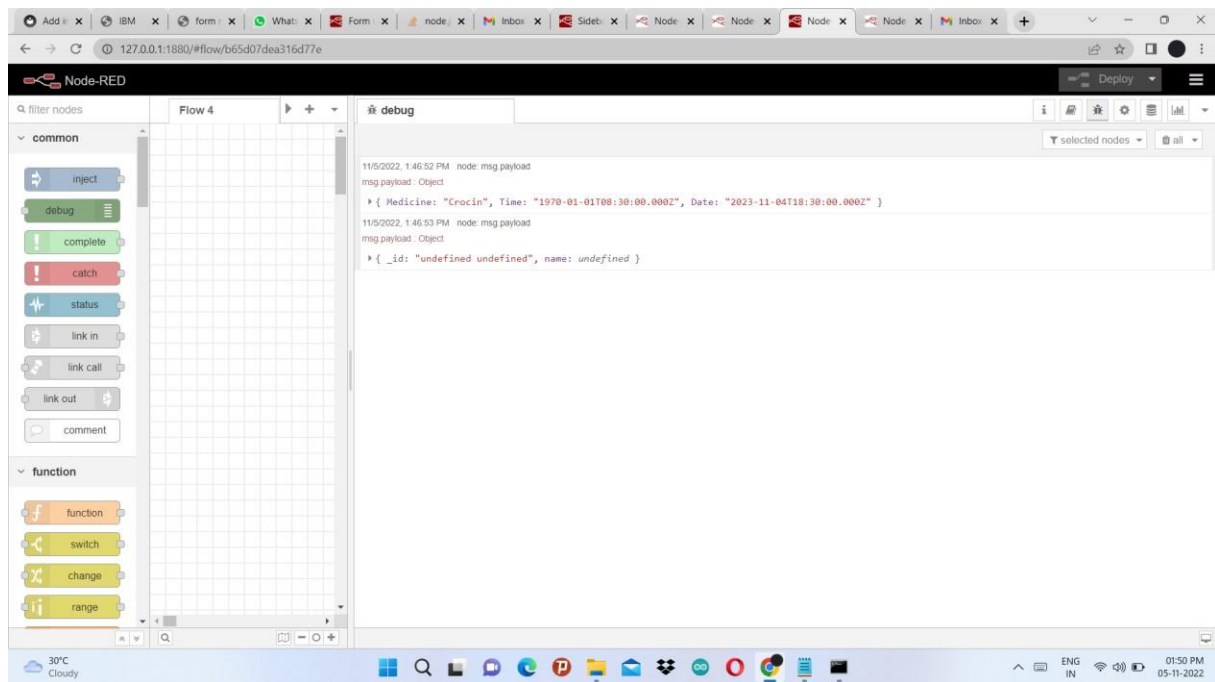
Upon comparison with time specified in function node the apt medicine name in the text format is passed on to the Text-To-Speech service (Implementation is available in sprint 1).

## DEBUG IN NODE-RED COMMAND PROMPT:

The screenshot shows the Node-RED command prompt window. It displays a series of debug messages for the "Medicine" node. The messages are as follows:

```
Nov 13:34:35 - [info] Stopped flows
Nov 13:34:35 - [info] Updated flows
Nov 13:34:35 - [info] Starting flows
Nov 13:34:35 - [info] Started flows
Nov 13:35:14 - [info] [debug:msg.payload]
{
  Medicine: 'Crocin',
  Time: '1970-01-01T08:10:00.000Z',
  Date: '2022-11-04T18:30:00.000Z'
}
Nov 13:35:14 - [info] [debug:msg.payload]
{ _id: 'undefined undefined', name: undefined }
Nov 13:41:44 - [info] Stopping flows
Nov 13:41:44 - [info] Stopped flows
Nov 13:41:44 - [info] Updated flows
Nov 13:41:44 - [info] Starting flows
Nov 13:41:44 - [info] Started flows
Nov 13:41:44 - [info] [debug:msg.payload]
{
  Medicine: 'Crocin',
  Time: '1970-01-01T08:10:00.000Z',
  Date: '2022-11-04T18:30:00.000Z'
}
Nov 13:42:40 - [info] [debug:msg.payload]
{ _id: 'undefined undefined', name: undefined }
Nov 13:43:26 - [info] Stopping flows
Nov 13:43:26 - [info] Stopped flows
Nov 13:43:26 - [info] Updated flows
Nov 13:43:26 - [info] Starting flows
Nov 13:43:26 - [info] Started flows
Nov 13:43:26 - [info] [debug:msg.payload]
{
  Medicine: 'Crocin',
  Time: '1970-01-01T08:10:00.000Z',
  Date: '2022-11-04T18:30:00.000Z'
}
Nov 13:43:55 - [info] [debug:msg.payload]
{ _id: 'undefined undefined', name: undefined }
Nov 13:46:16 - [info] Stopping flows
Nov 13:46:16 - [info] Stopped flows
Nov 13:46:16 - [info] Updated flows
Nov 13:46:16 - [info] Starting flows
Nov 13:46:16 - [info] Started flows
Nov 13:46:16 - [info] [debug:msg.payload]
{
  Medicine: 'Crocin',
  Time: '1970-01-01T08:10:00.000Z',
  Date: '2023-11-04T18:30:00.000Z'
}
```

## DEBUG IN NODE-RED PLATFORM:



### NOTE:

The above image indicates that the details has been saved in the database and readily available for accessibility through the node-red service.