

## PROJECT DEVELOPMENT PHASE

### DELIVERY OF SPRINT 2

Date	06 November 2022
Team ID	PNT2022TMID27573
Project Name	Project – Personal Assistance for senior citizens who are self-reliant

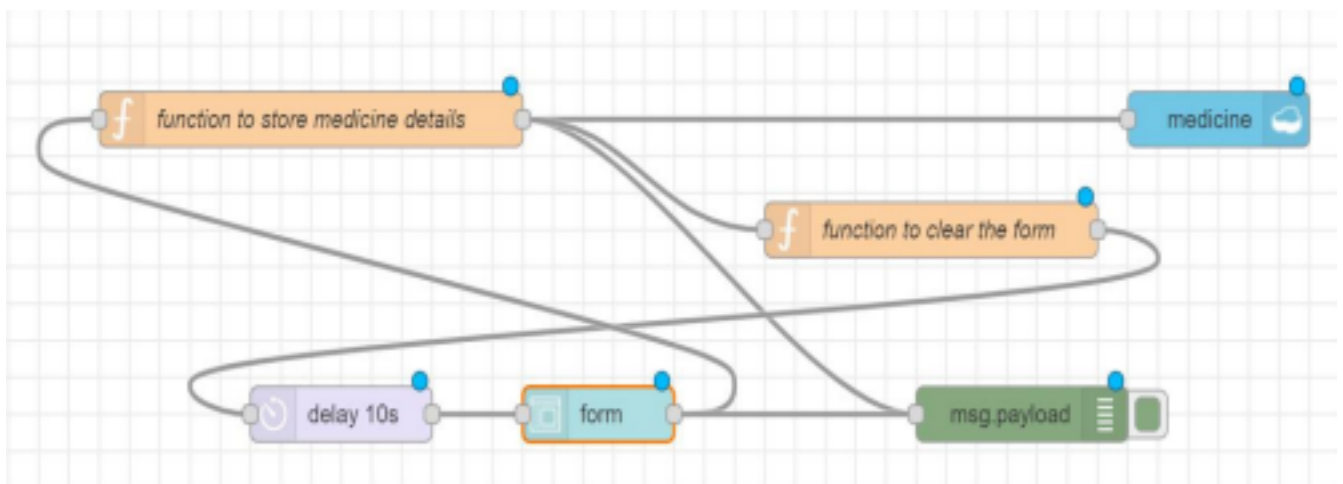
### SPRINT II: Development of Web User Interface in NodeRED service of IBM

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	<b>REGISTRATION:</b> Creation of IBM services like NODE RED, Cloudant DB and design of IOT system	USN-1	As a user, I should login into my IBM Cloud account	2	High	Vincy Veronica A, Preethi R
Sprint-2	<b>WEB UI:</b> Creating web UI using node red and connect it to IBM Cloudant DB	USN-2	As a user, I should be able to feed the medicine name and intake time in the web UI	2	High	Maria Anisha, Jane Ruffina Mary
Sprint-3	<b>SOFTWARE IMPLEMENTATION:</b> Developing Python code to retrieve data from cloudant db to send that data to IOT device at appropriate time	USN-3	As a user, I should be able to send the medicine name to the IOT device at the scheduled time	2	High	Preethi R, Maria Anisha
Sprint-4	<b>HARDWARE IMPLEMENTATION:</b> Converting data received from cloud as voice using IBM text to speech service	USN-4	As a user, I must be able to hear the medicine name which is to be taken at the appropriate time	2	High	Vincy Veronica, Jane Ruffina Mary

#### Objectives:

- To create a form UI in Node-Red platform to enter the medicine name and time of intake.
- To send the medicine name at the scheduled time.
- Complete Web UI flow
- Deploying Web UI

i)Creating a form in Node-Red to enter medicine details:



Name

Setup On Start **On Message** On Stop

```
1 var d=msg.payload.date
2 var t=msg.payload.time
3 msg.payload={
4   "_id":d+" "+t,
5   "name": msg.payload.name,
6 }
7 return msg;
```

Name

Setup On Start **On Message** On Stop

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

## Medicine Reminder

Enter medicine name \*

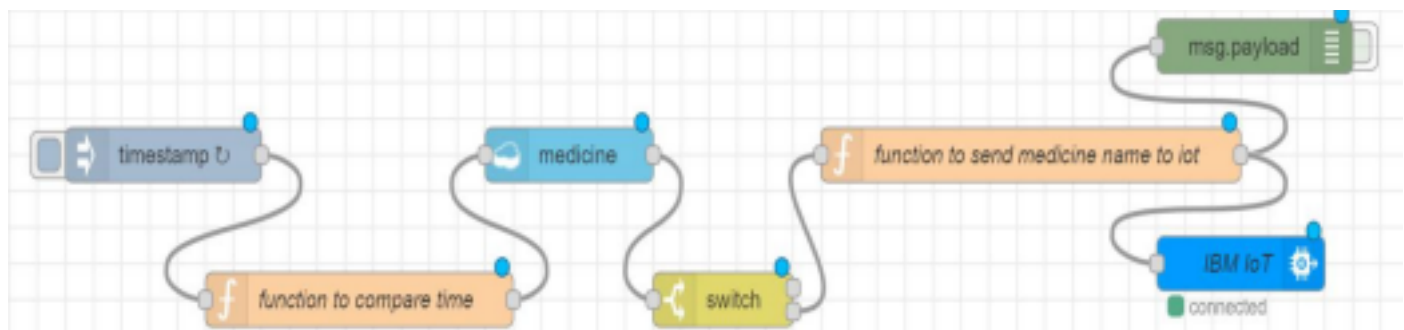
Time(HH:MM) \*

Date(YYYY-MM-DD) \*

SUBMIT

CANCEL

## ii) Sending the medicine name at the appropriate time:



Name

function to compare time

Setup

On Start

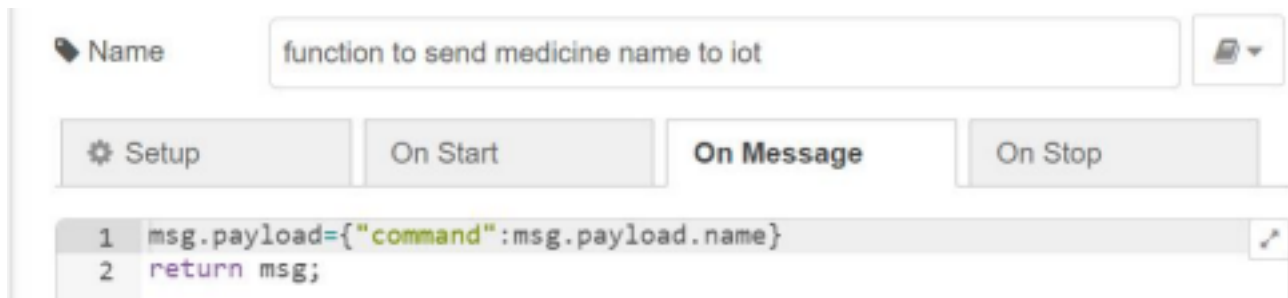
On Message

On Stop

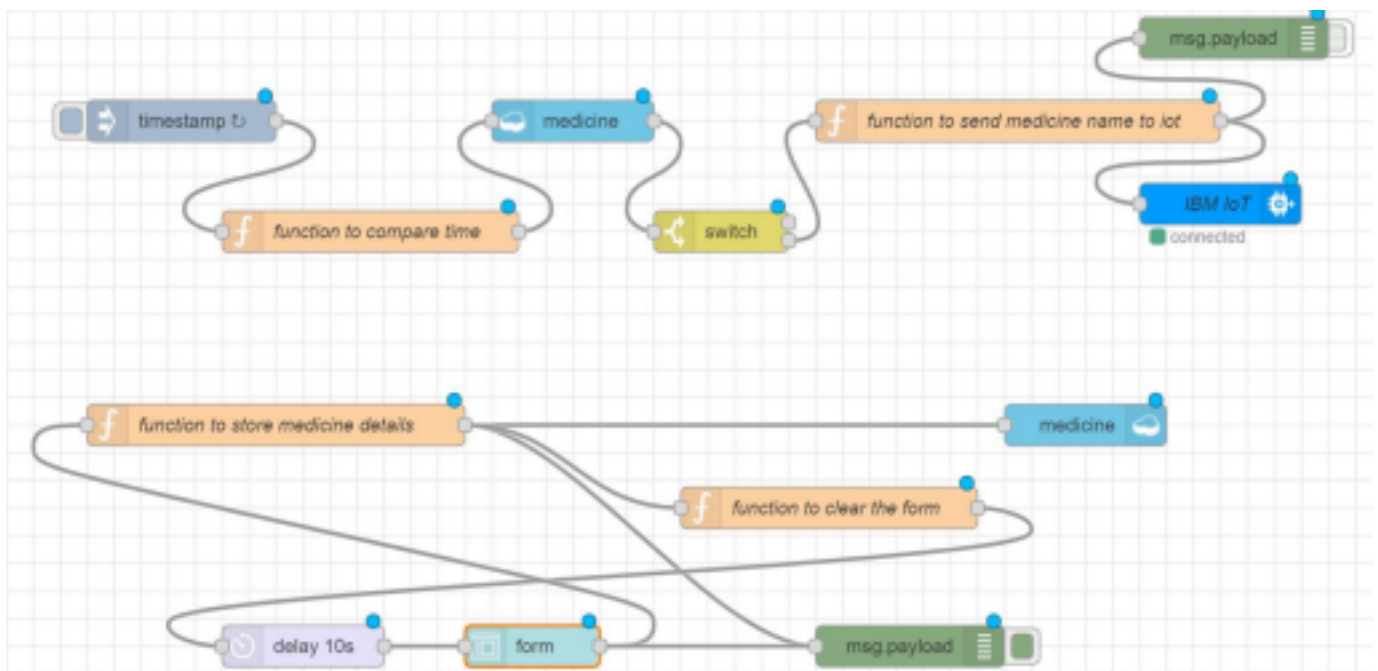
```

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;

```



### iii) Complete Web UI flow:



### iv) Deploying Web UI:

```
msg.payload : Object
  { command: "Dolo-650" }

11/11/2022, 8:17:03 AM node: 3deb0adea99203f0
msg : string[18]
"couch returned 404"

11/11/2022, 8:17:11 AM node: 79156875bd5e0858
msg.payload : Object
  { name: "Cetirizine", time: "20:18",
    date: "2022-11-11" }

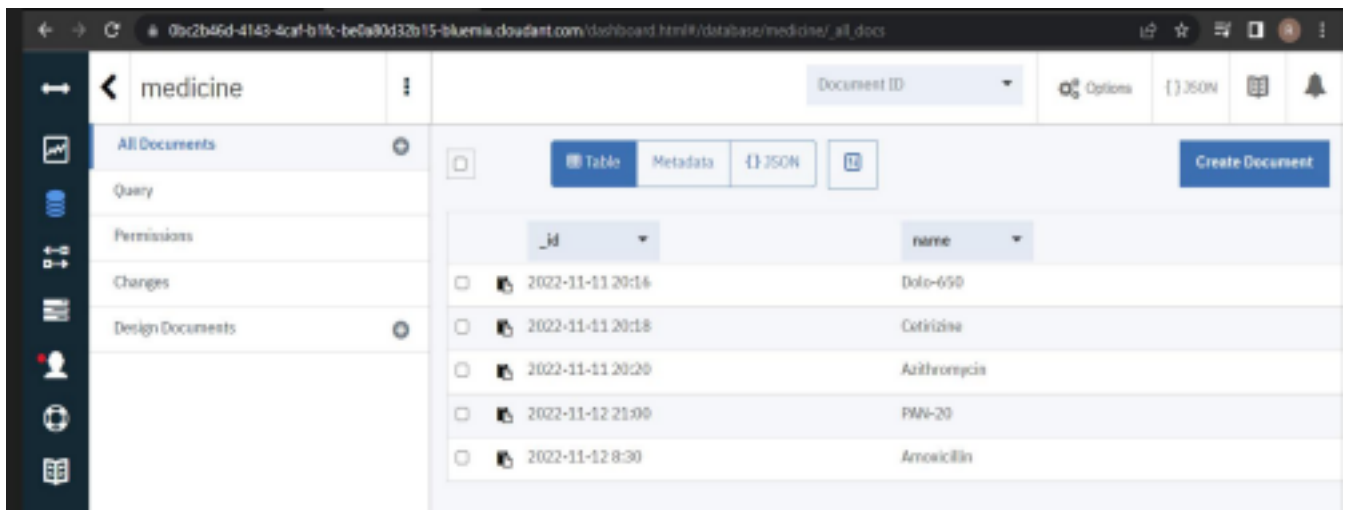
11/11/2022, 8:17:11 AM node: 79156875bd5e0858
msg.payload : Object
  { _id: "2022-11-11 20:18", name:
    "Cetirizine" }

11/11/2022, 8:18:03 AM node: 93e43a4beff1447f
msg.payload : Object
  { command: "Cetirizine" }

11/11/2022, 8:18:43 AM node: 79156875bd5e0858
msg.payload : Object
  { name: "Azithromycin", time:
    "20:20", date: "2022-11-11" }

11/11/2022, 8:18:44 AM node: 79156875bd5e0858
msg.payload : Object
  { _id: "2022-11-11 20:20", name:
    "Azithromycin" }
```

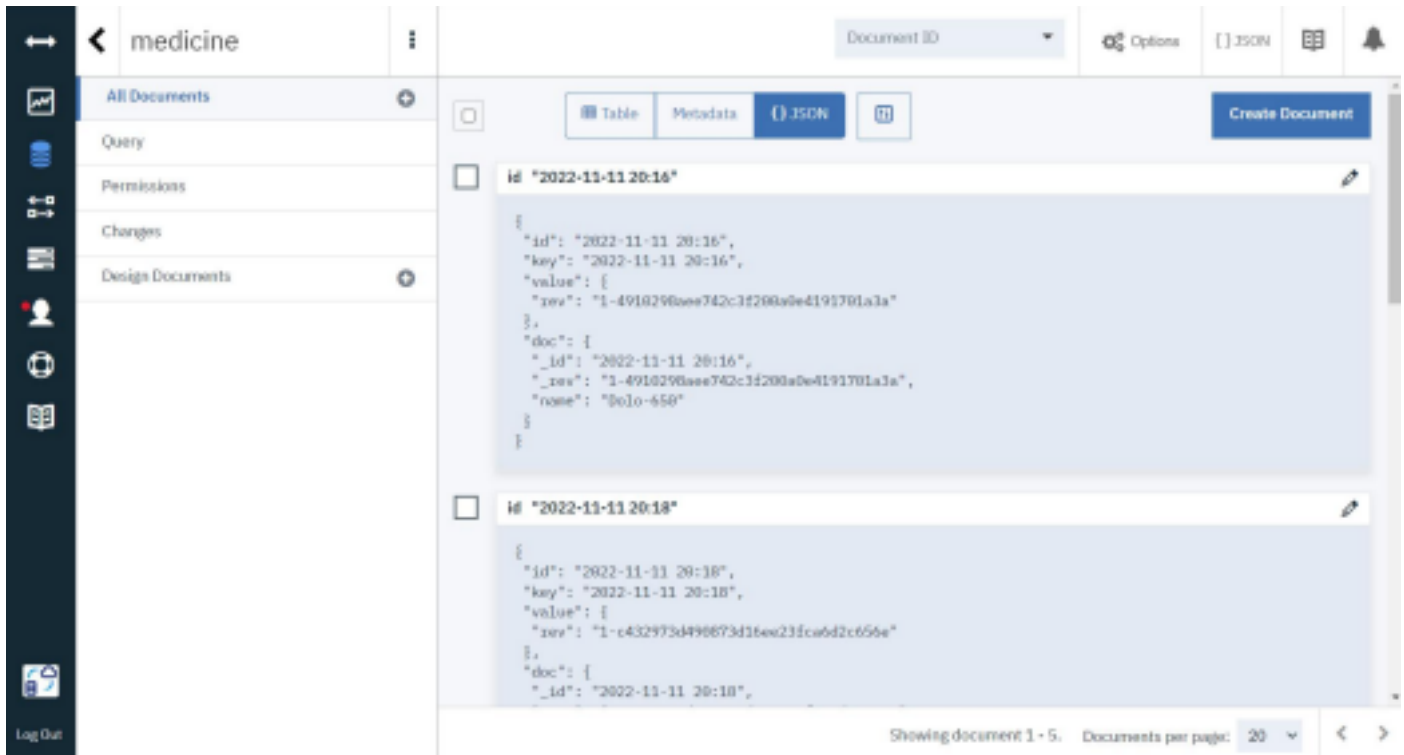
The following image shows the details of the medicine which is stored in the cloudant dbunderthemedicine database.



The screenshot shows the Cloudant dashboard for a database named 'medicine'. The interface includes a sidebar with navigation options like 'All Documents', 'Query', 'Permissions', 'Changes', and 'Design Documents'. The main area displays a table of documents. The table has two columns: '\_id' and 'name'. The documents listed are:

_id	name
2022-11-11 20:16	Dolo-650
2022-11-11 20:18	Cetirizine
2022-11-11 20:20	Azithromycin
2022-11-12 21:00	PAN-20
2022-11-12 8:30	Amoxicillin

The following image shows the details of the medicine in the medicine database in JSON format



The medicine reminder operation is depicted below:

- ✓ The medicine name Cetirizine and it's intake time 20:18 and date 11.11.2022 is stored in the database.

```
11/11/2022, 8:17:11 AM node: 79156875bd5e0858
msg.payload: Object
  {
    name: "Cetirizine",
    time: "20:18",
    date: "2022-11-11"
  }

11/11/2022, 8:17:11 AM node: 79156875bd5e0858
msg.payload: Object
  {
    _id: "2022-11-11 20:18",
    name: "Cetirizine"
  }
```

- ✓ When it's time to take the medicine, the medicine name is sent as a command to the IoT device.

```
11/11/2022, 8:17:11 AM node: 79156875bd5e0858
msg.payload: Object
  {
    _id: "2022-11-11 20:18",
    name: "Cetirizine"
  }

11/11/2022, 8:18:03 AM node: 93e43a4beff1447f
msg.payload: Object
  {
    command: "Cetirizine"
  }
```