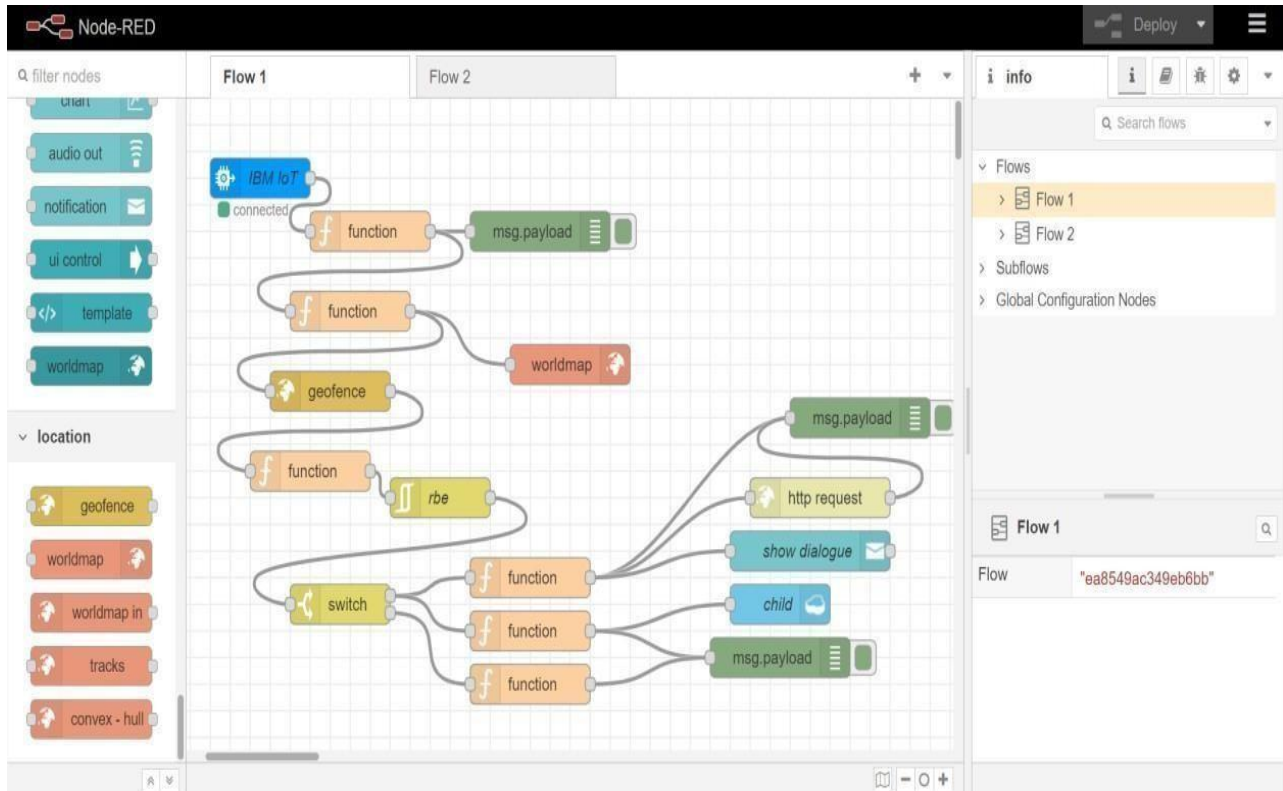


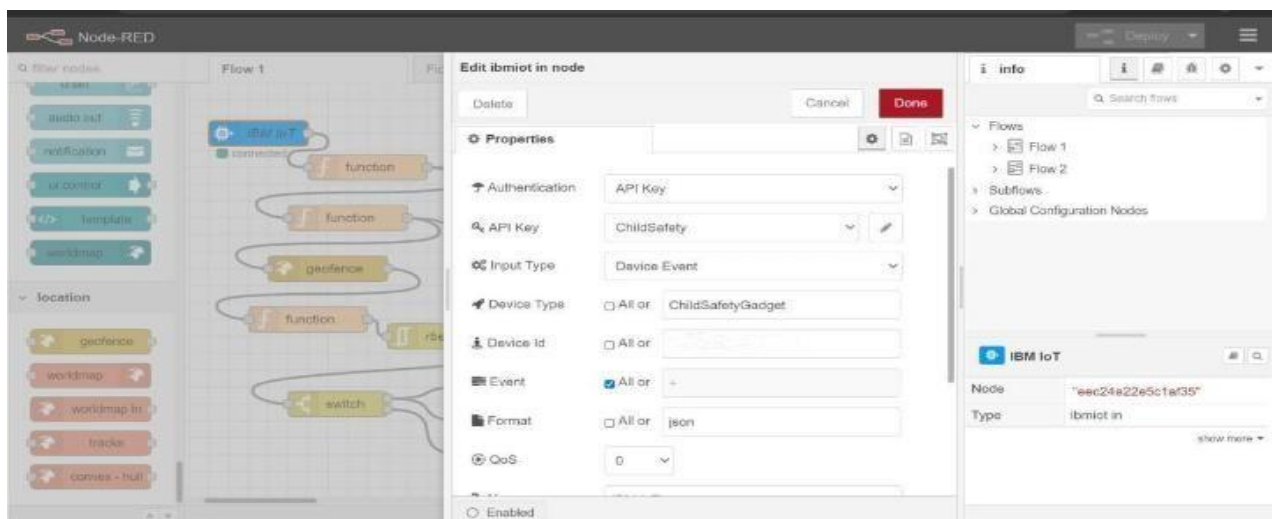
# Project Development Phase SPRINT-3

## NODE RED-Cloudant DB Communication:

### Step 1:Connections



### Step 2: Code in nodes



### Step 3:Edit function mode(code)

The screenshot shows the Node-RED interface with the 'Edit function node' dialog open. The dialog has a 'Properties' tab and a 'Code' tab. The 'Code' tab is active, showing the following JavaScript code:

```
1 var name = msg.payload.name
2 var lat = msg.payload.lat
3 var lon = msg.payload.lon
4 global.set('latitude', lat)
5 global.set('longitude', lon)
6 global.set('name', name)
7 return msg;
```

The 'Info' panel on the right shows the node's details:

- Node: "76ec77ccd270c2c7"
- Type: function

### Step 4:Changing code

The screenshot shows the Node-RED interface with the 'Edit function node' dialog open. The dialog has a 'Properties' tab and a 'Code' tab. The 'Code' tab is active, showing the following JavaScript code:

```
1 msg.payload={
2   'name':global.get('name'),
3   'lat':global.get('latitude'),
4   'lon': global.get('longitude')
5 }
6 return msg;
```

The 'Info' panel on the right shows the node's details:

- Node: "77b707bf262aa6f4"
- Type: function

## Step 5: Geofencing mode

The screenshot shows the Node-RED interface with the 'Edit geofence node' dialog open. The dialog has a 'Delete' button, 'Cancel', and 'Done' buttons. The 'Properties' tab is active, showing a map with a purple geofence circle. The map includes labels for 'St. pius pg college', 'St pius apurva's hostel', 'Vedic Systems', 'Sneha Apartments', 'Nacharam road', 'Dolphin', 'Mamas Kitchen', 'house', 'SHREE POOJA HOSPITAL', 'Bapuji Maternity and Nursing Home', and 'bapuji nagar park'. The 'Node' field in the properties panel contains the ID 'ee4695a46b4858f1' and the 'Type' is 'geofence'. The 'Enabled' checkbox is checked.

## Step 6: Worldmap node with properties

The screenshot shows the Node-RED interface with the 'Edit worldmap node' dialog open. The dialog has a 'Delete' button, 'Cancel', and 'Done' buttons. The 'Properties' tab is active, showing various settings for the worldmap node. The 'Start' section includes 'Latitude' (17.4219272), 'Longitude' (78.5400783), and 'Zoom' (16). The 'Map list' is set to '1 selected'. The 'Base map' is set to 'ESRI Satellite'. The 'Overlays' are set to '5 selected'. The 'Cluster when zoom level is less than' is set to '0 (0, off - 19)'. The 'Max age' is set to 'Remove markers after 600 seconds'. The 'User menu' is set to 'Show' and the 'Layer menu' is set to 'Hide'. The 'Lock map' and 'Lock zoom' are both set to 'False'. The 'Node' field in the properties panel contains the ID '5ed11e914b5614ea' and the 'Type' is 'worldmap'. The 'Enabled' checkbox is checked.

## Step 7: Apply the code with different node

The screenshot shows the Node-RED interface with the 'Edit function node' dialog open. The 'On Message' tab is selected, and the code is as follows:

```
1 msg.payload=msg.location.inarea
2 return msg;
```

The right sidebar shows the 'info' panel with the following details:

- Node: "3585a4ac8098085f"
- Type: function

## Step 8: Code with different node

The screenshot shows the Node-RED interface with the 'Edit switch node' dialog open. The 'Property' is set to 'msg.payload', and the conditions are 'is false' (output 1) and 'is true' (output 2). The 'checking all rules' checkbox is checked.

The right sidebar shows the 'info' panel with the following details:

- Node: "16b63596382b4cc6"
- Type: switch

## Step 9: Code with different node

The screenshot shows the Node-RED interface with the 'Edit function node' dialog open. The dialog has tabs for 'Setup', 'On Start', 'On Message', and 'On Stop'. The 'On Message' tab is selected, showing the following JavaScript code:

```
1 var d= new Date();
2 var utc = d.getTime() + (d.getTimezoneOffset() * 60000);
3 var offset = 5.5; // This is the offset for UTC+3, in your case (UTC+1
4 newDate = new Date(utc + (3600000*offset));
5 msg.payload = {
6   "message": "Entry",
7   "Time": newDate.toLocaleString(),
8   "name": global.get('name'),
9   "lat": global.get('latitude'),
10  "lon": global.get('longitude')
11 };
12 return msg;
```

The node ID is 322663c3205cc798. The node type is 'function'.

## Step 10: code with different node

The screenshot shows the Node-RED interface with the 'Edit function node' dialog open. The dialog has tabs for 'Setup', 'On Start', 'On Message', and 'On Stop'. The 'On Message' tab is selected, showing the following JavaScript code:

```
1 var d= new Date();
2 var utc = d.getTime() + (d.getTimezoneOffset() * 60000);
3 var offset = 5.5; // This is the offset for UTC+3, in your case (UTC+1
4 newDate = new Date(utc + (3600000*offset));
5 msg.payload = {
6   "message": "Entry",
7   "Time": newDate.toLocaleString(),
8   "name": global.get('name'),
9   "lat": global.get('latitude'),
10  "lon": global.get('longitude')
11 };
12 return msg;
```

The node ID is dfef78abe1ae2fd4. The node type is 'function'.



## Step 11: Code with different node

Node-RED interface showing the 'Edit function node' dialog. The code in the function node is as follows:

```
1 var d= new Date();
2 var utc = d.getTime() + (d.getTimezoneOffset() * 60000);
3 var offset = 5.5; // This is the offset for UTC+3, in your case (UTC+1
4 newDate = new Date(utc + (3600000*offset));
5 msg.payload = {
6   "message": "Entry",
7   "Time": newDate.toLocaleString(),
8   "name": global.get('name'),
9   "lat": global.get('latitude'),
10  "lon": global.get('longitude')
11 };
12 return msg;
```

The right sidebar shows the 'info' panel with the node ID "36df960d0bd4f874" and type "function".

## Step 12: Http request node->properties

Node-RED interface showing the 'Edit http request node' dialog. The properties are:

- Method: GET
- URL: <https://www.fast2sms.com/dev/wallet?authorizati>
- Payload: Ignore
- ☐ Enable secure (SSL/TLS) connection
- ☐ Use authentication
- ☐ Enable connection keep-alive
- ☐ Use proxy
- ☐ Only send non-2xx responses to Catch node
- Return: a UTF-8 string

The right sidebar shows the 'info' panel with the node ID "a43ddb0b4716e38d" and type "http request".

### Step 13: Notification node

The screenshot shows the Node-RED web interface. On the left, a palette of nodes is visible, including 'notification'. The main workspace shows a flow with several nodes: 'IBM IoT' (connected), 'function', 'function', 'geofence', 'function', and 'switch'. The 'Edit notification node' panel is open on the right. It contains the following fields and options:

- Layout:** OK / Cancel Dialog
- Send to all browser sessions:** ☐
- Default action label:** OK
- Secondary action label:** (optional label for Cancel button)
- Accept raw HTML/JavaScript input in msg.payload to format popup:** ☐
- Class:** [msg.className]
- Topic:** [msg.topic]
- Name:** show dialogue
- Enabled:** ☐

The right sidebar shows the 'info' panel with a search bar and a list of flows. The 'show dialogue' node is highlighted, showing its Node ID as '388bcb2648d1348b' and Type as 'ui\_toast'.

### Step 14: Cloudant node

The screenshot shows the Node-RED web interface. The main workspace shows the same flow as in Step 13. The 'Edit cloudant out node' panel is open on the right. It contains the following fields and options:

- Service:** Cloudant-pn-99850
- Database:** noderedapqzh20221108
- Operation:** insert
- Only store msg.payload object?:** ☐
- Name:** child
- Enabled:** ☐

The right sidebar shows the 'info' panel with a search bar and a list of flows. The 'child' node is highlighted, showing its Node ID as '7f199a311bbfac36' and Type as 'cloudant out'.