

**IOT BASED SAFETY
GADGET FOR CHILD
SAFETY MONITORING
AND NOTIFICATION**



IBM NALAIYA THIRAN PROJECT

REPORT SUBMITTED BY

TEAM ID : PNT2022TMID30986

KOWSHIK P	(620819104052)
PRAGATHEESHVARAN P	(620819104075)
PRASANTH R	(620819104077)
SURYAPRAKASH M	(620819104107)

INDUSTRY MENTOR

(BARADWAJ)

FACULTY MENTOR

(KRISHNAKUMAR R)



Table of Contents

1. INTRODUCTION	3
Project Overview	3
Purpose.....	3
2. LITERATURE SURVEY.....	4
Existing Problem	4
References.....	5
Problem Statement Definition	5
3. IDEATION & PROPOSED SOLUTION.....	6
Empathy map canvas	6
Ideation & Brainstorm	7
Proposed Solution.....	9
Problem Solution fit.....	9
4. REQUIREMENT ANALYSIS.....	11
Functional Requirement.....	11
Non-Functional Requirements	13
5. PROJECT DESIGN	15
Data Flow Diagrams	15
Solution & Technical Architecture.....	16
User Stories	17
6. PROJECT PLANNING & SCHEDULING	18
Sprint Planning & Estimation	18
7. CODING AND SOLUTIONING	20
Feature 1 and 2	20
8. RESULTS	35
OUTPUT.....	35
9. ADVANTAGES AND DISADVANTAGES	41
10. CONCLUSION.....	41
11. FUTURE SCOPE	41
12. APPENDIX.....	42

Source Code GitHub & Project demo link

1.INTRODUCTION

1.1 PROJECT OVERVIEW

The overall percentage of child abuse cases filed nowadays in the world is about 80%, out of which 74% are girl children and the rest are boys. For every 40 seconds, a child goes missing in this world. Children are the backbone of one's nation, if the future of children was affected, it would impact the entire growth of that nation. These innocent children are not responsible for what happens to them. So, parents are responsible for taking care of their own children. But, due to economic condition and aims to focus on their child's future and career, parents are forced to crave for money. Hence, it becomes difficult to cling on to their children all the time. In our system, we provide an environment where this problem can be resolved in an efficient manner. Our child tracker helps the parents in continuously monitoring the child's location. They can simply leave their children in school or parks and create a geofence around the particular location. By continuously checking the child's location notifications will be generated if the child crosses the geofence. Notifications will be sent according to the child's location to their parents or caretakers. The entire location data will be stored in the database

1.2 PURPOSE

Basically, children cannot complain about abuse cases which they face in their daily life to their parents. They can't even realize what actually happens to them at their age. It is also difficult for parents to identify their children are being abused. The main aim of this project is to prevent children before being attacked, by continuously monitoring the child's location. An autonomous real-time monitoring system is necessary for every child out there. In this system, the collected values from IOT device like the location value from GPS are used to detect the status of the child and alerts the respective guardians using GSM .

2.LITERATURE SURVEY

2.1 EXISTING PROBLEM

a. Gator Smart Watch

Gator, a kid's smartwatch from Gator Group Co. It comes with a SIM card and the free app is available on Play Store and Apple App Store [19]. Gator supports calling features up to 13 different numbers, enables two-way voice messages from the app and watch. The location tracking is based on GPS tracking when children are outdoors and Wi-Fi tracking when children are indoors. Notification will also be sent to parents when children leave the geofences. Pedometer sensor is included and the SOS alarm is supported which automatically calls 3 emergency contacts when pressed for 3 seconds. Other than that, school mode is available for setting up schedules to prevent callings during the school time. Not only that, Gator is splash proofing, enables remote voice monitoring and records historical routes

b. Explora Go

Explora Go, a waterproof watch phone for children branded Explora which includes pedometer, alarm clock and stopwatch. It possesses an app available at Playstore and Appstore. With GPS and multiple services, Explora Go shows children's' location and supports the setup of safety zones. Meanwhile, it contains a SIM card and acts like a phone enables voice calls from 10 pre-saved contacts. Similar to a phone, Explora Go can send and receive text messages, emojis, images and voice messages. It is also equipped with the SOS button that children can press to notify emergency contacts of their location. Beyond that, Wi-Fi and Bluetooth are available in Explora Go. It also supports the schedule function in which school schedules can be specified during which watch will only display time and make emergency calls.

2.2 REFERENCES

Title: Child Safety Wearable Device Using Raspberry Pi. Authors: Arun Francis G, Janani I, Kavya S and Ramiyadevi K.
Year: 2020

Title: Child Safety Monitoring System Based on IoT . Authors : D. Ezhilarasi, N. Senthamilarasi Bharathi and R.B. Sangavi
Year: 2019

Title: Smart Children Safety Using Wearable Device .Authors: Dr. R. Nagaraja and P. Elamathi .
Year:2019

Title: Smart IOT Device for Child Safety and Tracking. Authors: E Kusuma Kumari, K N H Srinivas, M Nandini Priyanka, S Murugan and T D S Sarveswararao .
Year:2019

2.3 PROBLEM STATEMENT DEFINITION:

The main aim of this project is to prevent children before being attacked, by continuously monitoring the child's location. An autonomous real-time monitoring system is necessary for every child out there. In this system, the collected values from IOT device like the location value from GPS are used to detect the status of the child and alerts the respective guardians using GSM .

3.IDEATION & PROPOSED SOLUTION

3.1 EMPATHY MAP CANVAS

GAIN:

- 1.We can save the child from disappearing
- 2.2.Parents will be without fear of their child safety

Parents will feel that their child is safe

Parents will not worry about their child's condition

They feel that their child is still in geofence

They will feel satisfied

THINK AND FEEL

They will be notified if the child cross the geo-fence

HEAR

SAY AND DO

They can Monitor the child

Can create Geo-fence

Aware about child

IoT Based Safety Gadget for Child Safety Monitoring and Notification

Child's location everytime by using an app

GPS Technology

SEE


Exact Location of child where him or her has gone or are going

PAIN:

- 1.It may be expensive
2. It should be charged every time

3.2 IDEATION AND BRAINSTORMING




Problem Statement (PS)	I am (Customer)	I'm trying to	But there are issues	Because of	Which makes me feel
PS-1	Responsible Parent	Make sure My Child's Safety	Child abuse	Lack of Intelligence	Disappointed
PS-2	Caretakers	Monitor My Child's Location	Kidnapping	Safety Awareness	Uncertain
PS-3	Parents	Protect My Child	Accidents	Lack of Intelligence	Insecure
PS-4	Responsible Parents	Get My Child's Accurate Location to Protect My Child	Accidents And Child Kidnapping	Lack of Ideas in Technology	Depressed




Brainstorm & idea prioritization


Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

10 minutes to prepare
1 hour to collaborate
2-8 people recommended


 **Before you collaborate**

A little bit of preparation goes a long way with this session. Here's what you need to do to get going.


 10 minutes

- ☐ **A Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
- ☐ **B Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.
- ☐ **C Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

[Open article](#)


 **Define your problem statement**

What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.

 5 minutes

PROBLEM

To develop child tracker to monitor the child location continuously and send the notifications to parents or caretakers if the child crosses the geo-fence and to store the entire location data in the database.



Key rules of brainstorming

To run an smooth and productive session

Key rules of brainstorming



Stay in topic.



Encourage wild



Defer



ideas.



judgment.



Listen to others.

Go for volume.

If possible, be visual.

2

Brainstorm

Write down any ideas that come to mind that address your problem statement.

🕒 10 minutes

TIP



You can select a sticky note and hit the pencil [switch to sketch] icon to start drawing!

Person 1

Development of Child Tracker

Creation of Mobile Application

Tracking the Child's Location Continuously

Integration with Shoe

Sending Notification in case of Emergency

Storing the child location in the Database

Person 2

Creation of Web based Application

Tracking the Child's Location through API's

RFID based Additional Tracking

Integration with Mobile as PWA

Notifying the Parents in case of Emergency

Person 3

Brilliant UI/ UX Design for Web Application

Notifying the Parents in case of Emergency

Integration with Cloud Services

Integration with Smart Watches

Monitoring the Child's Movements

Person 4

RFID based Tracking

Notifying the Police and Caretakers in case of Emergency

GPS Integrated Watch

GPS Integrated Shoe

GPS Integrated Bag

3.3 PROPOSED SOLUTION:

Project team shall fill the following information in proposed solution template.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Children are facing a lot of crimes nowadays in society such as kidnapping , Accidents and Child abuse.
2.	Idea / Solution description	The idea is to develop an IOT-based safety gadget to continuously monitor the children and ensure their safety and send the notification to the parents or caretakers.
3.	Novelty / Uniqueness	To create a device that is easily carryable, wearable, comfortable, cost-effective and safe for children touse.
4.	Social Impact / Customer Satisfaction	The user will be able to track their children throughout the day. if any emergency ,they will be intimated about it. The user will get the exact information about their child.
5.	Business Model (Revenue Model)	Portable, cost-effective, comfortable and small in size. To ensure that the device is safe for children to use.
6.	Scalability of the Solution	Reliable and cost-effective.

3.4 PROBLEM SOLUTION FIT :

Define CS, fit into CC	1. CUSTOMER SEGMENT(S) CS <p>Our Customers are mainly parents who are working and do not have enough time to take care of their children. Such parents are not provided with availability at anytime to look after their children . If the case so they are in need of something to make their children under the surveillance of them.</p>	6. CUSTOMER CONSTRAINTS CC <p>The constraints our customers facing are such connectivity issues or may be the protocols being used for communication. There may be chances of issues arised due to technical diseficiencies. Giving a second thought , price to be afforded for buying the developed solution kit might be the one they could not afford.</p>	5. AVAILABLE SOLUTIONS AS <p>Of course the solutions are available readily in the market such as angel monitoring system , Child GPS Tracking System , Child Safety GSM Kit , etc One such constraint the customers facing are cost and inefficiencies in the working once purchased.</p>	Explore AS, different
	2. JOBS-TO-BE-DONE / PROBLEMS CC <p>To enhance the operating condition of the developed solution the way it is not supposed to deal with any fault at any point of time so that the child safety can be highly ensured . To ensure the parents that their surveillance on their children can never be taken off.</p>	9. PROBLEM ROOT CAUSE RC <p>Considering the origination of the problem , it occurs in the base of merely irrespsective persons that are no way relatable to the children but for the currency kind of thing and also the child abuse(mainly in case of girl children)</p>	7. BEHAVIOUR BE <p>The proposed solution always keens/tends to make the gadget work in an efficient way so that it is not supposed meet up with any further constraints . Also the solution tries to ensure that efficient functionalities are to be provided to the fullest to the customers</p>	
Focus on J&P, tap into BE, understand RC	3. TRIGGERS TR <p>The trigger which induces the customers is the one that when other working parents give a try to this and comment a positive review on this , they also want to have a try to ensure their child safety.</p>	10. YOUR SOLUTION SL <p>Our Team has highly been intending to develop an efficient solution to overcome all the flaws that the existing solutions hold back still . We are highly on demand to ensure the efficient functionalities of the developing module the way it will not fail at anytime.</p>	8. CHANNELS of BEHAVIOUR CH <p>Our proposed solution has the modes of working in both offline and Online . In case of any disconnectivities happen the gadget which has been developed might tend to work on a plan B ehich includes the backup of the failure of actual working kit .</p>	Focus on J&P, tap into BE, understand RC

4. EMOTIONS: BEFORE / AFTER

EM

Customers(Parents) are being frustrated that their children are doing safe or not before using the gadget designed . Once they start to use the developed solution they might feel free to focus on their work and also the surveillance of their children would happen with ease at any point of time .

4. REQUIRMENT ANALYSIS

4.1 FUNCTIONAL REQUIRMENT

Following are the functional requirements of the proposed solution.

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	<ul style="list-style-type: none">✓ Registration through Gmail✓ Registration through phone number
FR-2	User Confirmation	<ul style="list-style-type: none">✓ Confirmation via EmailConfirmation via OTP
FR-3	App installation	<ul style="list-style-type: none">Installation through linkInstallation through play store
FR-4	Detecting child location	<ul style="list-style-type: none">✓ Detecting the child location through an android application✓ Detecting the child location through SMS
FR-5	User Interface	<ul style="list-style-type: none">✓ User Login Form.✓ Admin Login Form.
FR-6	Database	<ul style="list-style-type: none">✓ Stored in cloud for seamless connectivity. Parents and kids link with the distance and the location values obtained from the mobile devices are stored here.✓ The values include parent id ,kid id, distance, longitude, latitude etc.

Functional Requirement		Sub Requirement
FR-7	Server	<p>It connects the database and the front end application.</p> <p>The backend server has been implemented to run as a service and is deployed in an IBM cloud instance.</p> <p>✓ The backend server has been implemented to run as a service and is deployed in an IBM cloud instance.</p> <p>✓</p> <p>✓</p>
	GPS tracking	<p>✓ The system is implemented with a GPS module, which acquires the location information of the user and stores it to the database.</p>
FR-9	API	<p>✓ The value collected is sent to the database using an API.</p>
FR-10	React JS	<p>✓ We are using react js as front end for our project.</p> <p>✓ Node JS for the back end we are using node js</p>
FR-11	GPS modules	<p>✓ It receives data directly from satellites.</p>

FR-12	Battery Life	<p>✓ If the child or parent forgets to charge the device for a whole day then also the device will work. That's why we aim to make this device last the whole day with one charge.</p> <p>✓ ✓ It should be long-lasting.</p>
FR-13	Location History	<p>✓ The location history will help to track the child's activity so that they will be updated. Location history will be there for 30 days.</p> <p>✓ For example if the child gets missing with the help of location history they can track down their child's activity and also can find their child.</p>

4.2 NON-FUNCTIONAL REQUIRMENTS

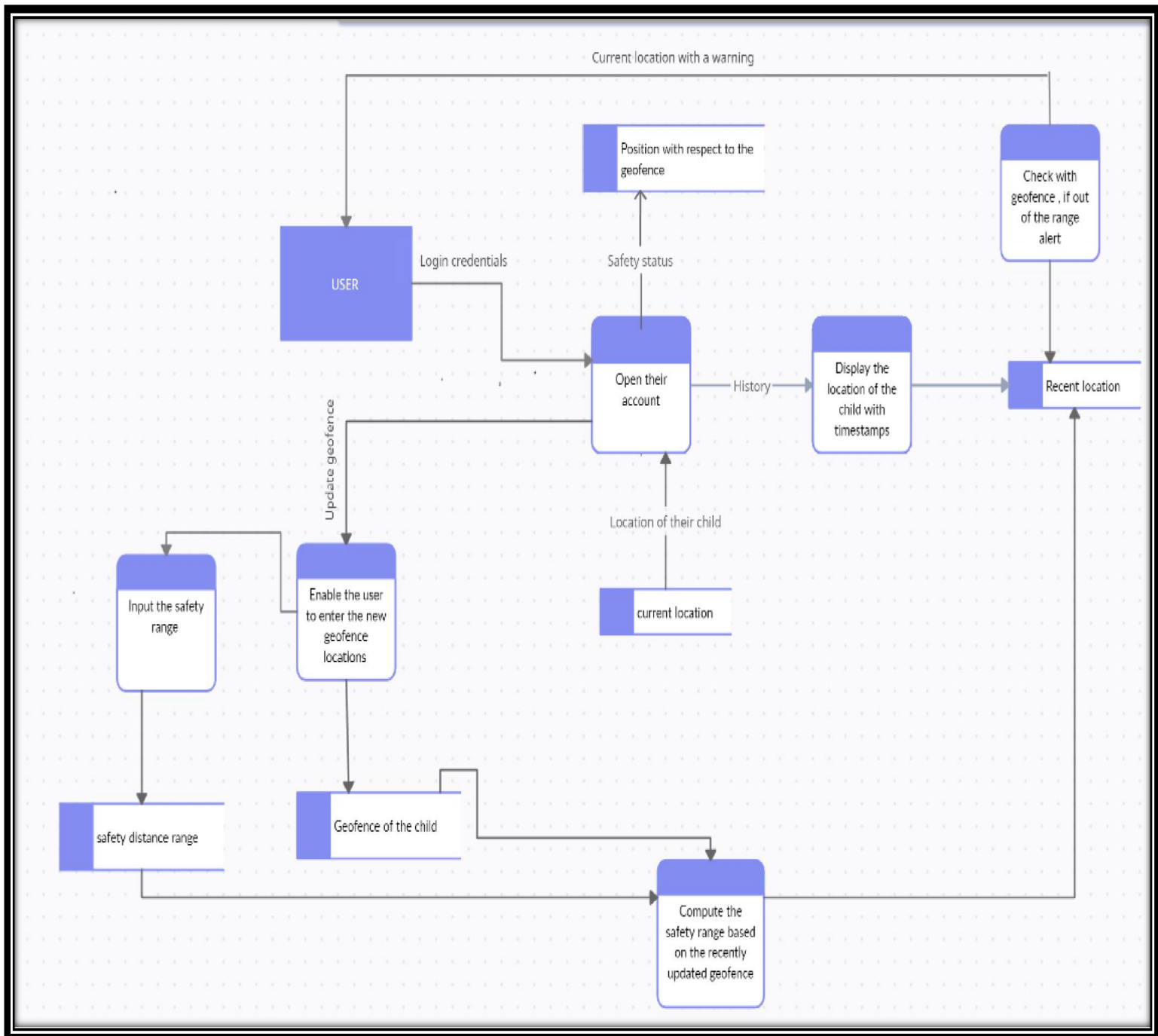
Following are the non-functional requirements of the proposed solution.

FR No.	Non-functional Requirements	Description
NFR-1	Usability	✓ Device have GSM can help to inform the parents or relatives about the current situations of the child by deliver the message immediately to save the child.
NFR-2	Security	✓ Make children parents more assure about their kid's security, we have a feature in our device called Geo-Fence. ✓ Whenever your child crosses that specific area, you will get an instant notification on your phone.
NFR-3	Reliability	Portable Easy to use Flexibility
NFR-4	Dynamicity	✓ IoT devices may have the capability to adapt dynamically and change based ontheir conditions.
NFR-5	Desirability	✓ Navigation should be made easy. ✓ The user should be able to search and find the information he needs withoutmuch hassle.

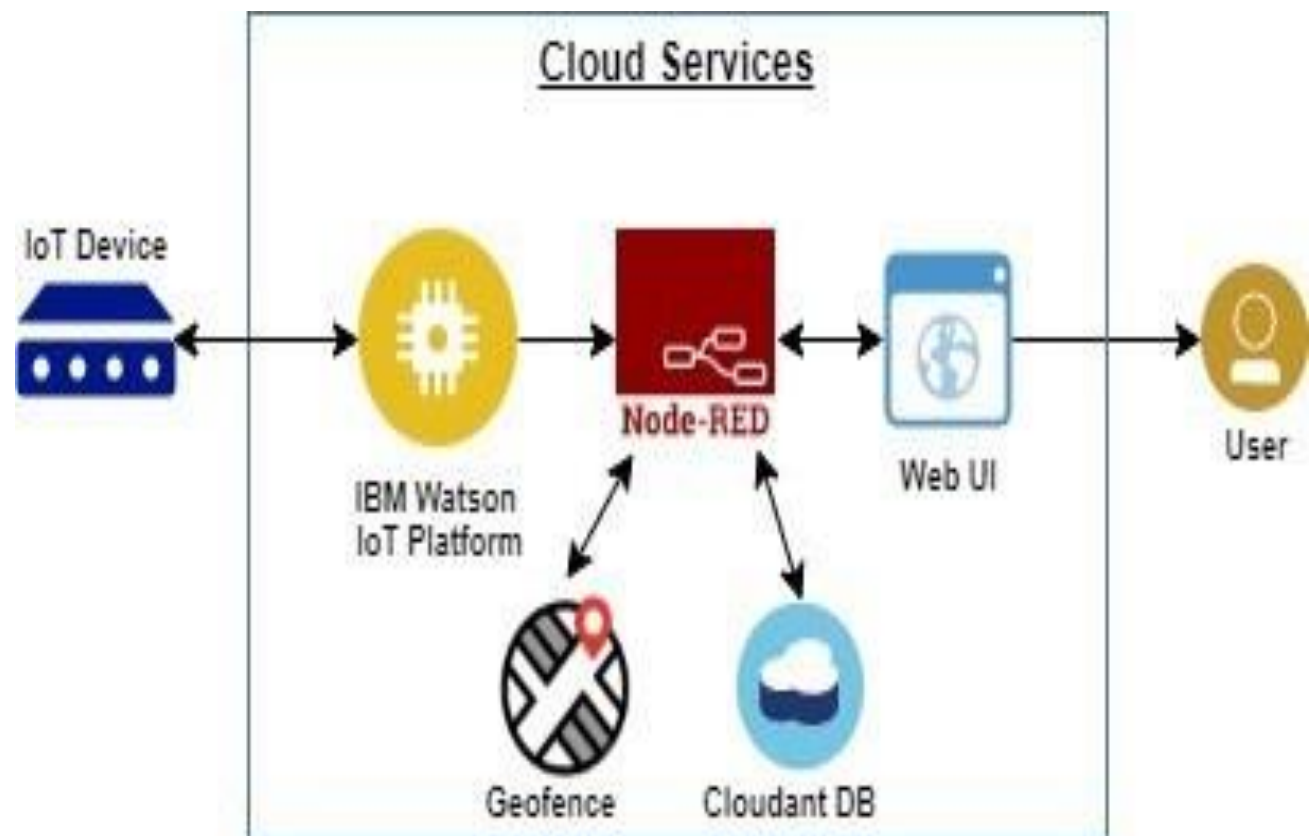
NFR-6	Performance	<p>✓ Create a Child tracker which helps the parents with continuously monitoring the child's location.</p> <p>The notification will be sent according to the child's location to their parents or caretakers.</p> <p>The entire location data will be stored in the database</p>
NFR-7	Availability	<p>Track your child even in a crowd Get travel details of kids at any meKnow the current location</p>
NFR-8	Scalability	<p>Gadget ensures the safety and tracking of the children.</p> <p>Parents need not worry about their children.</p>
NFR-9	Valuability	<p>The system should be able to deliver promptly to the financing authority.</p> <p>In the case of non-profit organizations, the solution should be 'advancing the mission'.</p>

5.PROJECT DESIGN

5.1 DATA FLOW DIAGRAM



5.2 SOLUTION & TECHNICAL ARCHITECTURE



5.3 USER STORIES

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user) and (Web user)	Registration	USN-1	As a user, I can register my account by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered myself	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through apple account	I can register & access the dashboard with apple account Login	High	Sprint-2
	Login	USN-4	As a user, I can log into the application by entering user id & password		High	Sprint-1
	Login	USN-5	As a user, I can log into the application through google account		High	Sprint-1
Customer Care Executive	Login		As I enter I can view the working of the application and scan for any glitches and monitor the operation and check if all the users are authorized.	I can login only with my provided credentials	Medium	Sprint - 3
Administrator	Login		Maintaining and making sure the database containing the locations are secure and accurate and updated constantly.	I can login only with my provided credentials	High	Sprint - 3

6. PROJECT PLANNING AND SCHEDULING

6.1 SPRINT PLANNING AND ESTIMATION

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, and password, and confirming my password.	4	High	KOWSHIK P
Sprint-1	Confirmation Email	USN-2	As a user, I will receive a confirmation email once I have registered for the application	4	High	KOWSHIK P
Sprint-1	Authentication	USN-3	As a user, I can register for the application through Gmail and mobile app.	4	Medium	SURYA PRAKASH, PRASANTH
Sprint-1	Login	USN-4	As a user, I can log into the application by entering email & password	4	High	PRAGATHEESHVARAN , PRASANTH
Sprint-1	Dashboard	USN-5	As a user, I am able to view the functions that I can perform	4	High	KOWSHIK, PRAGATHEESVARAN
Sprint-2	Notification	USN-1	As a user, I can notify my parent and guardian in emergency situations	10	High	KOWSHIK

<i>Sprint-2</i>	<i>Store data</i>	<i>USN-2</i>	<i>As a user, I need to continuously store my location data into the database.</i>	<i>10</i>	<i>Medium</i>	<i>PRAGATHEESVARAN</i>
<i>Sprint-3</i>	<i>Communication</i>	<i>USN-3,1</i>	<i>I can communicate with my parents</i>	<i>6</i>	<i>Low</i>	<i>PRASANTH, SURYAPRAKASH</i>
<i>Sprint</i>	<i>Functional Requirement (Epic)</i>	<i>User Story Number</i>	<i>User Story / Task</i>	<i>Story Points</i>	<i>Priority</i>	<i>Team Members</i>
<i>Sprint-3</i>	<i>IoT Device – Watson communication</i>	<i>USN-1,4</i>	<i>The data from IoT device should reach IBM Cloud</i>	<i>7</i>	<i>Medium</i>	<i>KOWSHIK</i>
<i>Sprint-3</i>	<i>Node RED- Cloudant DB communication</i>	<i>USN-5,2</i>	<i>The data stored in IBM Cloud should be properly integrated with Cloudant DB</i>	<i>7</i>	<i>High</i>	<i>PRAGATHEESVARAN, SURYA PRAKASH</i>
<i>Sprint-4</i>	<i>User – WebUI interface</i>	<i>USN-1,4</i>	<i>The Web UI should get inputs from the user</i>	<i>6</i>	<i>High</i>	<i>PRAGATHEESVARAN, PRASANTH, SURYA PRAKASH</i>
<i>Sprint-4</i>	<i>Geofencing</i>	<i>USN-2,3,5</i>	<i>The geofencing of the child should be done based on the geographical coordinates</i>	<i>7</i>	<i>High</i>	<i>KOWSHIK, PRAGATHEESVARAN, PRASANTH, SURYA PRAKASH</i>

7.CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 FEATURES 1

PYTHON CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

organization="mkgfko"
deviceType="raspberrypi"
deviceId="12345"
authMethod="token"
authToken="12345678"

try:
deviceOptions={"org":organization,"type":deviceType,"id":deviceId,"authmethod":authMethod,"authtoken":
authToken}
deviceCli=ibmiotf.device.Client(deviceOptions)
except Exception as e:
    print("caught exception connecting device:%s" % str(e))
    sys.exit()

deviceCli.connect()
while True:
    #in data
    name="kowshik"
    lattitude=11.229592;
    longtitude= 78.171158;
    #out data
    #lattitude=11.664325;
    #longtitude=78.146011;

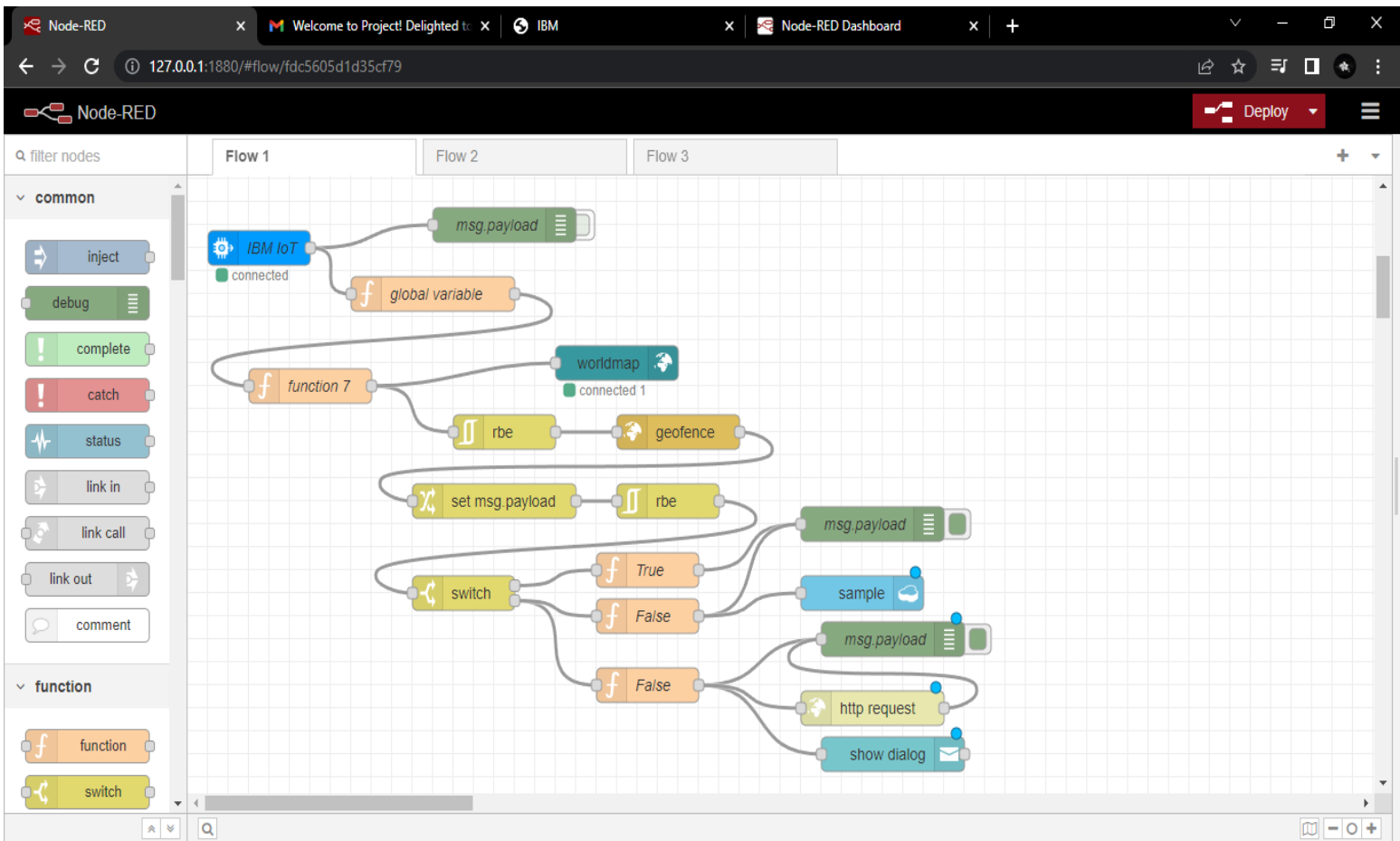
    data={'lat':lattitude,'lon':longtitude,'name':name}
    def myOnPublishCallback():
        print("publishedlattitude=%d"%lattitude,"longtitude=%d"%longtitude,"to ibm watson")

success=deviceCli.publishEvent("IotSensor","json",data,qos=0,on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTF")
```

```
time.sleep(3)  
deviceCli.disconnect()
```

7.2 FEATURES 2

NODE-RED :



GLOBAL VARIABLE NODE :

The screenshot displays the Node-RED web interface in a browser. The browser's address bar shows the URL `127.0.0.1:1880/#flow/fdc5605d1d35cf79`. The Node-RED interface includes a left sidebar with a 'filter nodes' search bar and two categories of nodes: 'common' (inject, debug, complete, catch, status, link in, link call, link out, comment) and 'function' (function, switch). The main workspace, labeled 'Flow 1', contains an 'IBM IoT' node (blue) with a 'connected' status, followed by a 'global variable' node (orange) and a 'function 7' node (orange). The 'global variable' node is selected, and its configuration panel is open on the right. The panel has tabs for 'Properties', 'Setup', 'On Start', 'On Message', and 'On Stop'. The 'Properties' tab is active, showing the node's name as 'global variable'. The 'On Message' tab is also visible, showing a JavaScript code editor with the following code:

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

At the bottom of the screen, the Windows taskbar is visible, showing the search bar, taskbar icons, and system tray information including the date and time (8:17 PM, 11/9/2022).

FUNCTION 7 NODE:

The screenshot displays the Node-RED web interface in a browser. The browser's address bar shows the URL `127.0.0.1:1880/#flow/fdc5605d1d35cf79`. The Node-RED interface includes a left sidebar with a 'filter nodes' search bar and two categories of nodes: 'common' (inject, debug, complete, catch, status, link in, link call, link out, comment) and 'function' (function, switch). The main workspace, labeled 'Flow 1', contains a flow starting with an 'IBM IoT' node (status 'connected') connected to a 'function 7' node. The 'Edit function node' dialog is open for 'function 7', showing the 'On Message' tab. The function code is as follows:

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

At the bottom of the dialog, the 'Enabled' checkbox is checked. The right sidebar shows a 'debug' tab and a 'Deploy' button. The Windows taskbar at the bottom indicates the time is 8:19 PM on 11/9/2022, with a temperature of 25°C and system language set to ENG.

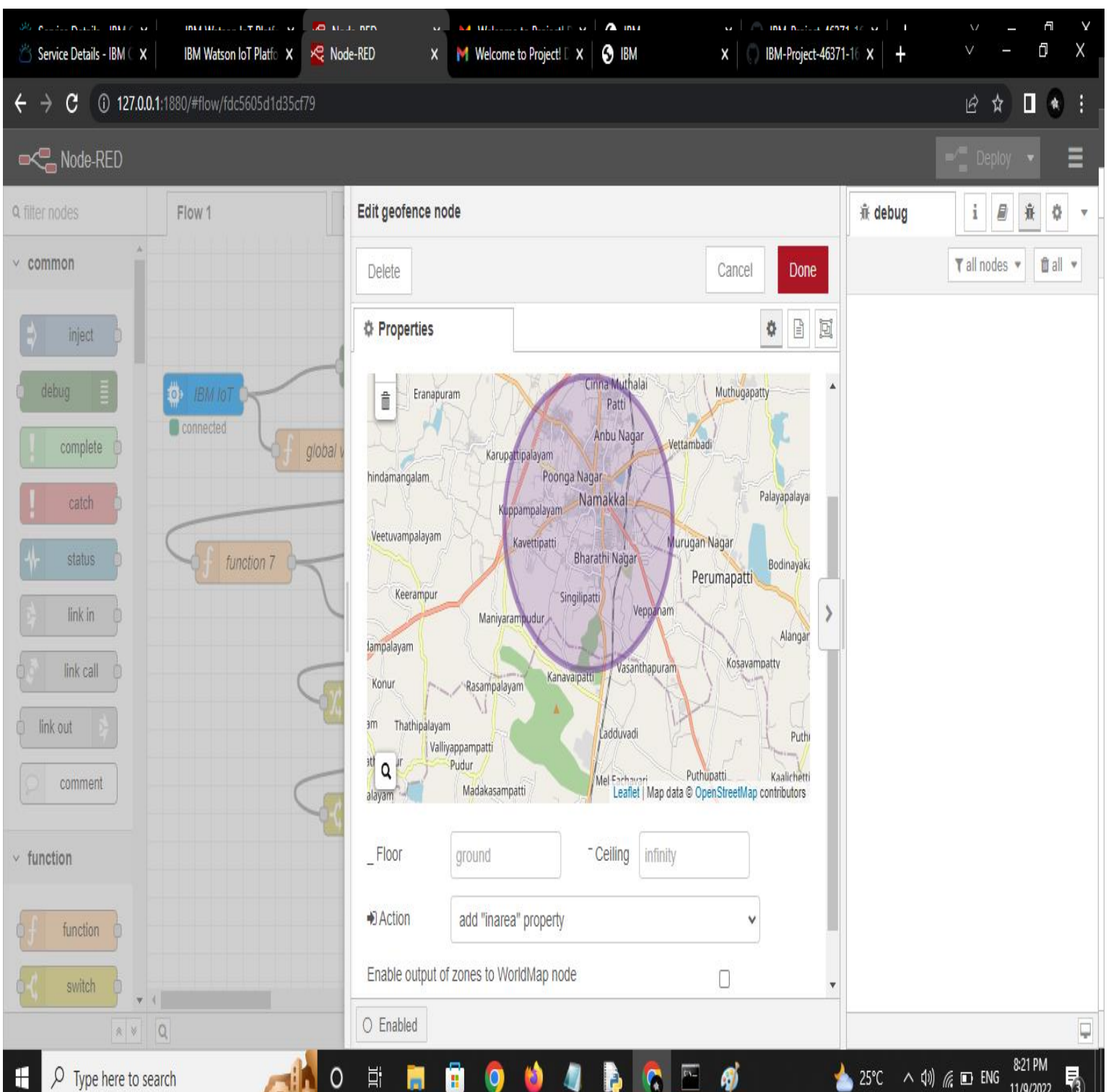
WORLD MAP NODE:

The screenshot displays the Node-RED web interface in a browser. The top navigation bar includes tabs for 'Service Details - IBM', 'IBM Watson IoT Platform', 'Node-RED', 'Welcome to Project!', 'IBM', and 'IBM-Project-46371-16'. The address bar shows the URL '127.0.0.1:1880/#flow/fdc5605d1d35cf79'. The main workspace shows a flow with nodes: 'inject', 'debug', 'complete', 'catch', 'status', 'link in', 'link call', 'link out', 'comment', 'function', and 'switch'. The 'Edit worldmap node' panel is open, showing the following configuration:

- Delete** button
- Cancel** button
- Done** button
- Properties** section:
 - Group**: [Child Tracker] Map
 - Size**: auto
 - Start**: Latitude (11.442882), Longitude (78.1792434), Zoom (9)
 - Map list**: 1 selected
 - Base map**: OpenStreetMap
 - Overlays**: 6 selected
 - Cluster when zoom level is less than**: 0 (0, off - 19)
 - Max age**: Remove markers after 600 seconds
 - User menu**: Show
 - Layer menu**: Hide
- Enabled** checkbox

The bottom status bar shows the system clock at 8:20 PM on 11/9/2022, along with temperature (25°C) and language (ENG) settings.

GEOFENCE NODE :



SWITCH NODE :

Service Details - IBM | IBM Watson IoT Platform | Node-RED | Welcome to Project! | IBM | IBM-Project-46371-16

127.0.0.1:1880/#flow/fdc5605d1d35cf79

Node-RED

Deploy

filter nodes

Flow 1

Flow 2

common

- inject
- debug
- complete
- catch
- status
- link in
- link call
- link out
- comment

function

- function
- switch

IBM IoT

connected

msg.payload

global variable

function 7

rbe

set msg.payload

switch

Edit switch node

Delete Cancel Done

Properties

Name

Property

msg.payload

is true → 1

is false → 2

+ add

checking all rules

☐ recreate message sequences

☐ Enabled

debug

all nodes

all

Type here to search

25°C

8:22 PM 11/9/2022

TRUE NODE :

The screenshot displays the Node-RED web interface in a browser. The browser's address bar shows the URL `127.0.0.1:1880/#flow/fdc5605d1d35cf79`. The interface includes a left sidebar with a 'filter nodes' search bar and two categories of nodes: 'common' (inject, debug, complete, catch, status, link in, link call, link out, comment) and 'function' (function, switch). The main workspace, titled 'Flow 1', contains an 'IBM IoT' node (labeled 'connected') and a 'function 7' node. The 'function 7' node is selected, opening the 'Edit function node' dialog. This dialog has a 'Delete' button, 'Cancel', and 'Done' buttons. Below these are tabs for 'Properties', 'Setup', 'On Start', 'On Message', and 'On Stop'. The 'Properties' tab is active, showing a 'Name' field with the value 'True'. The 'On Message' tab is also visible, showing a JavaScript code editor with the following code:

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

At the bottom of the dialog, there is an 'Enabled' checkbox. The right sidebar of the interface shows a 'debug' tab and filters for 'all nodes' and 'all'. The Windows taskbar at the bottom of the screen shows the search bar, task view, and system tray with a temperature of 25°C and the date/time 8:23 PM on 11/9/2022.

FALSE NODE :

The screenshot shows the Node-RED web interface in a browser. The browser's address bar displays the URL `127.0.0.1:1880/#flow/fdc5605d1d35cf79`. The interface includes a left sidebar with node categories (common and function), a central workspace with a flow diagram, and a right sidebar with a debug console. The 'Edit function node' panel is open, showing the configuration for a function node named 'False'. The 'Properties' section shows the name 'False'. The 'On Message' tab is selected, displaying a JavaScript code block. The code calculates the current date and time, adjusts for a 5.5-hour offset, and constructs a JSON payload with the following fields: 'message' (Exit), 'Date and Time' (formatted date), 'name' (from global context), 'lat' (from global context), and 'lon' (from global context). The payload is returned as the message. The bottom of the image shows a Windows taskbar with various application icons and a system tray displaying the temperature (25°C), time (8:24 PM), and date (11/9/2022).

Service Details - IBM x IBM Watson IoT Platform x Node-RED x Welcome to Project! x IBM x IBM-Project-46371-16 x +

Node-RED Deploy

filter nodes Flow 1

common

- inject
- debug
- complete
- catch
- status
- link in
- link call
- link out
- comment

function

- function
- switch

IBM IoT connected

function 7

Edit function node

Delete Cancel Done

Properties

Name False

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 var newDate = new Date(utc + (3600000 * offset));
5 msg.payload = {
6   "message": "Exit",
7   "Date and Time": newDate.toLocaleString(),
8   "name": global.get('name'),
9   "lat": global.get('latitude'),
10  "lon": global.get('longitude')
11 };
12 return msg;
13
```

Enabled

debug

all nodes all

Type here to search

25°C 8:24 PM 11/9/2022

FALSE NODE :

The screenshot displays the Node-RED web interface in a browser. The browser's address bar shows the URL `127.0.0.1:1880/#flow/fdc5605d1d35cf79`. The interface includes a left sidebar with node categories: 'common' (containing inject, debug, complete, catch, status, link in, link call, link out, and comment) and 'function' (containing function and switch). The main workspace, titled 'Flow 1', shows a flow starting with an 'IBM IoT' node, followed by a 'connected' node, and then a function node labeled 'function 7'. The 'Edit function node' panel is open, showing the 'Properties' tab with the 'Name' set to 'False'. Below the properties, there are tabs for 'Setup', 'On Start', 'On Message', and 'On Stop'. The 'On Message' tab is selected, displaying a JavaScript code editor with the following code:

```
1 msg.payload = {  
2   "message": "Person is not in the specified area"  
3 };  
4 return msg;
```

At the bottom of the panel, there is an 'Enabled' checkbox. The right sidebar contains a 'debug' tab and a 'Deploy' button. The Windows taskbar at the bottom shows the system clock as 8:24 PM on 11/9/2022, along with various system icons and open applications.

HTTP REQUEST NODE:

The screenshot displays the Node-RED web interface in a browser. The top navigation bar includes tabs for 'Service Details - IBM', 'IBM Watson IoT Platform', 'Node-RED', 'Welcome to Project!', 'IBM', and 'IBM-Project-46371-16'. The address bar shows the URL '127.0.0.1:1880/#flow/fdc5605d1d35cf79'. The main workspace shows 'Flow 1' with a sequence of nodes: 'global variable', 'function 7', 'rbe', 'set msg pay', and 'switch'. The 'Edit http request node' configuration panel is open on the right, showing the following settings:

- Delete:** Buttons for 'Delete', 'Cancel', and 'Done'.
- Properties:** A tab with a settings icon.
- Method:** A dropdown menu set to 'GET'.
- URL:** A text input field containing 'https://www.fast2sms.com/dev/bulkV2?authorizati'.
- Payload:** A dropdown menu set to 'Send as request body'.
- Options:** A list of checkboxes for 'Enable secure (SSL/TLS) connection', 'Use authentication', 'Enable connection keep-alive', 'Use proxy', 'Only send non-2xx responses to Catch node', and 'Disable strict HTTP parsing', all of which are currently unchecked.
- Return:** A dropdown menu set to 'a UTF-8 string'.
- Headers:** A section for adding headers, currently empty.
- Enabled:** A radio button labeled 'Enabled'.

The bottom of the screen shows a Windows taskbar with a search bar, system icons (temperature, volume, network), and the date/time '8:26 PM 11/9/2022'.

SHOW DIALOG NODE:

The screenshot displays the Node-RED web interface in a browser. The top navigation bar includes tabs for 'Service Details - IBM', 'IBM Watson IoT Platform', 'Node-RED', 'Welcome to Project!', 'IBM', and 'IBM-Project-46371-16'. The address bar shows the URL '127.0.0.1:1880/#flow/fdc5605d1d35cf79'. The main workspace shows a flow with nodes: 'global variable', 'function 7', 'rbe', 'set msg.payload', and 'switch'. The 'Edit notification node' dialog is open, featuring a 'Delete' button, 'Cancel' button, and a red 'Done' button. The 'Properties' section includes a 'Layout' dropdown set to 'OK / Cancel Dialog', a checked 'Send to all browser sessions' option, a 'Default action label' of 'OK', and a 'Secondary action label' field with the placeholder '(optional label for Cancel button)'. There is an unchecked checkbox for 'Accept raw HTML/JavaScript input in msg.payload to format popup.' Below this, the 'Class' field contains '[msg.className]', the 'Topic' field contains 'person is not in the specified area', and the 'Name' field contains 'Name'. A yellow note at the bottom of the dialog states: 'Note: checking Accept raw HTML/JavaScript can allow injection of'. The 'Enabled' checkbox is checked. The right sidebar shows a 'debug' tab and filters for 'all nodes' and 'all'. The Windows taskbar at the bottom shows the search bar, task view, and various application icons, with the system clock indicating 8:25 PM on 11/9/2022.

Service Details - IBM x IBM Watson IoT Platform x Node-RED x Welcome to Project! x IBM x IBM-Project-46371-16 x +

127.0.0.1:1880/#flow/fdc5605d1d35cf79

Node-RED Deploy

filter nodes Flow 1 Flow 2

common

inject

debug

complete

catch

status

link in

link call

link out

comment

function

function

switch

global variable

function 7

rbe

set msg.payload

switch

Edit notification node

Delete Cancel Done

Properties

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 person is not in the specified area

Name Name

Note: checking Accept raw HTML/JavaScript can allow injection of

Enabled

debug

all nodes all

Type here to search

25°C 8:25 PM 11/9/2022

SAMPLE NODE (DATABASE NODE) :

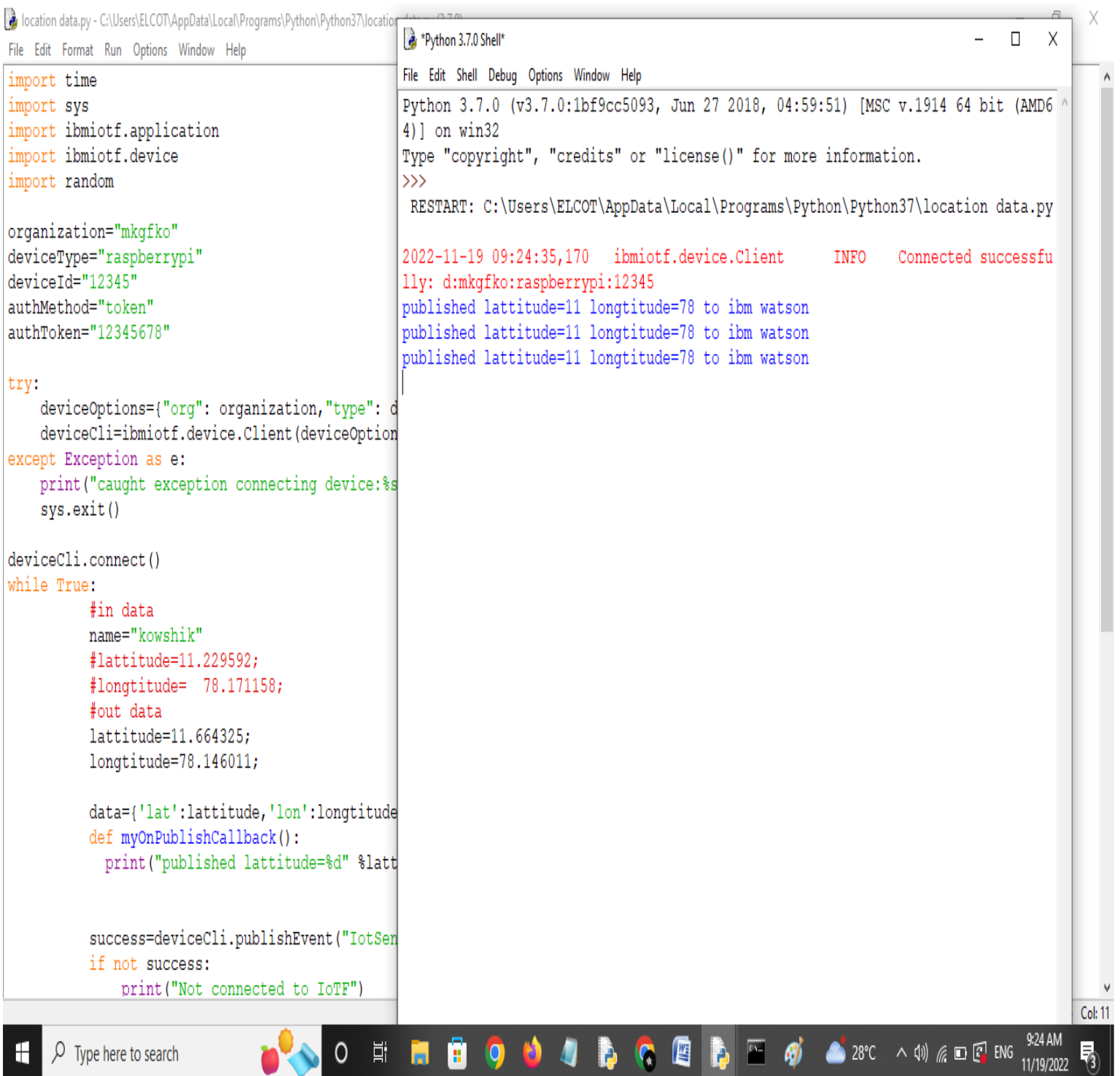
The screenshot shows the Node-RED web interface running in a browser. The browser's address bar displays the URL `127.0.0.1:1880/#flow/fdc5605d1d35cf79`. The Node-RED interface includes a left sidebar with a 'filter nodes' search bar and two categories of nodes: 'common' (inject, debug, complete, catch, status, link in, link call, link out, comment) and 'function' (function, switch). The main workspace shows 'Flow 1' with a 'global variable' node connected to a 'function 7' node, which is then connected to a 'switch' node. A 'debug' node is also present. On the right, the 'Edit cloudant node' dialog is open, showing the 'Properties' tab. The dialog has buttons for 'Delete', 'Cancel', and 'Update'. The properties include: Host (`https://apikey-v2-21m0n00eyc0gearojre4cacrhjx`), Username (`apikey-v2-21m0n00eyc0gearojre4cacrhjx8kdlbp`), Password (masked with dots), and Name (`Name`). At the bottom of the dialog, there is a status bar indicating 'Enabled' and '1 node uses this config', with a dropdown menu set to 'On all flows'. The bottom of the screen shows a Windows taskbar with a search bar, several application icons, and system status information including temperature (25°C), time (8:26 PM), and date (11/9/2022).

SET.MSG.PAYLOAD NODE

The screenshot displays the Node-RED web interface in a browser. The top navigation bar includes tabs for 'Service Details - IBM', 'IBM Watson IoT Platform', 'Node-RED', 'Welcome to Project!', 'IBM', and 'IBM-Project-46371-1'. The address bar shows the URL '127.0.0.1:1880/#flow/fdc5605d1d35cf79'. The main workspace shows 'Flow 1' with a sequence of nodes: 'IBM IoT' (connected), 'global variable', 'function 7', 'rbe', 'set msg.payload', and 'switch'. The 'set msg.payload' node is selected, and the 'Edit change node' dialog is open. The dialog has a 'Delete' button, 'Cancel' and 'Done' buttons, and a 'Properties' section. The 'Name' field is empty. The 'Rules' section shows a 'Set' rule for 'msg.payload' to the value 'msg.location.inarea', with an unchecked 'Deep copy value' option. A '+ add' button is at the bottom of the rules list. The right sidebar shows a 'debug' tab. The Windows taskbar at the bottom includes a search bar and various application icons. The system tray shows '25°C', '8:22 PM', and '11/9/2022'.

8.RESULTS

PYTHON OUTPUT :



```
location data.py - C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\location data.py (3.7.0)
File Edit Format Run Options Window Help

import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

organization="mkgfko"
deviceType="raspberrypi"
deviceId="12345"
authMethod="token"
authToken="12345678"

try:
    deviceOptions={"org": organization,"type": deviceType}
    deviceCli=ibmiotf.device.Client(deviceOptions)
except Exception as e:
    print("caught exception connecting device:%s" % e)
    sys.exit()

deviceCli.connect()
while True:
    #in data
    name="kowshik"
    #latitude=11.229592;
    #longitude= 78.171158;
    #out data
    latitude=11.664325;
    longitude=78.146011;

    data={'lat':latitude,'lon':longitude}
    def myOnPublishCallback():
        print("published latitude=%d" %latitude)

    success=deviceCli.publishEvent("IotSensorData",data,myOnPublishCallback)
    if not success:
        print("Not connected to IoT")

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help

Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
RESTART: C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\location data.py

2022-11-19 09:24:35,170 ibmiotf.device.Client INFO Connected successfully: d:mkgfko:raspberrypi:12345
published latitude=11 longitude=78 to ibm watson
published latitude=11 longitude=78 to ibm watson
published latitude=11 longitude=78 to ibm watson
```

Node-RED IN AREA OUTPUT :

location data.py - C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\location data.py (3.7.0)

File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
```

Node-RED Welcome to Project! Delighted to... IBM Node-RED Dashboard

127.0.0.1:1880/#flow/fdc5605d1d35cf79

Node-RED Deploy

filter nodes

Flow 1 Flow 2 Flow 3

common

- inject
- debug
- complete
- catch
- status
- link in
- link call
- link out
- comment

function

- function

Flow 1

msg.payload

IBM IoT

global variable

function 7

worldmap

rbe

geofence

set msg.payload

switch

True

False

False

msg.payload

sample

msg.payload

http request

show dialog

debug

all nodes all

11/8/2022, 7:44:02 PM node: msg.payload

iot-2/type/raspberrypi/id/12345/evt/lotSensor/fmt/json : msg.payload :

Object

```
{ message: "Entry", Date and Time: "11/9/2022, 9:14:01 AM", name: "kowshik", lat: 11.229592, lon: 78.171158 }
```


IN AREA WEB UI:

Node-RED x Welcome to Project! Delighted to x IBM x Node-RED Dashboard x +

127.0.0.1:1880/ui/#!/0?socketid=7VWRY6YITFU9nQz1AAAb

Child Tracker

Map



Windows taskbar: Type here to search, 11°C, 7:45 PM, 11/8/2022

Node – RED OUT AREA OUTPUT:

The screenshot displays the Node-RED web interface in a browser. The top bar shows the URL `127.0.0.1:1880/#flow/fdc5605d1d35cf79`. The interface is divided into three main sections: a left sidebar with node categories, a central workspace for building flows, and a right sidebar for debugging.

Left Sidebar: Contains two categories of nodes: **common** (inject, debug, complete, catch, status, link in, link call, link out, comment) and **function** (function, switch).

Central Workspace (Flow 1): The flow starts with an **IBM IoT** node (connected). It connects to a **msg.payload** node, then a **global variable** node, and finally a **function 7** node. The **function 7** node is connected to a **worldmap** node (connected 1). The **worldmap** node connects to an **rbe** (Rule Based Engine) node, which then connects to a **geofence** node. The **geofence** node connects to another **rbe** node, which then connects to a **set msg.payload** node. The **set msg.payload** node connects to a **switch** node. The **switch** node has three outputs: **True**, **False**, and **False**. The **True** output connects to a **sample** node, which then connects to a **msg.payload** node. The **False** outputs connect to an **http request** node and a **show dialog** node. The **http request** node connects to a **msg.payload** node.

Right Sidebar (debug): The debug console shows three messages:

```
11/8/2022, 7:48:23 PM node: msg.payload
iot-2/type/raspberrypi/id/12345/evt/lotSensor/fmt/json : msg.payload :
Object
  { message: "Person is not in the specified.." }

11/8/2022, 7:48:23 PM node: msg.payload
iot-2/type/raspberrypi/id/12345/evt/lotSensor/fmt/json : msg.payload :
Object
  { message: "Exit", Date and Time: "11/9/2022, 9:18:23 AM", name: "kowshik", lat: 12.7345, lon: 13.202 }

11/8/2022, 7:48:24 PM node: msg.payload
iot-2/type/raspberrypi/id/12345/evt/lotSensor/fmt/json : msg.payload :
string[111]

{"status_code":999,"message":"You need to complete one transaction of 100 INR or more before using API route."}
```

OUT AREA WEB UI:

Node-RED x Welcome to Project! Delighted to x IBM x Node-RED Dashboard x +

127.0.0.1:1880/ui/#/i/0?socketid=7VWRV6YITFU9nQz1AAAb

Child Tracker

Map

person is not in the specified area

[object Object]

OK

Windows Type here to search 11°C 7:46 PM 11/8/2022

The screenshot shows a web browser window with multiple tabs. The active tab is 'Node-RED Dashboard'. The address bar shows the URL '127.0.0.1:1880/ui/#/i/0?socketid=7VWRV6YITFU9nQz1AAAb'. The web application has a blue header with the title 'Child Tracker'. The main content area is a map. A white error dialog box is centered on the screen, displaying the message 'person is not in the specified area' and '[object Object]'. The dialog has an 'OK' button. The map shows various locations including Muthalaipatti, Marupattu, Semparaipudur, Namakkal, Ganga Nagar, Andavar Nagar, Kurinji Nagar, Bharathi Nagar, Kondichettipatti, Singilipatti, Veppanam, and Kanavaipatti. The Windows taskbar is visible at the bottom, showing the search bar, system tray with temperature (11°C), time (7:46 PM), and date (11/8/2022).

STORED DATA IN CLOUDANT DB:

Node-REDWelcome to Project!IBMNode-RED DashboardService Details - IBMCloudant Dashboard

bdd5f81b-d4c5-4c57-ae34-3a91d13afe81-bluemix.cloudant.com/dashboard.html#database/sample/ff932ace48ba73c9cbec51d85a0456ca

sample > ff932ace48ba73c9cbec51d85a0456ca

JSON

Save ChangesCancel

Upload AttachmentClone DocumentDelete

```
1
2  "_id": "ff932ace48ba73c9cbec51d85a0456ca",
3  "_rev": "2-bdd606a634923a0fc17f941778161bed",
4  "topic": "iot-2/type/raspberrypi/id/12345/evt/IotSensor/fmt/json",
5  "payload": {
6    | "message": "Exit",
7    | "Date and Time": "11/9/2022, 5:01:14 AM",
8    | "name": "kowshik",
9    | "lat": 12.7345,
10   | "lon": 13.202
11  },
12  "deviceId": "12345",
13  "deviceType": "raspberrypi",
14  "eventType": "IotSensor",
15  "format": "json",
16  "location": {
17    | "inarea": false
18  }
19
```

Log Out

Type here to search

Low... 9:13 PM 11/8/2022

9.ADVANTAGES AND DISADVANTAGES

ADVANTAGES:

- Gives exact location of child where he or she has gone or is going
- Quick and Reliable output
- Parents do not need to worry about condition of their childrens

DISADVANTAGES:

- Requires internet connection

10. CONCLUSION

The word Future resembles the word Children. As Dr. A.P.J Abdul Kalam's words "Youngsters are the future pillars of one's nation", today's children are tomorrow's youngsters, preserving their dreams and life for a better future is necessary. Therefore, each and every parent should take care of their own children, without letting them to fall into the dark world of abuse, which entirely ruin them physically, mentally and emotionally destroying our future. Hence, considering the importance of our future, our project makes it easy for parents to track their children and to visually monitor them on regular basis, which makes them ensure the safety of their children and reduces the rate of incidents of child abuse.

11 . FUTURE SCOPE

In our system, we automatically monitor the child in real time using Internet of Things, with the help of GPS, GSM, and Raspberry Pi. This system requires network connectivity, satellite communication, and high-speed data connection when we use web camera and GPS to live monitor. It is difficult to monitor when there occurs any hindrance to satellite communication or any network issue. There also occurs time delay in video streaming through the server. Hence in the future, these issues can be overcome by using Zigbee concept or accessing the system without internet and using high-speed server transmission.

12.APPENDIX

SOURCE CODE

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

organization="mkgfko"
deviceType="raspberrypi"
deviceId="12345"
authMethod="token"
authToken="12345678"

try:
    deviceOptions={"org":organization,"type":deviceType,"id":deviceId,"authmethod":authMethod,"authtoken":
authToken}
    deviceCli=ibmiotf.device.Client(deviceOptions)
except Exception as e:
    print("caught exception connecting device:%s" % str(e))
    sys.exit()

deviceCli.connect()
while True:
    #in data
    name="kowshik"
    lattitude=11.229592;
    longtitude= 78.171158;
    #out data
    #lattitude=11.664325;
    #longtitude=78.146011;

    data={'lat':lattitude,'lon':longtitude,'name':name}
    def myOnPublishCallback():
        print("publishedlattitude=%d"%lattitude,"longtitude=%d"%longtitude,"to ibm watson")

    success=deviceCli.publishEvent("IotSensor","json",data,qos=0,on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTf")
        time.sleep(3)
deviceCli.disconnect()
```

GITHUB LINK: <https://github.com/IBM-EPBL/IBM-Project-46371-1660746244>

DEMO LINK: <https://github.com/IBM-EPBL/IBM-Project-46371-1660746244/blob/main/Final%20Deliverables/ibm%20project%20demo%20video/IBM%20DEMO%20VIDEO.mp4>

