IoT Based Safety Gadget for Child Safety Monitoring and Notification

PROJECT REPORT

Submitted By

Team ID : PNT2022TMID21372

Team Leader : KAURI SHEETAL P (917719D038),
Team Member : HARIPPRIYADHARSHINI J (917719D027),
Team Member : HARSHAVARDHINI C (917719D029),
Team Member : HITHAYATHUN NIHMA J (917719D033).

in partial fulfillment for the award of the degree of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING.

THIAGARAJAR COLLEGE OF ENGINEERING,

MADURAI-625015.

TABLE OF CONTENTS

1. INTRODUCTION:

1.1 Project Overview:

The internet of things (IoT) refers to the set of devices and systems that stay interconnected with real-world sensors and to the internet. For years' Child safety has been under threat and it is very important to provide a technology-based solution which will help them under panic situations and monitor them using a smart gadget. It is observed that more families are now spending time on work and social duties, resulting in a lack of interaction with their children. This causes increased concerns towards their safety and whereabouts which has made keeping a track of their activities quite challenging.the proposed system must alert the parents when the child walks too far away and/or outside the geofence in their absence. In today's world, children are the most precious resources as they are imminent of the country. The parents always look forward to having their children in a safeguarded place where they can make their time without any complications. Unluckily children are threatened. The violation has been growing increasingly. The security of children is unsafe. During children's transit to and from the school,there are many miserable cases observed in the media. Due to the absence of preventative considerations children seem to be nowhere to be found and later end up in trouble. During the transit of the children to and from school,and playing outside parents are anxious. Therefore, in current days, the security issue of children is to be given supreme significance. This Child Monitoring system helps monitor or track the child and their activities .

1.2 Purpose:

The purpose of the proposed solution is to periodically track the kid's location and update the same to the parent whenever the child crosses the geofence area. The aim of this system is to develop a Tracking system which provides the details of entry and exit of the child from geofence. The proposed system provides a facility to track the exact location of the child using the smart gadget thereby sending notification to the parent when the child exits the geofence. This system helps the parents to track their child which inturn guarantees much needed peace of mind. This system plays an important role. It tracks whether the children are safe. Some prominent features of this system are Geo-fencing, Long battery life, Real-Time Tracking.

2.LITERATURE SURVEY:

2.1 Existing problem:

The solutions that are available in the market today are not able to address all the issues in one device. There are a few ways that the existing solutions work. First, with the use of a smartphone. This method might seem handy, but providing a young child with a smartphone in hand is not an ideal case, counter to the monetary investment for the phone, and the additional responsibility that the child has to take to handle and take care of the phone. This makes it a less feasible solution. The other way is via smartwatches that a child wears on the wrist. This may seem like an ideal solution, but the problem with this arises when the kidnapper is aware of such a device, and immediately removes the device from the child's wrist and destroys it. With the proposed solution, we make a discreet-looking device that doesn't look like a tracking device but is always with the child.Because of the way it looks, it does not distract the child, and with its small size.

2.2 Reference:

Paper-1:

NAME OF PAPER*:* Application of machine learning and IoT to enable child safety at home environment.

NAME OF AUTHOR: V. Shenbagalakshmi , T. Jaya

JOURNAL PUBLISHED: The Journal of Supercomputing,Volume 78,Issue 8 MONTH AND **YEAR PUBLISHED:** January 2022

OBJECTIVE OF THE PROJECT:

Safety of children is of utmost importance in any home environment. IoT when combined with machine learning is found to offer tremendous benefits in creating smart and safe homes to the society. The aim of this research is to apply machine learning models, in order to detect the anomaly on the dataset gathered from three IoT devices. The environmental parameters for which the anomaly is detected are smoke emission, light illumination, LPG gas emission, CO emission, motion detection, humidity changes and temperature-level changes. The research makes use of three machine learning models namely K-Means clustering, Isolation Forest and Interquartile Range to detect anomalies. In addition to that, it also uses Facebook Prophet Model to predict the daily trends in the data predicted by the three models. The evaluation of performance shows that the accuracy of predicting anomaly is greater for the Interquartile range model when compared with that of the remaining two machine learning models. The accuracy obtained by the IQR model is 99% whereas the models K-means and Isolation Forest render an accuracy of 94% each. The study also provides a scheme of hardware as a part of the future work that could be implemented in order to implement child safety in a better way in the near future.

Paper-2:

NAME OF PAPER: IoT-based Child Security Monitoring System

NAME OF AUTHOR: Lai Yi Heng,Intan Farahana Binti Kamsin

JOURNAL PUBLISHED: 3rd International Conference on Integrated Intelligent Computing Communication & Security (ICIIC 2021)

MONTH AND YEAR PUBLISHED: January 2021

OBJECTIVE OF THE PROJECT:

The crime rate associated with children keeps increasing due to which draws people's' attention regarding child safety. This research is conducted to propose a child security smart band utilizing IoT technology. Online questionnaire and semi-structured interview are methodologies used to collect data. The online questionnaire gains feedbacks by sending questions electronically, where answers need to be submitted online. In the semi structured interview, researcher meets and asks respondents some predetermined questions while other being asked are not planned in advanced. Through information obtained, a smart band have been proposed to monitor the safety of children. By this, parents know what is happening remotely and can take actions if something goes wrong. The future improvements of this device will be adding functions and software to make it works like a phone such as messaging, gallery, Google, YouTube, meanwhile, adding more child security features so that child safety is guaranteed.

Paper-3:

NAME OF PAPER: IoT-Enabled Smart Child Safety Digital System Architecture

NAME OF AUTHOR: Madhuri Madhuri; Asif Qumer Gill; Habib Ullah Khan

JOURNAL PUBLISHED: 2020 IEEE 14th International Conference on Semantic Computing **(ICSC) MONTH AND YEAR PUBLISHED:** February 2020

OBJECTIVE OF THE PROJECT:

Safety of a child in a large public event is a major concern for event organizers and parents. This paper addresses this important concern and proposes an architecture model of the IoT-enable smart child safety tracking digital system. This IoT-enabled digital system architecture integrates the Cloud, Mobile and GPS technology to precisely locate the geographical location of a child on an event map. The proposed architecture model describes the people, information, process, and technology architecture elements, and their relationships for the complex IoT-enable smart child safety tracking digital system. The proposed architecture model can be used as a reference or guide to assist in the safe architecture driven development of the various child tracking digital systems for different public events.

Paper-4:

NAME OF PAPER: iBaby: A Mobile Children Monitoring and Finding System with Stranger Holding Detection Based on IoT Technologies

NAME OF AUTHOR: Lien-Wu Chen,Tsung-Ping Chen,Chia-Chun Weng

JOURNAL PUBLISHED: The ACM SIGCOMM 2019 Conference Posters and Demos

MONTH AND YEAR PUBLISHED: August 2019

OBJECTIVE OF THE PROJECT:

This paper designs and implements a mobile children monitoring and finding system, called iBaby, using wearable devices and nearby smartphones to detect unexpected holding and find missing children through Internet of Things (IoT) technologies, respectively. In the monitoring mode, the iBaby system can prevent young children from taking away by strangers/people with bad intentions. In the finding mode, the iBaby system can cooperatively find missing children equipped with hand wearable devices consisting of the mobile iBeacon and 3-axis accelermeter through crowdsourced sensing networks formed by smartphone users with outdoor GPS and indoor IoT localization. To accurately detect stranger holding behaviors, multi-feature based, artificial neural network based, and convolutional neural network based posture recognition methods are designed to improve recognition success rates of iBaby as much as possible. In particular, an iOS-based prototype with Arduino wearable devices and static iBeacon nodes is implemented to verify the feasibility and correctness of our iBaby system.

Paper-5:

NAME OF PAPER: Smart IOT Device for Child Safety and Tracking

NAME OF AUTHOR: M Nandini Priyanka, S Murugan, KNH Srinivas, TDS Sarveswararao, E Kusuma Kumari.

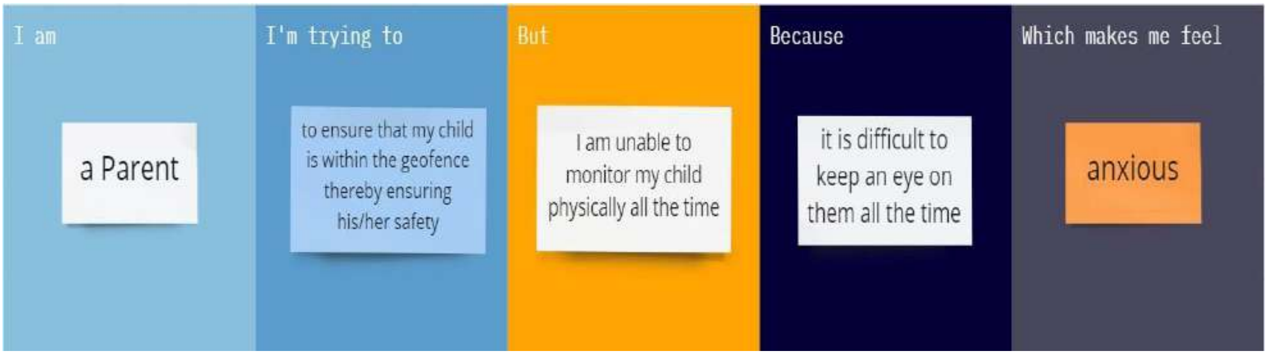
JOURNAL PUBLISHED: International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-8

MONTH AND YEAR PUBLISHED: 8 June, 2019

OBJECTIVE OF THE PROJECT:

Child safety and tracking is a major concern as the number of crimes on children are reported nowadays. With this motivation, a smart IoT device for child safety and tracking is developed to help the parents to locate and monitor their children. The system is developed using LinkIt ONE board programmed in embedded C and interfaced with temperature, heartbeat, touch sensors and also GPS, GSM & digital camera modules. The novelty of the work is that the system automatically alerts the parent/caretaker by sending SMS, when immediate attention is required for the child during an emergency. The parameters such as touch, temperature & heartbeat of the child are used for parametric analysis and results are plotted for the same. The above system ensures the safety and tracking of children.

2.3 Problem Statement Definition:



I am

a Parent

I'm trying to

to ensure that my child is within the geofence thereby ensuring his/her safety

But

I am unable to monitor my child physically all the time

Because

it is difficult to keep an eye on them all the time

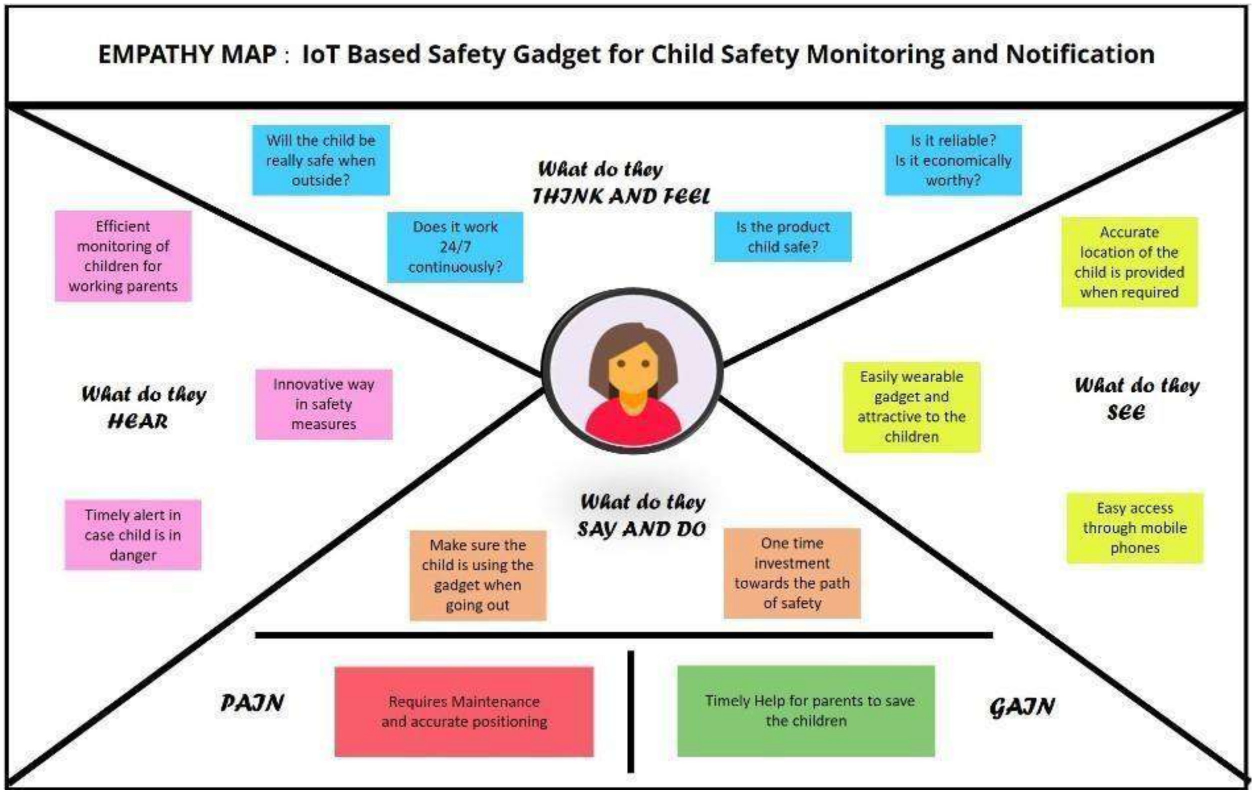
Which makes me feel

anxious

3.IDEATION & PROPOSED SOLUTION:

3.1 Empathy Map Canvas:

An empathy map is a simple, easy-to-digest visual that captures knowledge about a user’s behaviors and attitudes. It is a useful tool to help teams better understand their users. Creating an effective solution requires understanding the true problem and the person who is experiencing it. The exercise of creating the map helps participants consider things from the user’s perspective along with his or her goals and challenges..


EMPATHY MAP : IoT Based Safety Gadget for Child Safety Monitoring and Notification
What do they THINK AND FEEL
Will the child be really safe when outside?
Does it work 24/7 continuously?
Is the product child safe?
Is it reliable? Is it economically worthy?
What do they HEAR
Efficient monitoring of children for working parents
Innovative way in safety measures
Timely alert in case child is in danger
What do they SEE
Accurate location of the child is provided when required
Easily wearable gadget and attractive to the children
Easy access through mobile phones
What do they SAY AND DO
Make sure the child is using the gadget when going out
One time investment towards the path of safety
PAIN
Requires Maintenance and accurate positioning
GAIN
Timely Help for parents to save the children

3.2 Ideation & Brainstorming:

Step-1:Team Gathering ,Collaboration and Select the Problem statement:

We have followed the first step of brainstorming ,we have discussed as a team to decide a problem statement,

As per the guideline the following is done

1.Team Gathering

2.Collaboration

3.Deciding the problem statement

2.3 Problem Statement Definition:

PROBLEM STATEMENT:

To create a smart wearable device for children that uses advanced technology to ensure their safety all the time.

Step-2: Brainstorm, Idea Listing and Grouping:

BRAINSTORMING:

TEAM MEMBER	MEMBERS OPNION	IDEAS
HARSHAVARDHINI	MAY GIV…	Collecting vital signs of the child like Blood Volume Pulse(BVP) Galvanic Skin Response (GSR) to help parents to monitor child. An abrupt change in this should trigger alert.
HARIPPRIYADHARSHINI	GOOD	Detecting locations using GPS and sending the information to parents using GSM.
HITHAYATHUN NIHMA	BEST	Detecting location using GPS and sending the information to parents through cloud services.
KAURI SHEETAL P	COSTLY	Application of Machine Learning and cameras to enable child safety by monitoring every movement.

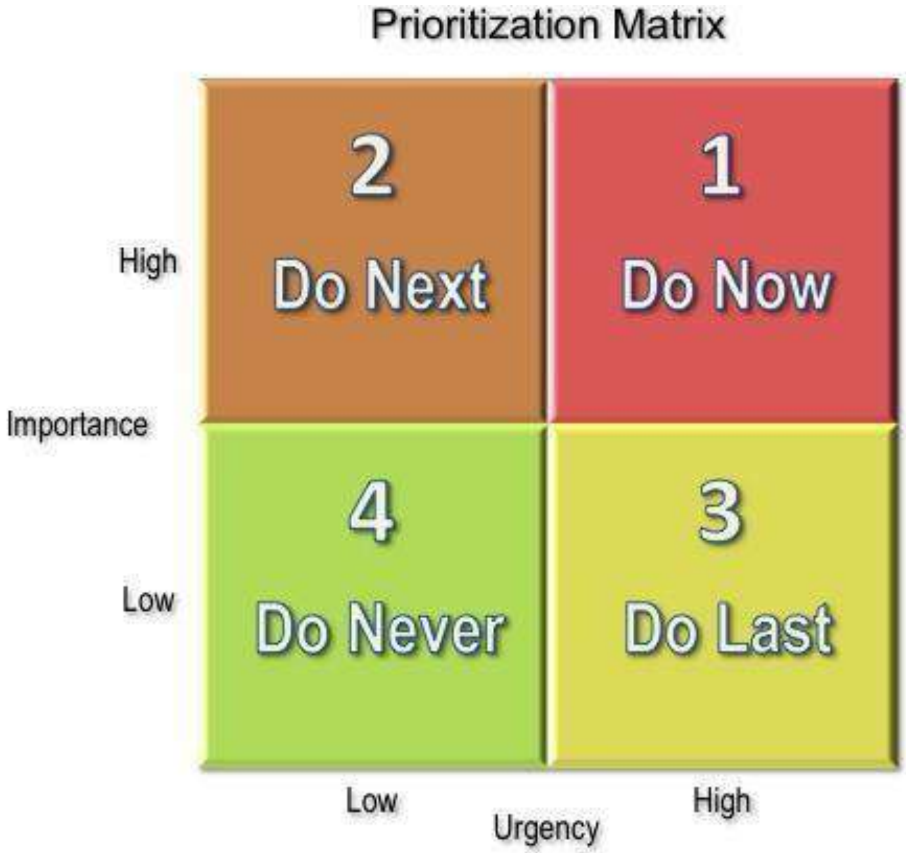
BY GROUPING IDEAS:

1.We have planned to use GSP for detecting location and to sent the message parents we can use GSM or Cloud.

2.The location will be shared in application or as message to parents

3.The notification is also sent incase of emergencies

Step-3: Idea Prioritization:



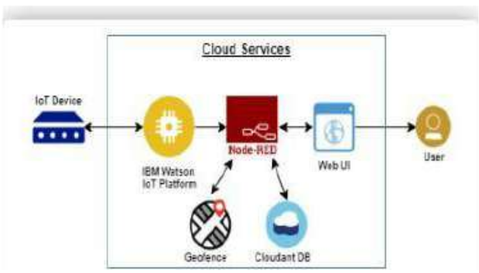
1.Using location trackers like GPS inorder to detect the location of the child.

2.The GPS are connected to IOT application inorder notify the parents

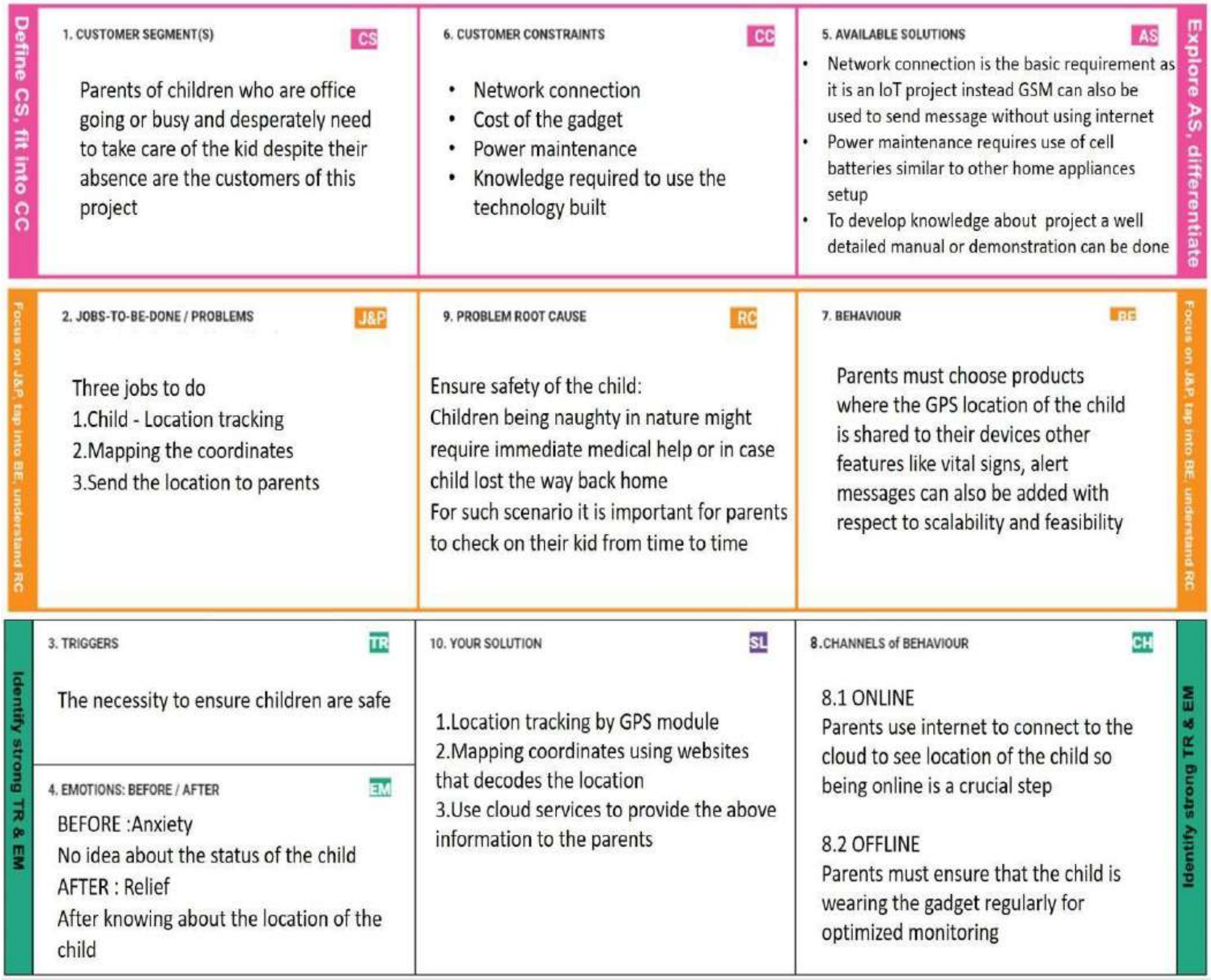
3.Use other sensors if required for the particular application such as temperature,heart beats etc.

3.3 Proposed Solution:

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	A parent has to ensure the safety of his child from anywhere in case of need
2.	Idea / Solution description	Three Steps: **1.Location tracking 2.Mapping the coordinates 3.Send message to parents** **Step 1:** generally requires a GPS module and the coordinates need to be updated periodically Here we are going to use an assumed longitude latitude location in python code **Step 2:** is done using world map node in NODE RED app facility in IBM Cloud that maps the coordinates on map as a pin location This location is also fed to geofence node that checks whether the location is within the radius or not **Step 3:** ensures message is shown if location is not inside geofence area For real world application GPS module can be given to children in form of watch or any other sophisticated gadget

3.	Novelty / Uniqueness	Parents will receive accurate location anytime anywhere (they will know if the child is in the geofence or not)
4.	Social Impact / Customer Satisfaction	Child Safety will be within reach for the parents. Gadget will be childsafe and environment friendly
5.	Business Model (Revenue Model)	Technical Architecture:  Cloud Services IoT Device IBM Watson IoT Platform Node-RED Web UI User Geofence Cloudant DB
6.	Scalability of the Solution	Requires proper internet connection for working with cloud

3.4 Problem Solution fit:



Define CS, fit into CC

1. CUSTOMER SEGMENT(S) CS

Parents of children who are office going or busy and desperately need to take care of the kid despite their absence are the customers of this project

6. CUSTOMER CONSTRAINTS CC

- Network connection
- Cost of the gadget
- Power maintenance
- Knowledge required to use the technology built

5. AVAILABLE SOLUTIONS AS

- Network connection is the basic requirement as it is an IoT project instead GSM can also be used to send message without using internet
- Power maintenance requires use of cell batteries similar to other home appliances setup
- To develop knowledge about project a well detailed manual or demonstration can be done

Explore AS, differentiate

Focus on J&P, tap into BE, understand RC

2. JOBS-TO-BE-DONE / PROBLEMS J&P

Three jobs to do
1.Child - Location tracking
2.Mapping the coordinates
3.Send the location to parents

9. PROBLEM ROOT CAUSE RC

Ensure safety of the child:
Children being naughty in nature might require immediate medical help or in case child lost the way back home
For such scenario it is important for parents to check on their kid from time to time

7. BEHAVIOUR BE

Parents must choose products where the GPS location of the child is shared to their devices other features like vital signs, alert messages can also be added with respect to scalability and feasibility

Focus on J&P, tap into BE, understand RC

Identify strong TR & EM

3. TRIGGERS TR

The necessity to ensure children are safe

4. EMOTIONS: BEFORE / AFTER EM

BEFORE :Anxiety
No idea about the status of the child
AFTER : Relief
After knowing about the location of the child

10. YOUR SOLUTION SL

1.Location tracking by GPS module
2.Mapping coordinates using websites that decodes the location
3.Use cloud services to provide the above information to the parents

8.CHANNELS of BEHAVIOUR CH

8.1 ONLINE
Parents use internet to connect to the cloud to see location of the child so being online is a crucial step

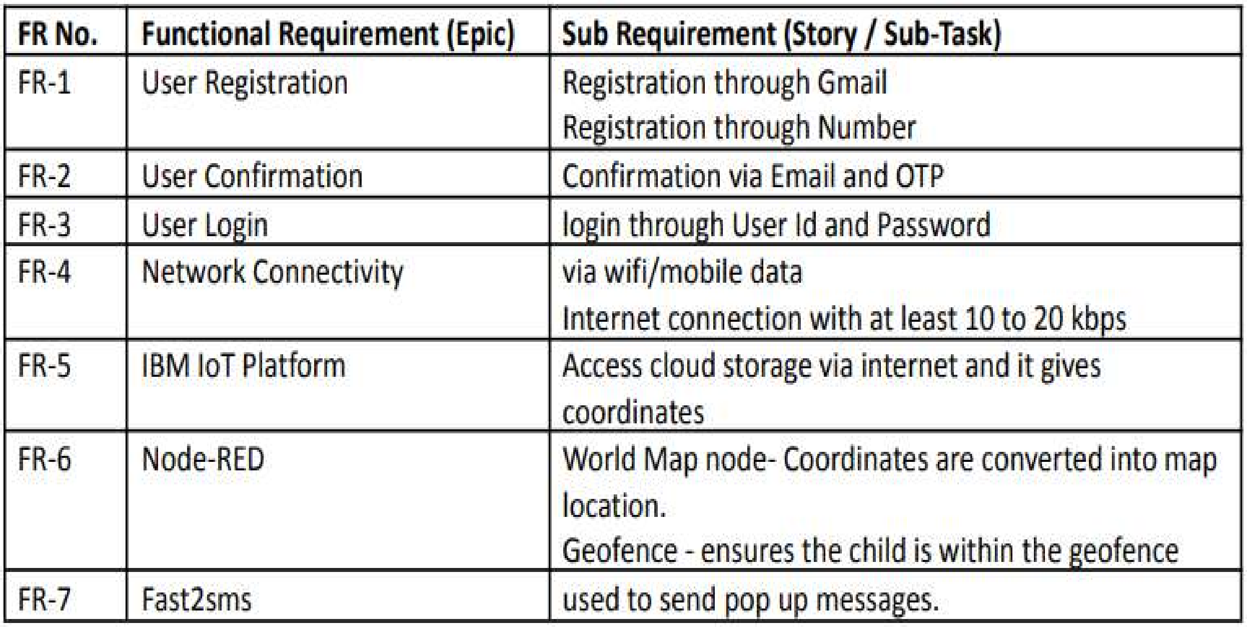
8.2 OFFLINE
Parents must ensure that the child is wearing the gadget regularly for optimized monitoring

Identify strong TR & EM

4. REQUIREMENT ANALYSIS

4.1 Functional requirement

Following are the functional requirements of the proposed solution



FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	User Registration	Registration through Gmail Registration through Number
FR-2	User Confirmation	Confirmation via Email and OTP
FR-3	User Login	login through User Id and Password
FR-4	Network Connectivity	via wifi/mobile data Internet connection with at least 10 to 20 kbps
FR-5	IBM IoT Platform	Access cloud storage via internet and it gives coordinates
FR-6	Node-RED	World Map node- Coordinates are converted into map location. Geofence - ensures the child is within the geofence
FR-7	Fast2sms	used to send pop up messages.

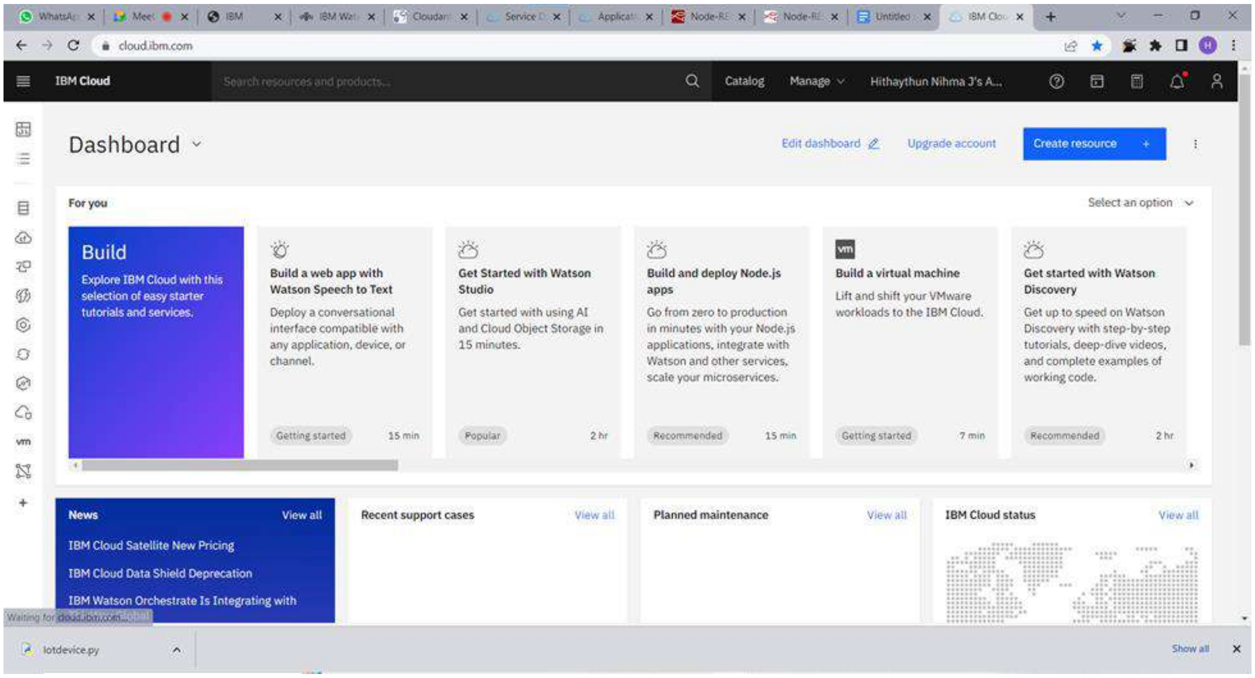
1.Proper network facility is required to import and export tracking information of the child's location to and from the IBM cloud and Cloudant database

2.The Python Script developed should be running all the time in order to detect the inarea and out area location of the child which will be sent to the IBM cloud and pass through several functions and from which the result will be shown in the World Map node.

3.Since the Python code keeps on running the data keeps on updating in the IBM cloud which leads to increase in power consumption and also we need a reset option to clear data in cloudant whenever not in use.

4.The fast2sms connection should be ensured in order to send sms to the respective parent whenever the child crosses the geofence area.

IBM CLOUD:



Dashboard

For you

Build
Explore IBM Cloud with this selection of easy starter tutorials and services.

Build a web app with Watson Speech to Text
Deploy a conversational interface compatible with any application, device, or channel.
Getting started 15 min

Get Started with Watson Studio
Get started with using AI and Cloud Object Storage in 15 minutes.
Popular 2 hr

Build and deploy Node.js apps
Go from zero to production in minutes with your Node.js applications, integrate with Watson and other services, scale your microservices.
Recommended 15 min

Build a virtual machine
Lift and shift your VMware workloads to the IBM Cloud.
Getting started 7 min

Get started with Watson Discovery
Get up to speed on Watson Discovery with step-by-step tutorials, deep-dive videos, and complete examples of working code.
Recommended 2 hr

News View all
IBM Cloud Satellite New Pricing
IBM Cloud Data Shield Deprecation
IBM Watson Orchestrate Is Integrating with

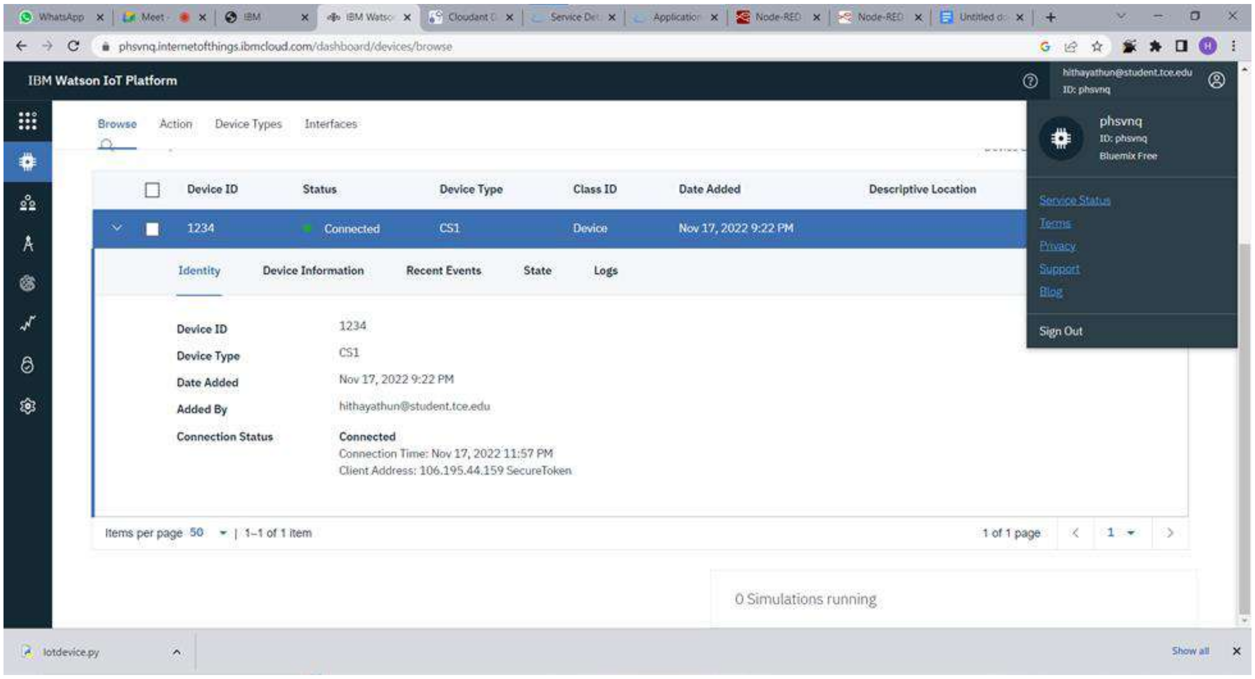
Recent support cases View all

Planned maintenance View all

IBM Cloud status View all

.

IBM IoT Platform: (Device Creation)



□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
□	1234	Connected	CS1	Device	Nov 17, 2022 9:22 PM	

Identity | Device Information | Recent Events | State | Logs

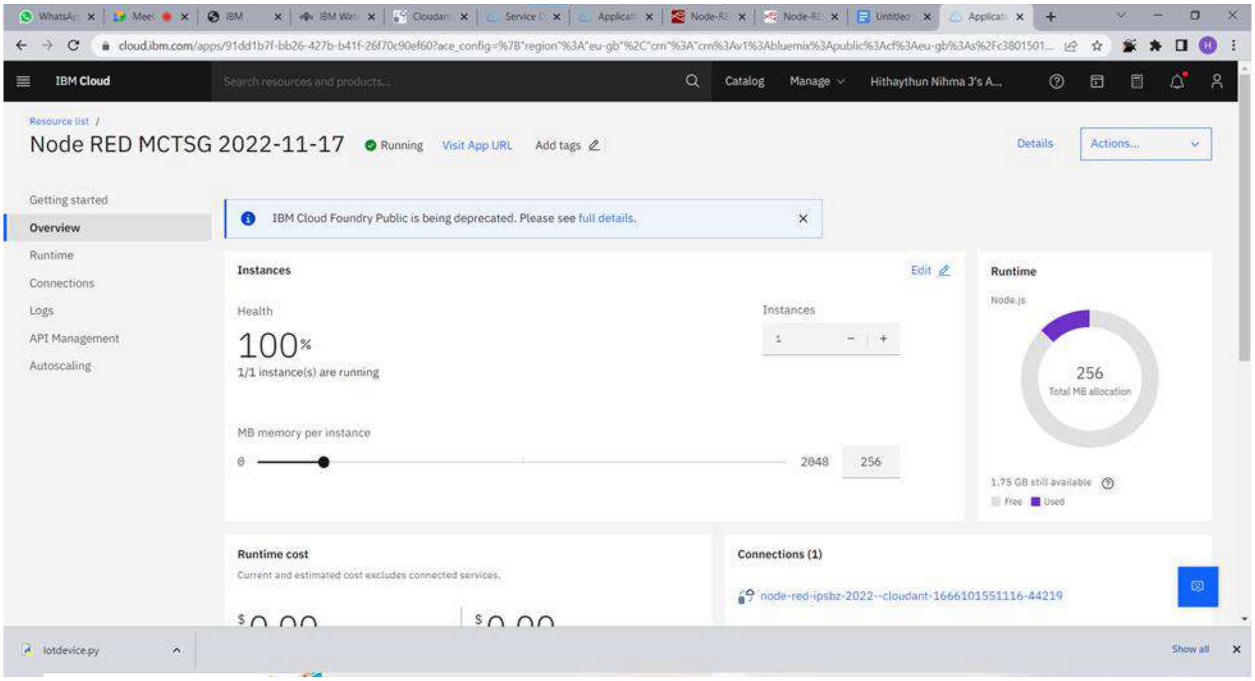
Device ID	1234
Device Type	CS1
Date Added	Nov 17, 2022 9:22 PM
Added By	hithayathun@student.tce.edu
Connection Status	Connected Connection Time: Nov 17, 2022 11:57 PM Client Address: 106.195.44.159 SecureToken

Items per page 50 | 1–1 of 1 item

1 of 1 page

0 Simulations running

NODERED:



cloud.ibm.com/apps/91dd1b7f-bb26-427b-b41f-26f70c90ef60?ace_config=%7B"region"%3A"eu-gb"%2C"crn"%3A"crn%3Av1%3Abluemix%3Apublic%3Acf%3Aeu-gb%3As%2Fc3801501...

Node RED MCTSG 2022-11-17 Running Visit App URL Add tags

IBM Cloud Foundry Public is being deprecated. Please see full details.

Instances

Health

100%

1/1 instance(s) are running

Instances 1

MB memory per instance 0 2048 256

Runtime

256 Total MB allocation

1.75 GB still available

Runtime cost

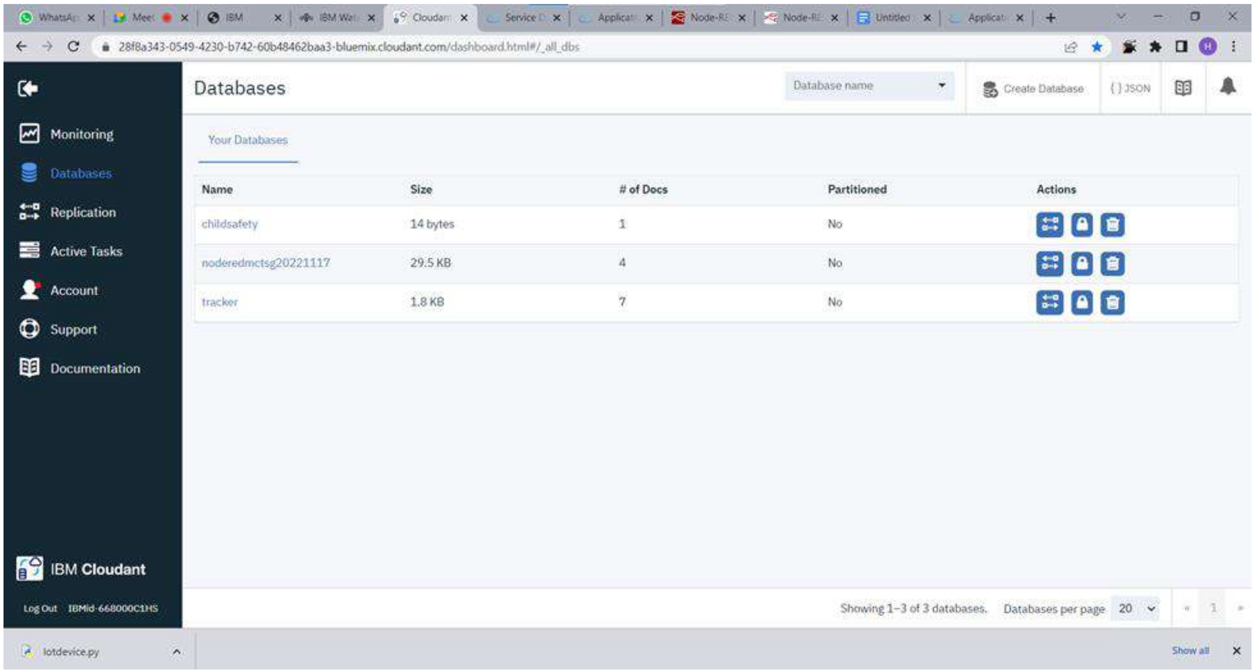
Current and estimated cost excludes connected services.

$0.00 $0.00

Connections (1)

node-red-ipsbz-2022--cloudant-1666101551116-44219

CLOUDANT DATABASE:



28f8a343-0549-4230-b742-60b48462baa3-bluemix.cloudant.com/dashboard.html#/_all_dbs

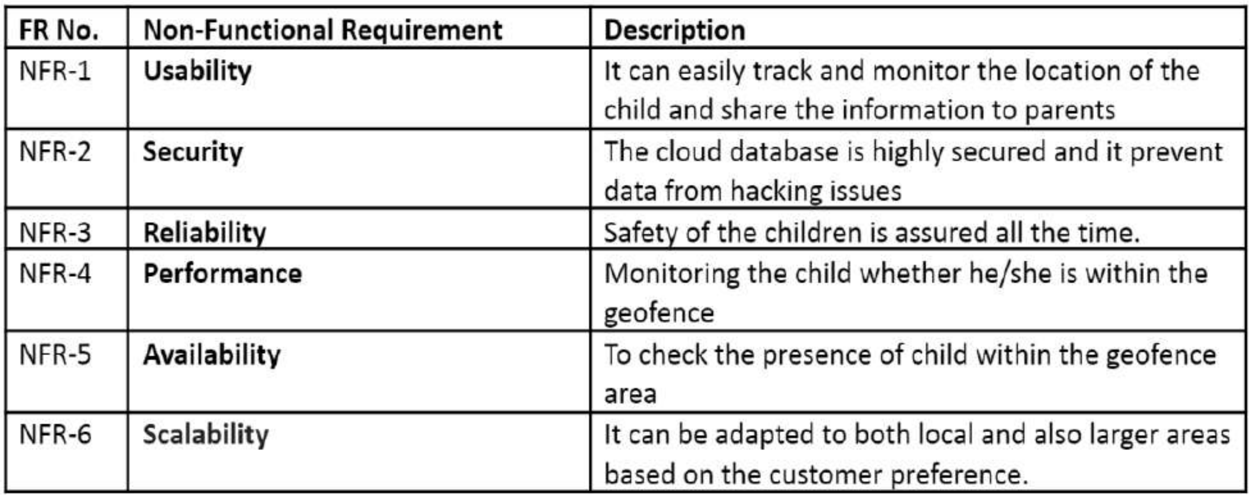
Databases

Name	Size	# of Docs	Partitioned	Actions
childsafety	14 bytes	1	No	
noderedmctsg20221117	29.5 KB	4	No	
tracker	1.8 KB	7	No	

Showing 1–3 of 3 databases. Databases per page 20

4.2 Non-Functional requirement

Following are the non functional requirements of proposed solution.



FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	It can easily track and monitor the location of the child and share the information to parents
NFR-2	**Security**	The cloud database is highly secured and it prevent data from hacking issues
NFR-3	**Reliability**	Safety of the children is assured all the time.
NFR-4	**Performance**	Monitoring the child whether he/she is within the geofence
NFR-5	**Availability**	To check the presence of child within the geofence area
NFR-6	**Scalability**	It can be adapted to both local and also larger areas based on the customer preference.

1.The child should wear the tracking gadget all time so that the location information is updated periodically.

2.The parent should have access to fast2sms api inorder to get location information of the child.

3.The gadget should consume less power and it should have long battery life.

Incase of losing the gadget by playfulness of children is a drawback that cannot be technically met.

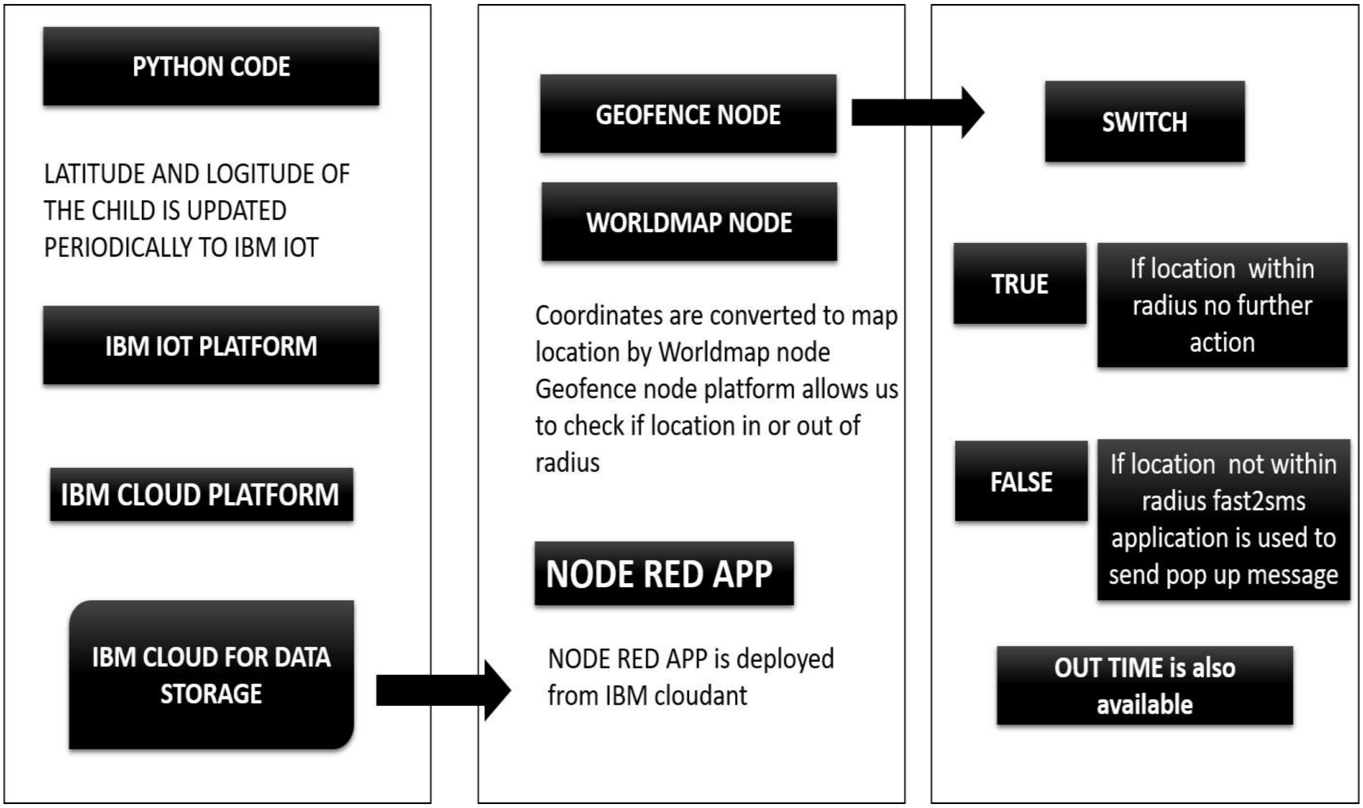
4.To overcome this a backup device can be used without the knowledge of the child.GPS Module should be placed in the gadget to get the latitude and longitude details of the child.

5.PROJECT DESIGN:

5.1 Data Flow Diagrams:

A data-flow diagram is a way of representing a flow of data through a process or a system. The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow — there are no decision rules and no loops

A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement. They are often elements of a formal methodology such as Structured Systems Analysis and Design Method (SSADM).



PYTHON CODE

LATITUDE AND LOGITUDE OF THE CHILD IS UPDATED PERIODICALLY TO IBM IOT

IBM IOT PLATFORM

IBM CLOUD PLATFORM

IBM CLOUD FOR DATA STORAGE

GEOFENCE NODE

WORLDMAP NODE

Coordinates are converted to map location by Worldmap node Geofence node platform allows us to check if location in or out of radius

NODE RED APP

NODE RED APP is deployed from IBM cloudant

SWITCH

TRUE

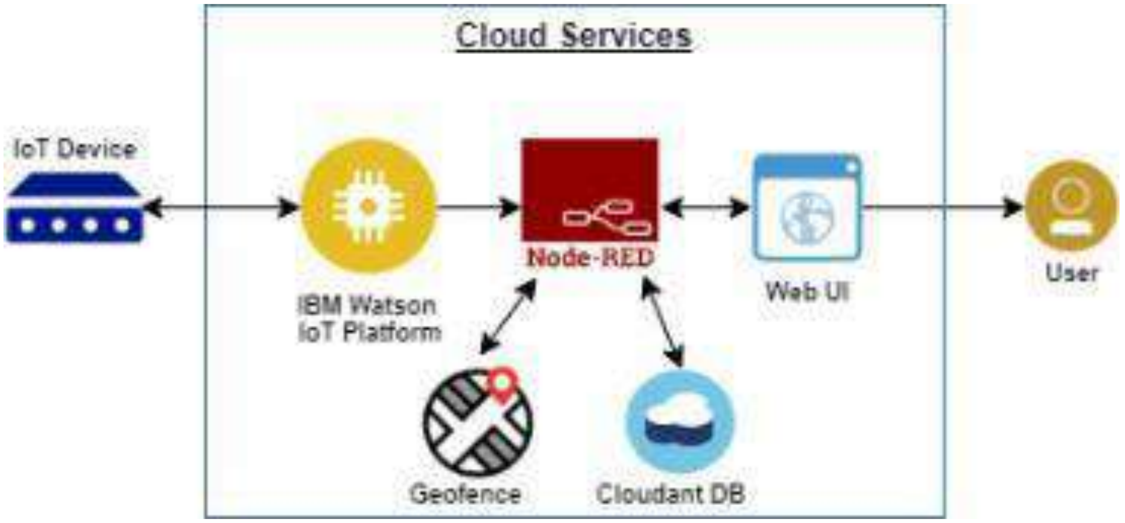
If location within radius no further action

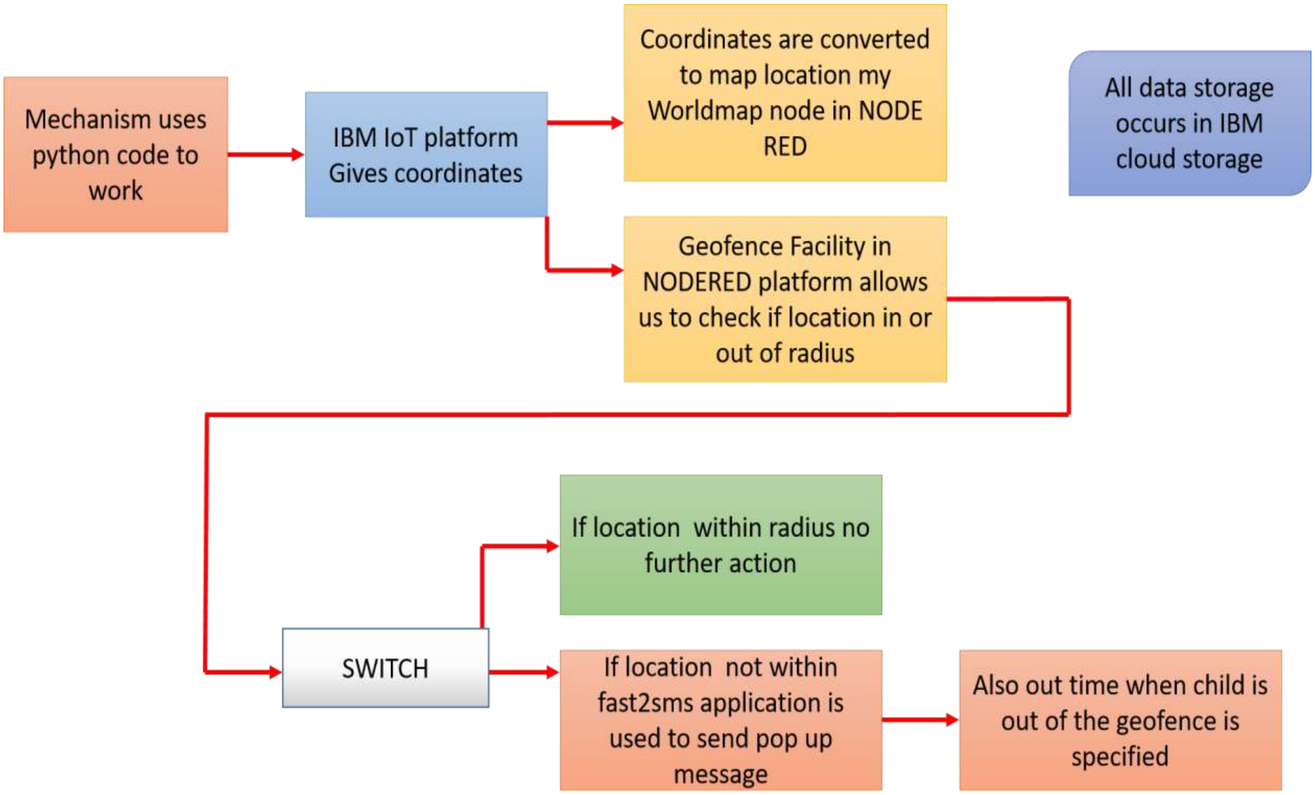
FALSE

If location not within radius fast2sms application is used to send pop up message

OUT TIME is also available

5.2 Solution & Technical Architecture:





Mechanism uses python code to work

IBM IoT platform Gives coordinates

Coordinates are converted to map location my Worldmap node in NODE RED

All data storage occurs in IBM cloud storage

Geofence Facility in NODERED platform allows us to check if location in or out of radius

SWITCH

If location within radius no further action

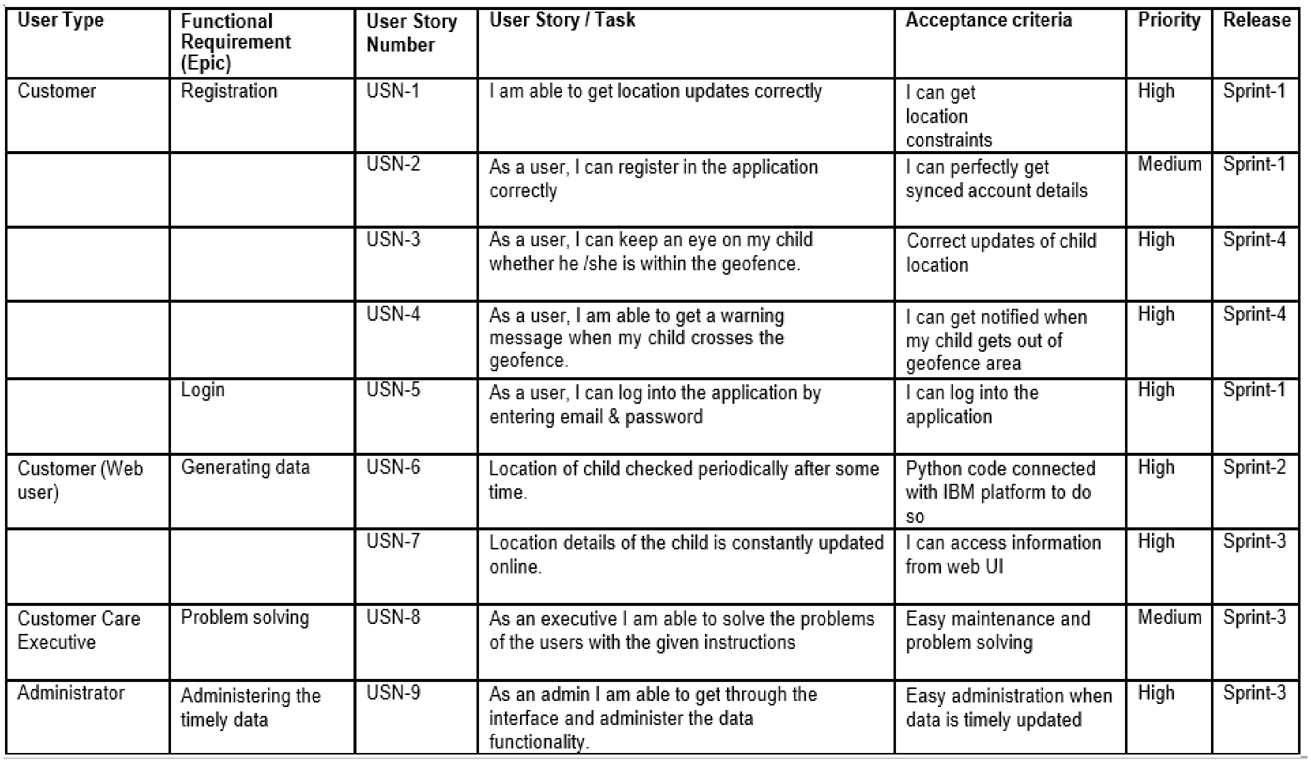
If location not within fast2sms application is used to send pop up message

Also out time when child is out of the geofence is specified

5.3 User Stories:

In software development and product management, a user story is an informal, natural language description of features of a software system. They are written from the perspective of an end user or user of a system, and may be recorded on index cards, Post-it notes, or digitally in project management software.

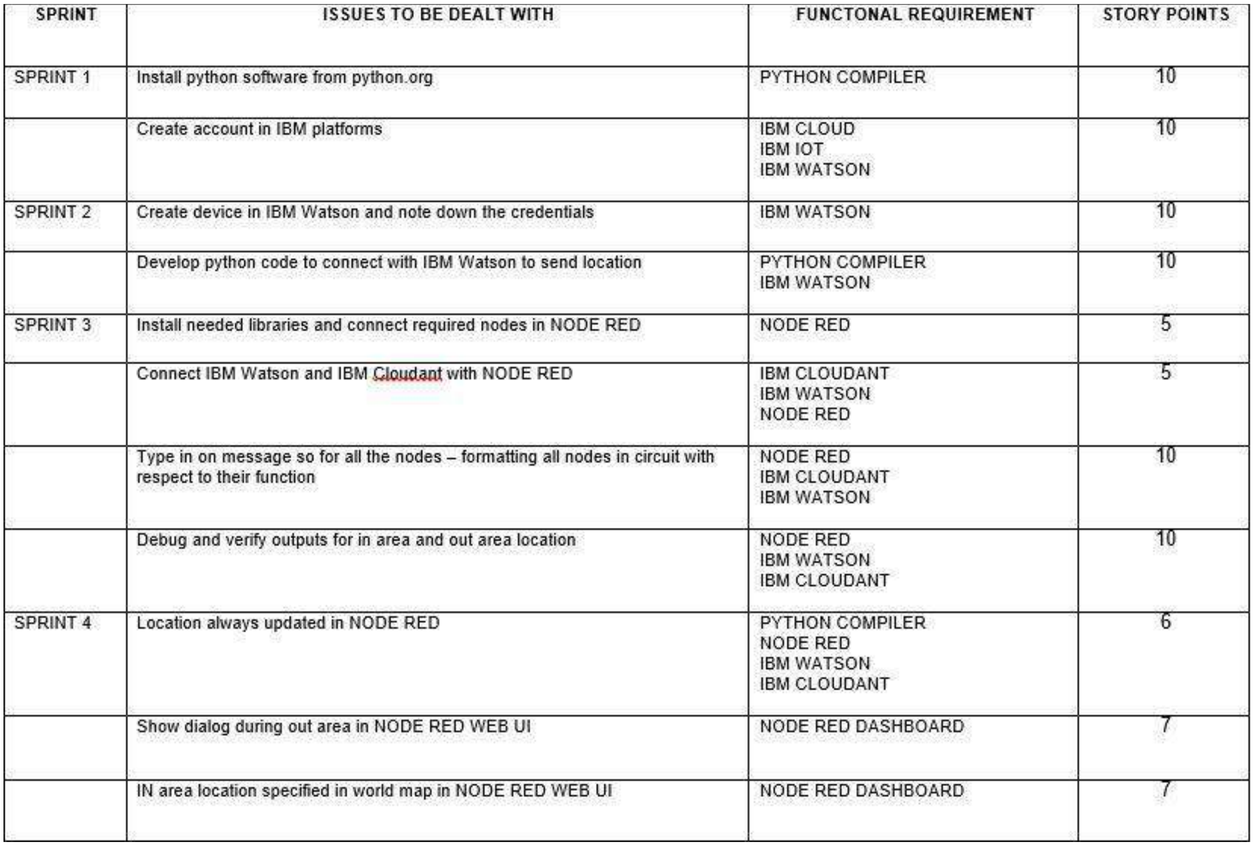
In agile software development, user stories help articulate what value a product feature can bring and have a better understanding of why users want a certain functionality. It helps the product manager and development team shift their focus from writing about the software features to discussing the features



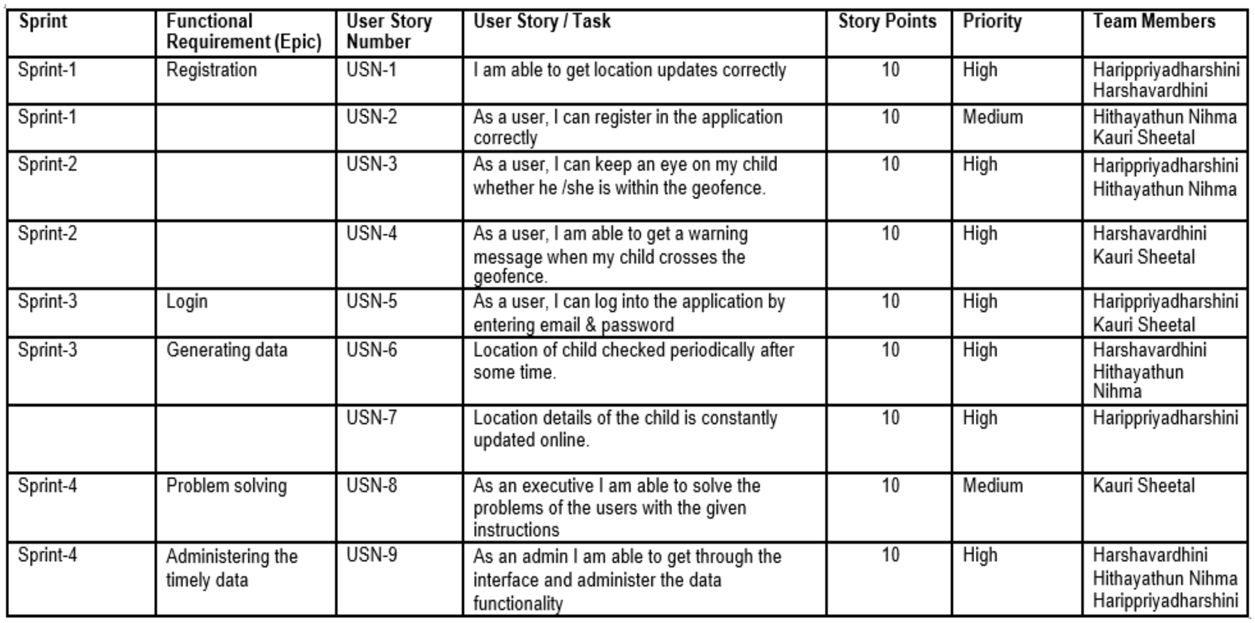
User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer	Registration	USN-1	I am able to get location updates correctly	I can get location constraints	High	Sprint-1
		USN-2	As a user, I can register in the application correctly	I can perfectly get synced account details	Medium	Sprint-1
		USN-3	As a user, I can keep an eye on my child whether he /she is within the geofence.	Correct updates of child location	High	Sprint-4
		USN-4	As a user, I am able to get a warning message when my child crosses the geofence.	I can get notified when my child gets out of geofence area	High	Sprint-4
	Login	USN-5	As a user, I can log into the application by entering email & password	I can log into the application	High	Sprint-1
Customer (Web user)	Generating data	USN-6	Location of child checked periodically after some time.	Python code connected with IBM platform to do so	High	Sprint-2
		USN-7	Location details of the child is constantly updated online.	I can access information from web UI	High	Sprint-3
Customer Care Executive	Problem solving	USN-8	As an executive I am able to solve the problems of the users with the given instructions	Easy maintenance and problem solving	Medium	Sprint-3
Administrator	Administering the timely data	USN-9	As an admin I am able to get through the interface and administer the data functionality.	Easy administration when data is timely updated	High	Sprint-3

6.PROJECT PLANNING AND SCHEDULING

6.1 Sprint Planning and Estimation

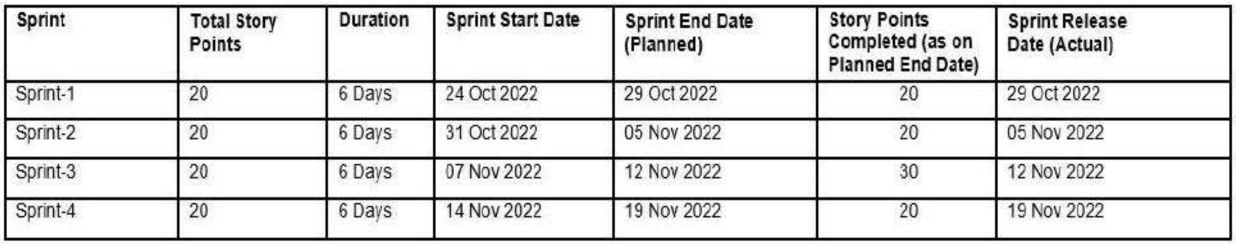


SPRINT	ISSUES TO BE DEALT WITH	FUNCTONAL REQUIREMENT	STORY POINTS
SPRINT 1	Install python software from python.org	PYTHON COMPILER	10
	Create account in IBM platforms	IBM CLOUD IBM IOT IBM WATSON	10
SPRINT 2	Create device in IBM Watson and note down the credentials	IBM WATSON	10
	Develop python code to connect with IBM Watson to send location	PYTHON COMPILER IBM WATSON	10
SPRINT 3	Install needed libraries and connect required nodes in NODE RED	NODE RED	5
	Connect IBM Watson and IBM Cloudant with NODE RED	IBM CLOUDANT IBM WATSON NODE RED	5
	Type in on message so for all the nodes – formatting all nodes in circuit with respect to their function	NODE RED IBM CLOUDANT IBM WATSON	10
	Debug and verify outputs for in area and out area location	NODE RED IBM WATSON IBM CLOUDANT	10
SPRINT 4	Location always updated in NODE RED	PYTHON COMPILER NODE RED IBM WATSON IBM CLOUDANT	6
	Show dialog during out area in NODE RED WEB UI	NODE RED DASHBOARD	7
	IN area location specified in world map in NODE RED WEB UI	NODE RED DASHBOARD	7



Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	I am able to get location updates correctly	10	High	Harippriyadharshini Harshavardhini
Sprint-1		USN-2	As a user, I can register in the application correctly	10	Medium	Hithayathun Nihma Kauri Sheetal
Sprint-2		USN-3	As a user, I can keep an eye on my child whether he /she is within the geofence.	10	High	Harippriyadharshini Hithayathun Nihma
Sprint-2		USN-4	As a user, I am able to get a warning message when my child crosses the geofence.	10	High	Harshavardhini Kauri Sheetal
Sprint-3	Login	USN-5	As a user, I can log into the application by entering email & password	10	High	Harippriyadharshini Kauri Sheetal
Sprint-3	Generating data	USN-6	Location of child checked periodically after some time.	10	High	Harshavardhini Hithayathun Nihma
		USN-7	Location details of the child is constantly updated online.	10	High	Harippriyadharshini
Sprint-4	Problem solving	USN-8	As an executive I am able to solve the problems of the users with the given instructions	10	Medium	Kauri Sheetal
Sprint-4	Administering the timely data	USN-9	As an admin I am able to get through the interface and administer the data functionality	10	High	Harshavardhini Hithayathun Nihma Harippriyadharshini

6.2 Sprint Delivery Schedule



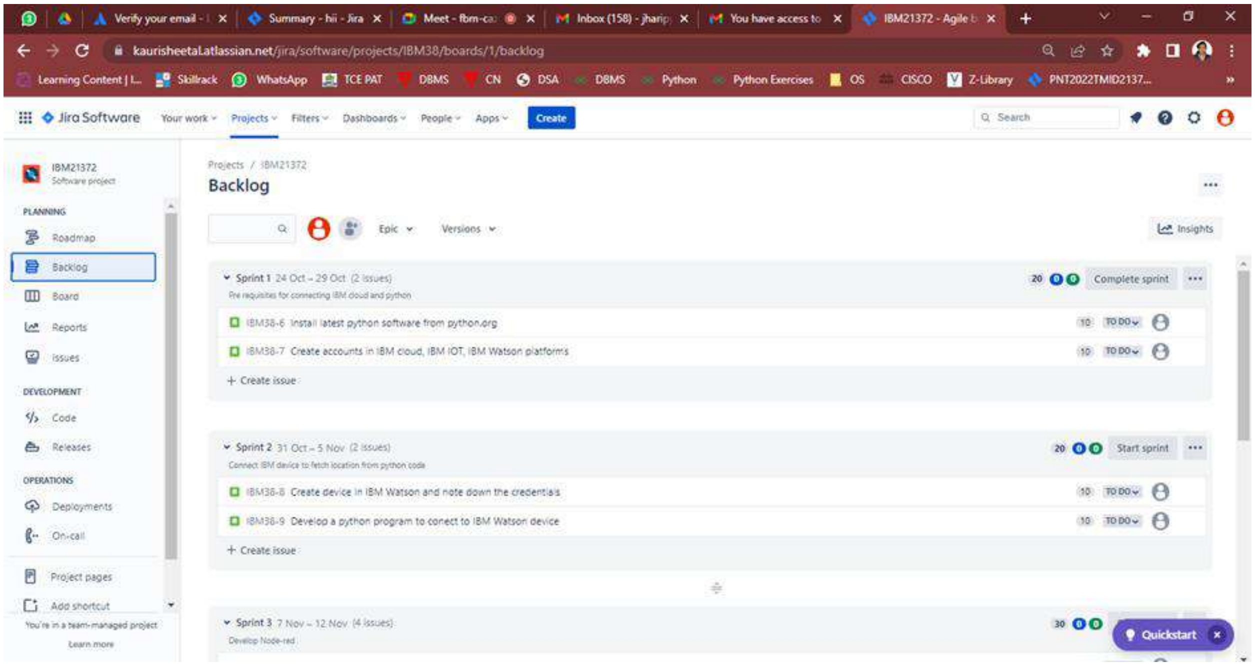
Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	24 Oct 2022	29 Oct 2022	20	29 Oct 2022
Sprint-2	20	6 Days	31 Oct 2022	05 Nov 2022	20	05 Nov 2022
Sprint-3	20	6 Days	07 Nov 2022	12 Nov 2022	30	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Velocity:
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

6.3 Reports from JIRA

Sprint Creation



Backlog

Sprint 1 24 Oct – 29 Oct (2 issues) 20 0 0
Pre requisites for connecting IBM cloud and python

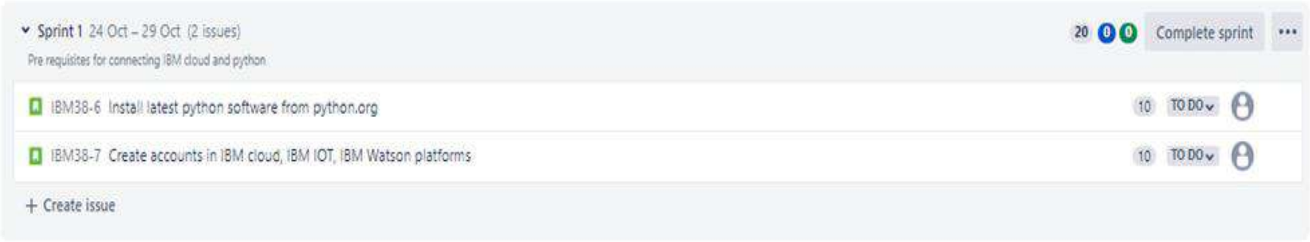
- IBM38-6 Install latest python software from python.org 10 TO DO
- IBM38-7 Create accounts in IBM cloud, IBM IOT, IBM Watson platforms 10 TO DO

Sprint 2 31 Oct – 5 Nov (2 issues) 20 0 0
Connect IBM device to fetch location from python code

- IBM38-8 Create device in IBM Watson and note down the credentials 10 TO DO
- IBM38-9 Develop a python program to conect to IBM Watson device 10 TO DO

Sprint 3 7 Nov – 12 Nov (4 issues) 30 0 0
Develop Node-red

Sprints:

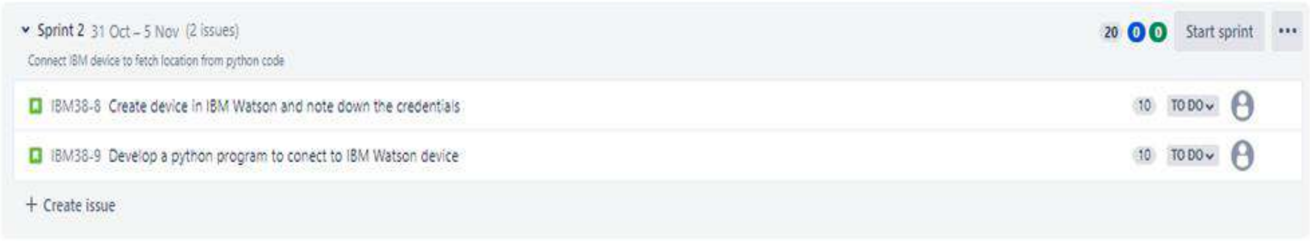


Sprint 1 24 Oct – 29 Oct (2 issues) 20 0 0 Complete sprint ···

Pre requisites for connecting IBM cloud and python

- IBM38-6 Install latest python software from python.org 10 TO DO
- IBM38-7 Create accounts in IBM cloud, IBM IOT, IBM Watson platforms 10 TO DO

+ Create issue

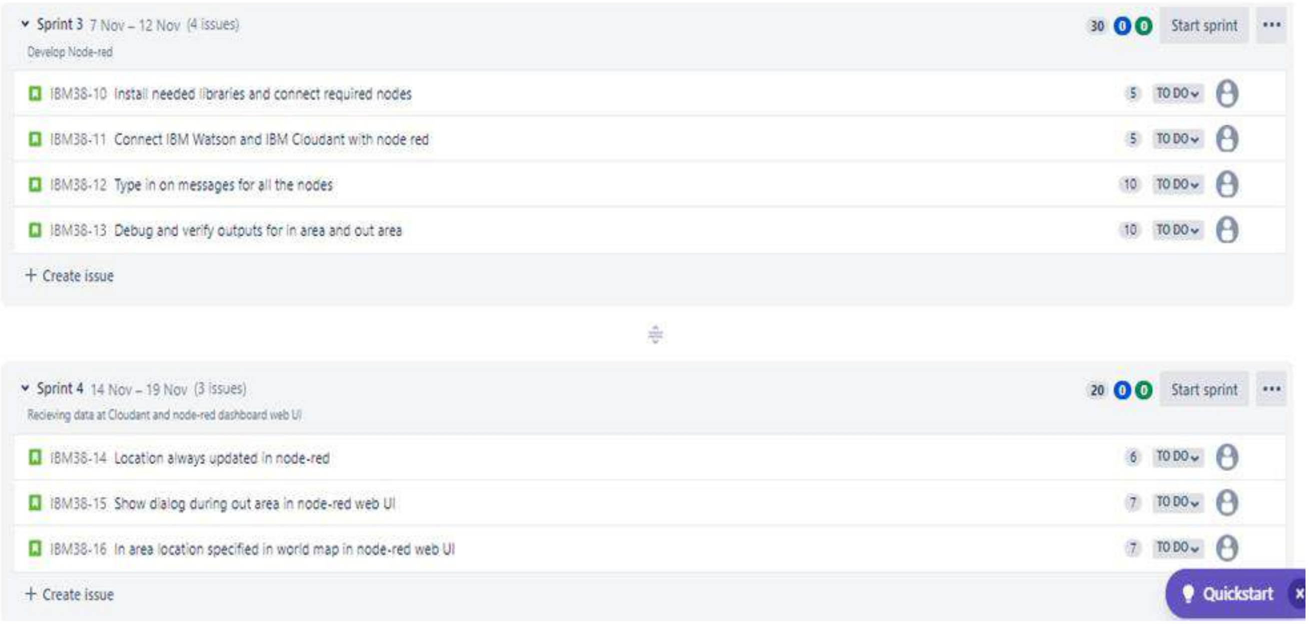


Sprint 2 31 Oct – 5 Nov (2 issues) 20 0 0 Start sprint ···

Connect IBM device to fetch location from python code

- IBM38-8 Create device in IBM Watson and note down the credentials 10 TO DO
- IBM38-9 Develop a python program to conect to IBM Watson device 10 TO DO

+ Create issue



Sprint 3 7 Nov – 12 Nov (4 issues) 30 0 0 Start sprint ···

Develop Node-red

- IBM38-10 Install needed libraries and connect required nodes 5 TO DO
- IBM38-11 Connect IBM Watson and IBM Cloudant with node red 5 TO DO
- IBM38-12 Type in on messages for all the nodes 10 TO DO
- IBM38-13 Debug and verify outputs for in area and out area 10 TO DO

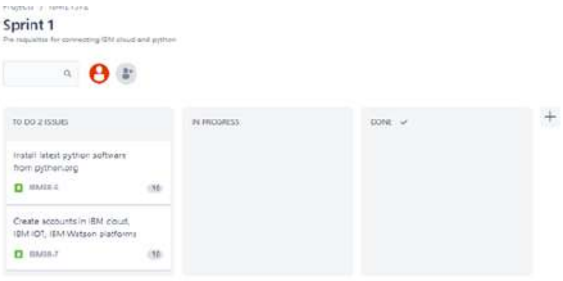
+ Create issue

Sprint 4 14 Nov – 19 Nov (3 issues) 20 0 0 Start sprint ···

Recieving data at Cloudant and node-red dashboard web UI

- IBM38-14 Location always updated in node-red 6 TO DO
- IBM38-15 Show dialog during out area in node-red web UI 7 TO DO
- IBM38-16 In area location specified in world map in node-red web UI 7 TO DO

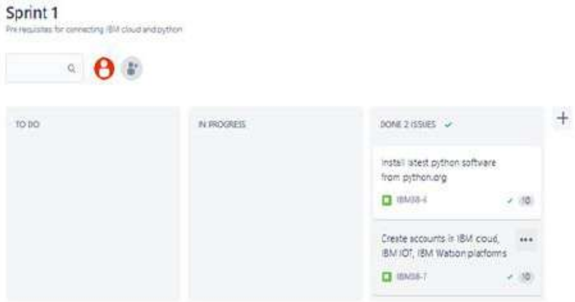
+ Create issue



Sprint 1

Pre requisites for connecting IBM cloud and python

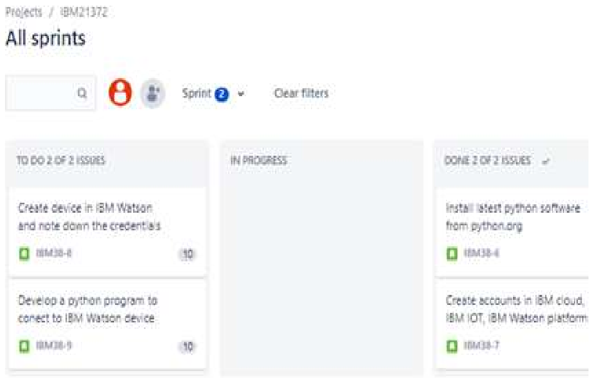
TO DO 2 ISSUES	IN PROGRESS	DONE ✓
Install latest python software from python.org IBM38-6 10 Create accounts in IBM cloud, IBM IOT, IBM Watson platforms IBM38-7 10		



Sprint 1

Pre requisites for connecting IBM cloud and python

TO DO	IN PROGRESS	DONE 2 ISSUES ✓
		Install latest python software from python.org IBM38-6 ✓ 10 Create accounts in IBM coud, IBM IOT, IBM Watson platforms IBM38-7 ✓ 10



Projects / IBM21372

All sprints

Sprint 2 · Clear filters

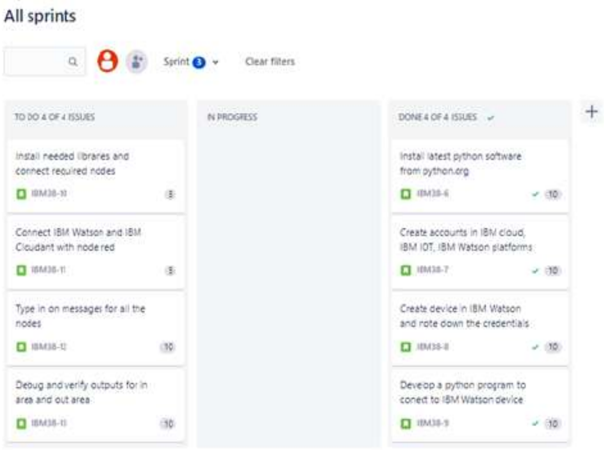
TO DO 2 OF 2 ISSUES

- Create device in IBM Watson and note down the credentials — IBM38-8 — 10
- Develop a python program to conect to IBM Watson device — IBM38-9 — 10

IN PROGRESS

DONE 2 OF 2 ISSUES ✓

- Install latest python software from python.org — IBM38-6
- Create accounts in IBM cloud, IBM IOT, IBM Watson platform — IBM38-7



All sprints

Sprint 3 · Clear filters

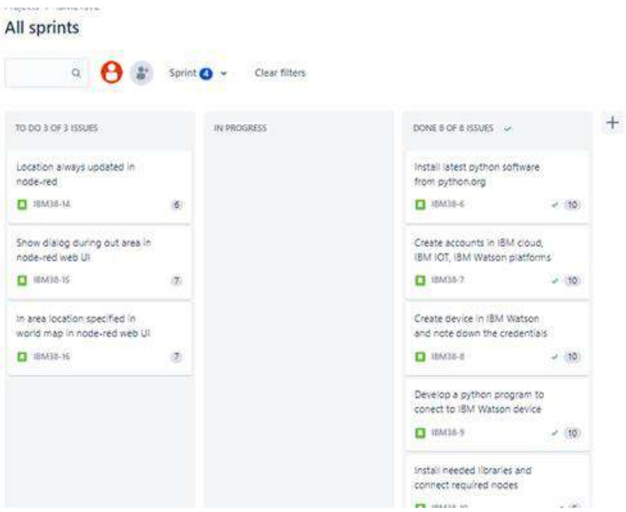
TO DO 4 OF 4 ISSUES

- Install needed libraries and connect required nodes — IBM38-10 — 5
- Connect IBM Watson and IBM Cloudant with node red — IBM38-11 — 5
- Type in on messages for all the nodes — IBM38-12 — 10
- Debug and verify outputs for in area and out area — IBM38-11 — 10

IN PROGRESS

DONE 4 OF 4 ISSUES ✓

- Install latest python software from python.org — IBM38-6 — ✓ 10
- Create accounts in IBM cloud, IBM IOT, IBM Watson platforms — IBM38-7 — ✓ 10
- Create device in IBM Watson and note down the credentials — IBM38-8 — ✓ 10
- Develop a python program to conect to IBM Watson device — IBM38-9 — ✓ 10



All sprints

Sprint 4 · Clear filters

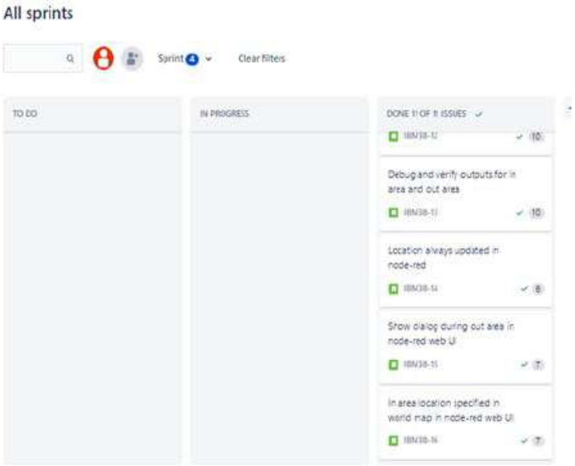
TO DO 3 OF 3 ISSUES

- Location always updated in node-red — IBM38-14 — 6
- Show dialog during out area in node-red web UI — IBM38-15 — 7
- In area location specified in world map in node-red web UI — IBM38-16 — 7

IN PROGRESS

DONE 8 OF 8 ISSUES ✓

- Install latest python software from python.org — IBM38-6 — ✓ 10
- Create accounts in IBM cloud, IBM IOT, IBM Watson platforms — IBM38-7 — ✓ 10
- Create device in IBM Watson and note down the credentials — IBM38-8 — ✓ 10
- Develop a python program to conect to IBM Watson device — IBM38-9 — ✓ 10
- Install needed libraries and connect required nodes



All sprints

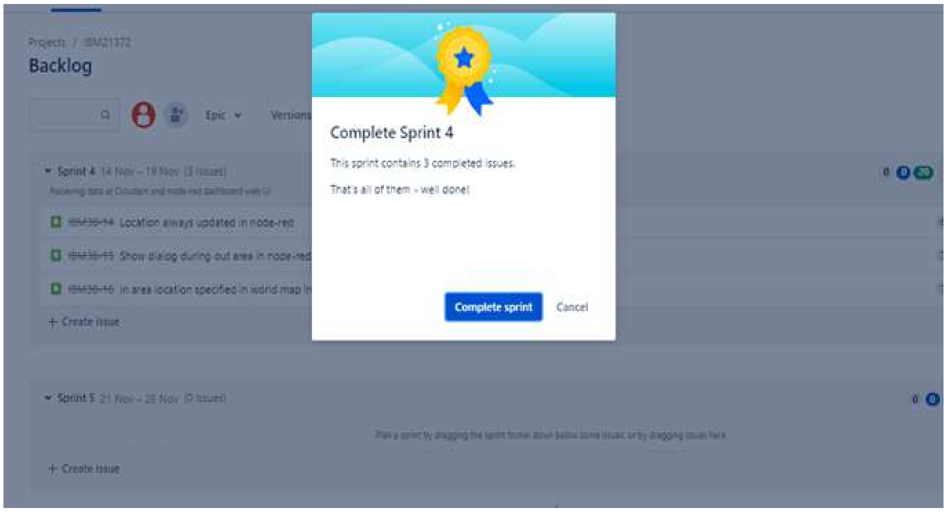
Sprint 4 · Clear filters

TO DO

IN PROGRESS

DONE 11 OF 11 ISSUES ✓

- IBM38-12 — ✓ 10
- Debug and verify outputs for in area and out area — IBM38-11 — ✓ 10
- Location always updated in node-red — IBM38-14 — ✓ 6
- Show dialog during out area in node-red web UI — IBM38-15 — ✓ 7
- In area location specified in world map in node-red web UI — IBM38-16 — ✓ 7



Projects / IBM21372

Backlog

Epic · Versions

Sprint 4 14 Nov – 19 Nov (3 issues)

Recieving data at Cloudant and node-red dashboard web UI

- ~~IBM38-14~~ Location always updated in node-red
- ~~IBM38-15~~ Show dialog during out area in node-red
- ~~IBM38-16~~ In area location specified in world map in

+ Create issue

Sprint 5 21 Nov – 26 Nov (0 issues)

Plan a sprint by dragging the sprint footer down below some issues, or by dragging issues here.

+ Create issue

Complete Sprint 4

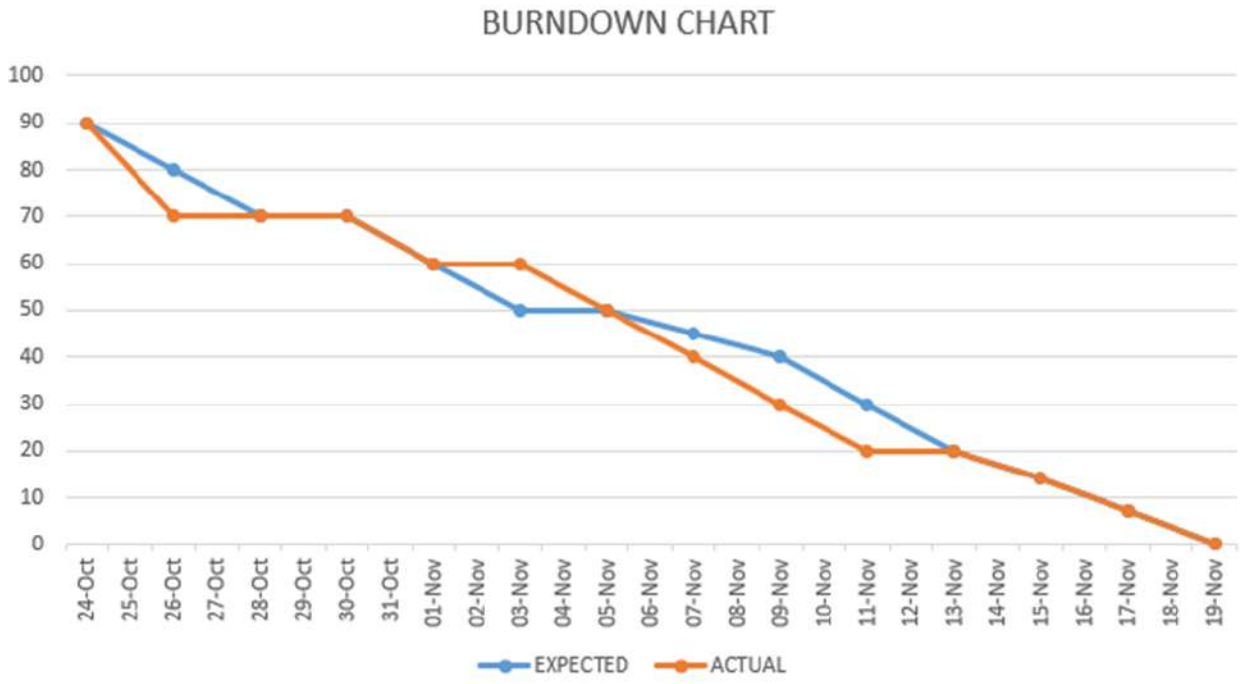
This sprint contains 3 completed issues.

That's all of them - well done!

Complete sprint · Cancel

Sprint Burndown Chart

A burndown chart shows the amount of work that has been completed in an epic or sprint, and the total work remaining. Burndown charts are used to predict your team's likelihood of completing their work in the time available. They're also great for keeping the team aware of any scope creep that occurs**.**



7.CODING AND SOLUTIONING

7.1 Feature 1 - PYTHON SCRIPT

Sprint 1 :

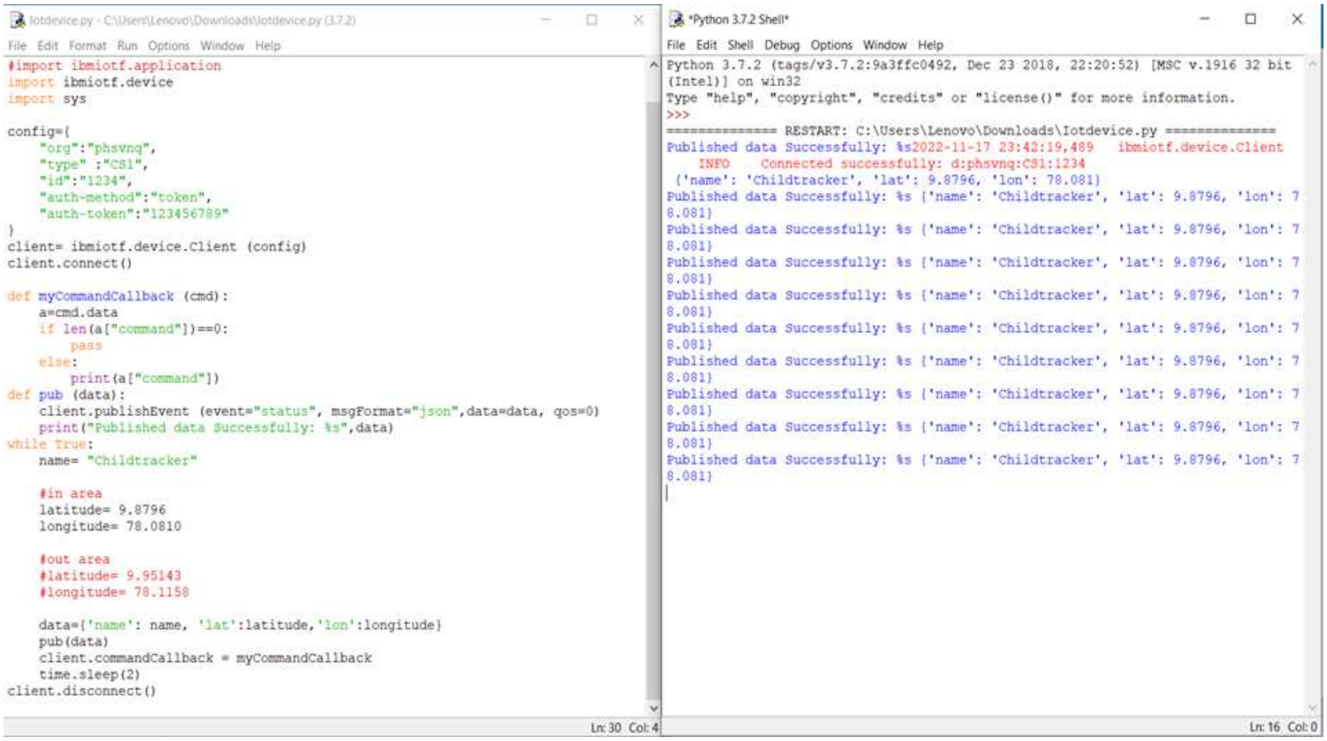
Develop a python script and connect it with IBM Watson device to send the location specified in code

Python code function is to constantly update the location of the child and provide the information to IBM IOT platform.

CODE

```
import time

import random

#import ibmiotf.application

import ibmiotf.device

import sys

config={

    "org":"phsvnq",

    "type" :"CS1",

    "id":"1234",

    "auth-method":"token",

    "auth-token":"123456789"

}

client= ibmiotf.device.Client (config)

client.connect()
```

```
def myCommandCallback (cmd):
          a=cmd.data
          if len(a["command"])==0:
          pass
          else:
        print(a["command"])
def pub (data):
    client.publishEvent (event="status", msgFormat="json",data=data, qos=0)
    print("Published data Successfully: %s",data)
while True:
          name= "Childtracker"

          #in area
          latitude= 9.8796
          longitude= 78.0810

          #out area
          #latitude= 9.95143
          #longitude= 78.1158
        data={'name': name, 'lat':latitude,'lon':longitude}
            pub(data)
            client.commandCallback = myCommandCallback
        time.sleep(2)
client.disconnect()
```


Iotdevice.py - C:\Users\Lenovo\Downloads\Iotdevice.py (3.7.2)
File Edit Format Run Options Window Help
Ln: 30 Col: 4
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Ln: 16 Col: 0

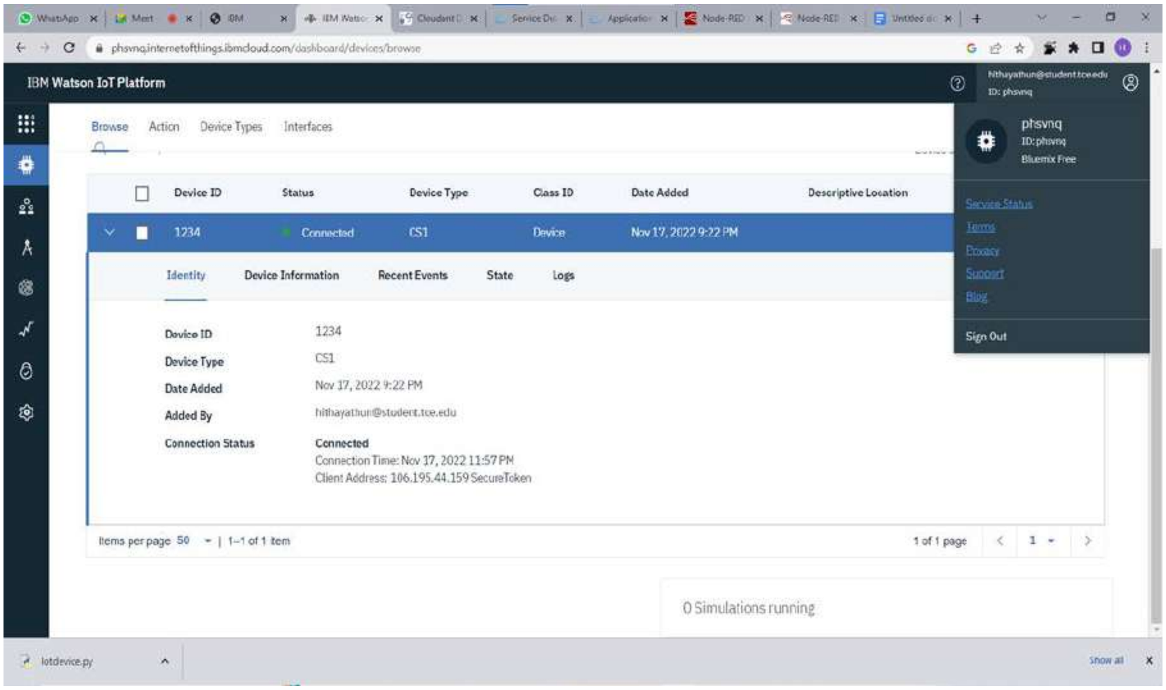
```
#import ibmiotf.application
import ibmiotf.device
import sys

config={
    "org":"phsvnq",
    "type" :"CS1",
    "id":"1234",
    "auth-method":"token",
    "auth-token":"123456789"
}
client= ibmiotf.device.Client (config)
client.connect()

def myCommandCallback (cmd):
    a=cmd.data
    if len(a["command"])==0:
        pass
    else:
        print(a["command"])
def pub (data):
    client.publishEvent (event="status", msgFormat="json",data=data, qos=0)
    print("Published data Successfully: %s",data)
while True:
    name= "Childtracker"

    #in area
    latitude= 9.8796
    longitude= 78.0810

    #out area
    #latitude= 9.95143
    #longitude= 78.1158

    data={'name': name, 'lat':latitude,'lon':longitude}
    pub(data)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============== RESTART: C:\Users\Lenovo\Downloads\Iotdevice.py ==============
Published data Successfully: %s2022-11-17 23:42:19,489   ibmiotf.device.Client
    INFO    Connected successfully: d:phsvnq:CS1:1234
 {'name': 'Childtracker', 'lat': 9.8796, 'lon': 78.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
```

7.2 Feature 2 - IBM WATSON AND IBM CLOUD

Sprint 2 :

Creating device in IBM Watson IOT platform

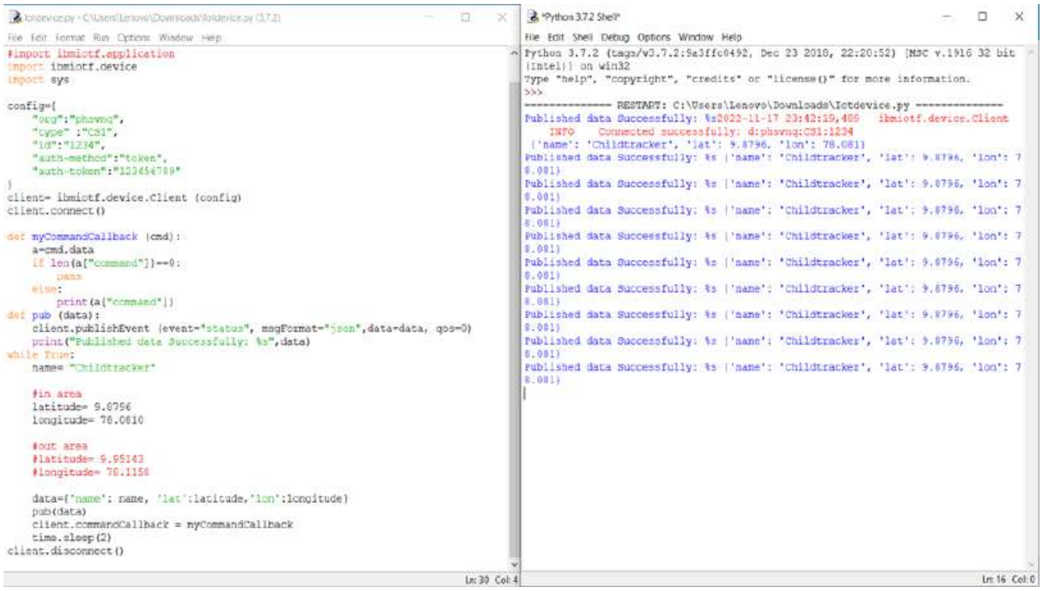


Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
1234	Connected	CS1	Device	Nov 17, 2022 9:22 PM	

Device ID 1234
Device Type CS1
Date Added Nov 17, 2022 9:22 PM
Added By hithayathun@student.tce.edu
Connection Status Connected
Connection Time: Nov 17, 2022 11:57 PM
Client Address: 106.195.44.159 SecureToken

Connecting IBM Watson and python code

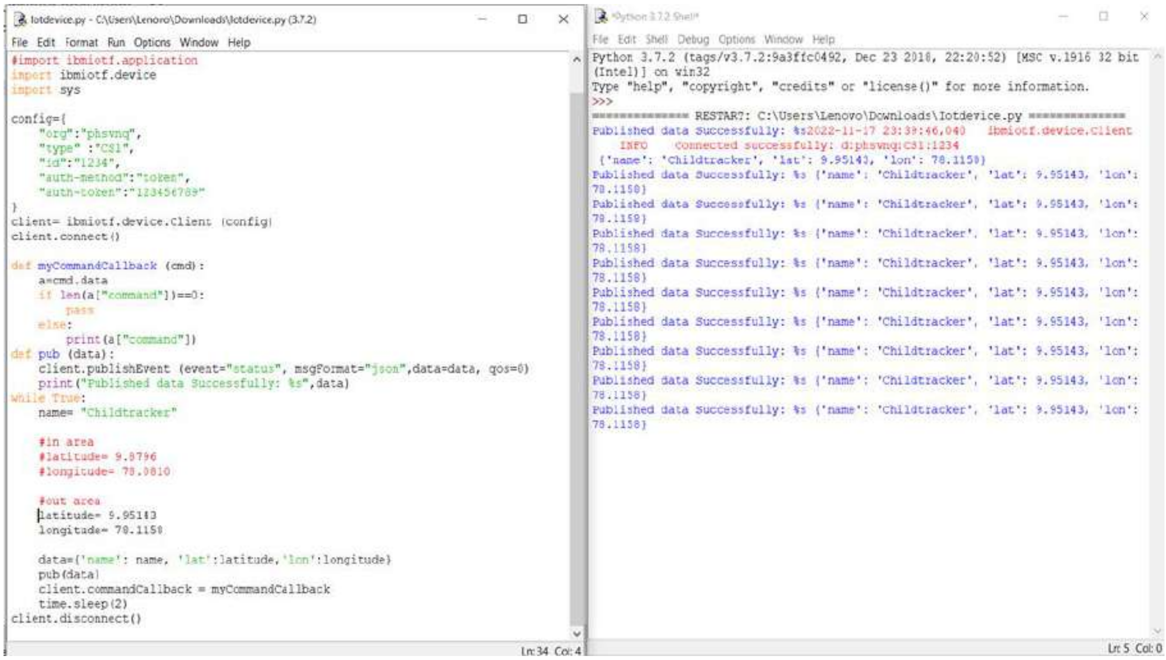
In area Location


Iotdevice.py - C:\Users\Lenovo\Downloads\Iotdevice.py (3.7.2)
File Edit Format Run Options Window Help
Ln: 30 Col: 4
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Ln: 16 Col: 0

```
#import ibmiotf.application
import ibmiotf.device
import sys

config={
    "org":"phsvnq",
    "type" :"CS1",
    "id":"1234",
    "auth-method":"token",
    "auth-token":"123456789"
}
client= ibmiotf.device.Client (config)
client.connect()

def myCommandCallback (cmd):
    a=cmd.data
    if len(a["command"])==0:
        pass
    else:
        print(a["command"])
def pub (data):
    client.publishEvent (event="status", msgFormat="json",data=data, qos=0)
    print("Published data Successfully: %s",data)
while True:
    name= "Childtracker"

    #in area
    latitude= 9.8796
    longitude= 78.0810

    #out area
    #latitude= 9.95143
    #longitude= 78.1158

    data={'name': name, 'lat':latitude,'lon':longitude}
    pub(data)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============== RESTART: C:\Users\Lenovo\Downloads\Iotdevice.py ==============
Published data Successfully: %s2022-11-17 23:42:19,409   ibmiotf.device.Client
    INFO    Connected successfully: d:phsvnq:CS1:1234
 {'name': 'Childtracker', 'lat': 9.8796, 'lon': 78.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.8796, 'lon': 7
8.081}
```

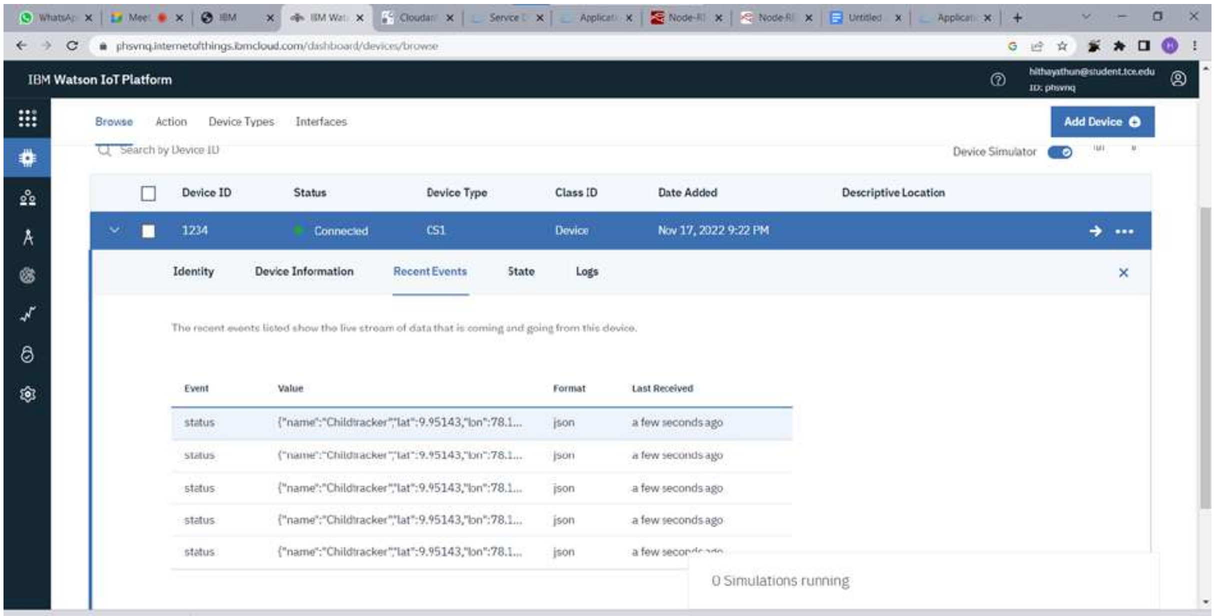
Out Area Location


Iotdevice.py - C:\Users\Lenovo\Downloads\Iotdevice.py (3.7.2)
File Edit Format Run Options Window Help
Ln: 34 Col: 4
Python 3.7.2 Shell
File Edit Shell Debug Options Window Help
Ln: 5 Col: 0

```
#import ibmiotf.application
import ibmiotf.device
import sys

config={
    "org":"phsvnq",
    "type" :"CS1",
    "id":"1234",
    "auth-method":"token",
    "auth-token":"123456789"
}
client= ibmiotf.device.Client (config)
client.connect()

def myCommandCallback (cmd):
    a=cmd.data
    if len(a["command"])==0:
        pass
    else:
        print(a["command"])
def pub (data):
    client.publishEvent (event="status", msgFormat="json",data=data, qos=0)
    print("Published data Successfully: %s",data)
while True:
    name= "Childtracker"

    #in area
    #latitude= 9.8796
    #longitude= 78.0810

    #out area
    latitude= 9.95143
    longitude= 78.1158

    data={'name': name, 'lat':latitude,'lon':longitude}
    pub(data)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

```
Python 3.7.2 (tags/v3.7.2:9a3ffc0492, Dec 23 2018, 22:20:52) [MSC v.1916 32 bit
(Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
============== RESTART: C:\Users\Lenovo\Downloads\Iotdevice.py ==============
Published data Successfully: %s2022-11-17 23:39:46,040   ibmiotf.device.Client
    INFO    Connected successfully: d:phsvnq:CS1:1234
 {'name': 'Childtracker', 'lat': 9.95143, 'lon': 78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
Published data Successfully: %s {'name': 'Childtracker', 'lat': 9.95143, 'lon':
78.1158}
```

Connecting python code to IBM Watson device



	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
□	1234	Connected	CS1	Device	Nov 17, 2022 9:22 PM	

The recent events listed show the live stream of data that is coming and going from this device.

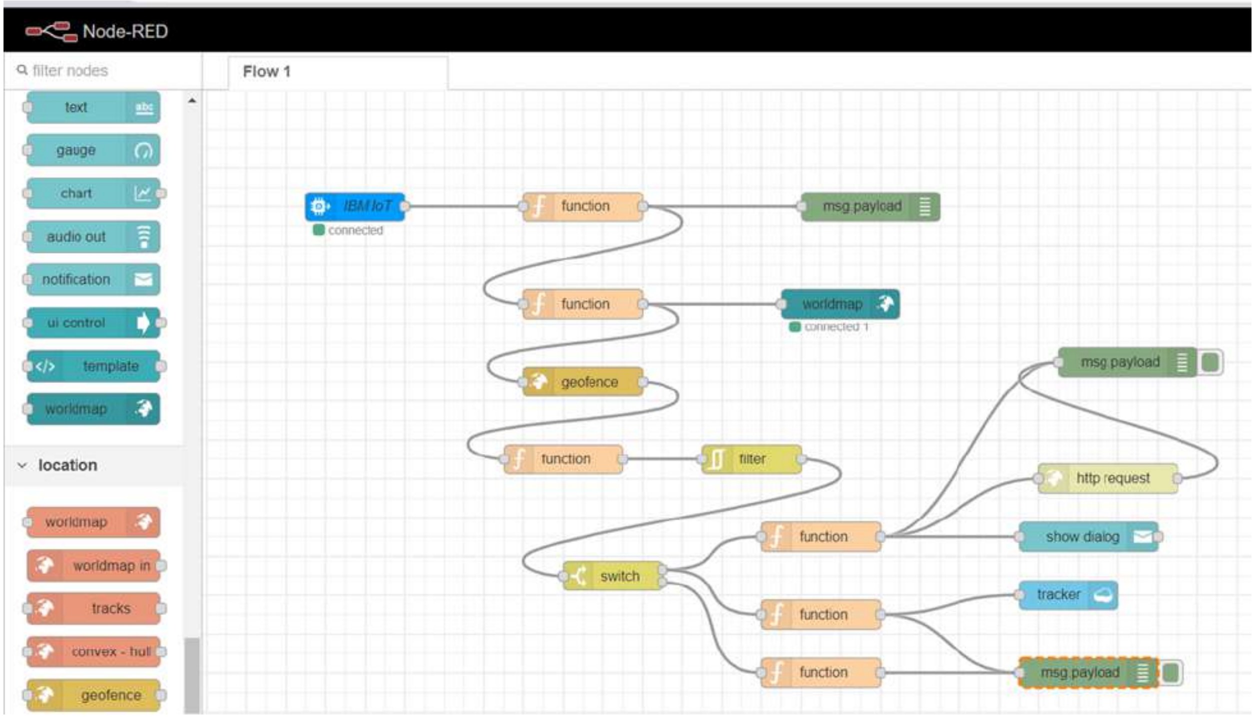
Event	Value	Format	Last Received
status	{"name":"Childtracker","lat":9.95143,"lon":78.1...	json	a few seconds ago
status	{"name":"Childtracker","lat":9.95143,"lon":78.1...	json	a few seconds ago
status	{"name":"Childtracker","lat":9.95143,"lon":78.1...	json	a few seconds ago
status	{"name":"Childtracker","lat":9.95143,"lon":78.1...	json	a few seconds ago
status	{"name":"Childtracker","lat":9.95143,"lon":78.1...	json	a few secon

7.3 Feature 3- NODE RED

Sprint 3 :CREATING NOD RED SERVICE AND CONNECTING WITH IBM

Nodes Used in the project:

1. IBM IOT
2. Function
3. Msg.payload
4. Switch
5. Filter
6. Geofence
7. Show dialog
8. Tracker-Cloudant db
9. http request

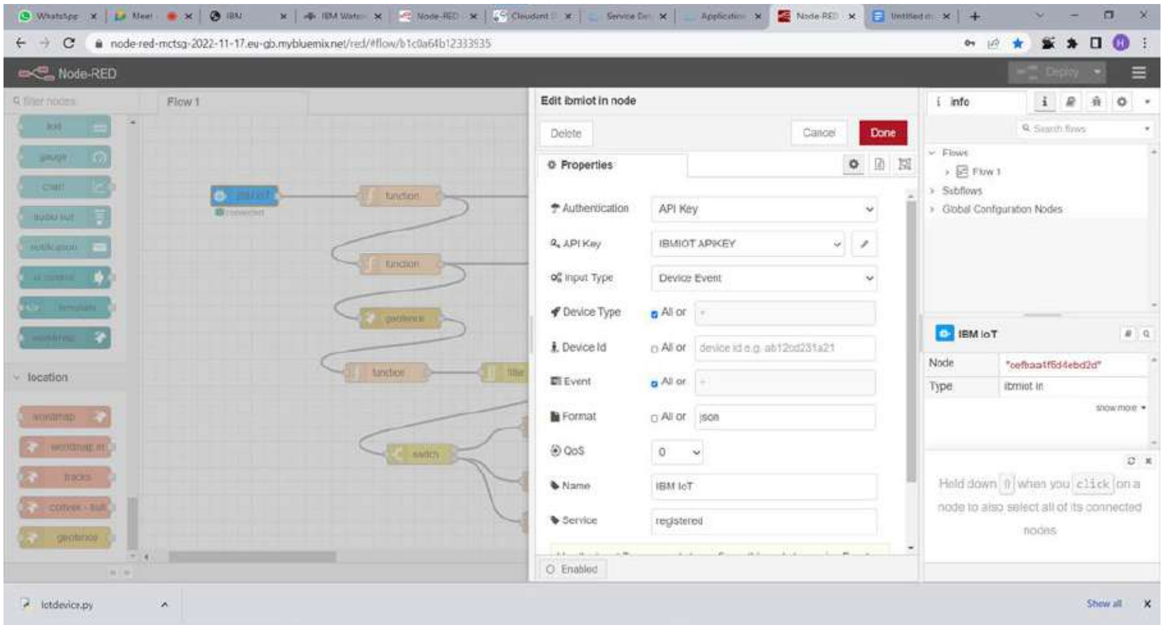


Now we need to format each node in NODE RED in on messages to do their specific function

Additionally we need to configure and connect world map and IBM IOT with NODE RED

Codes in Each Node :

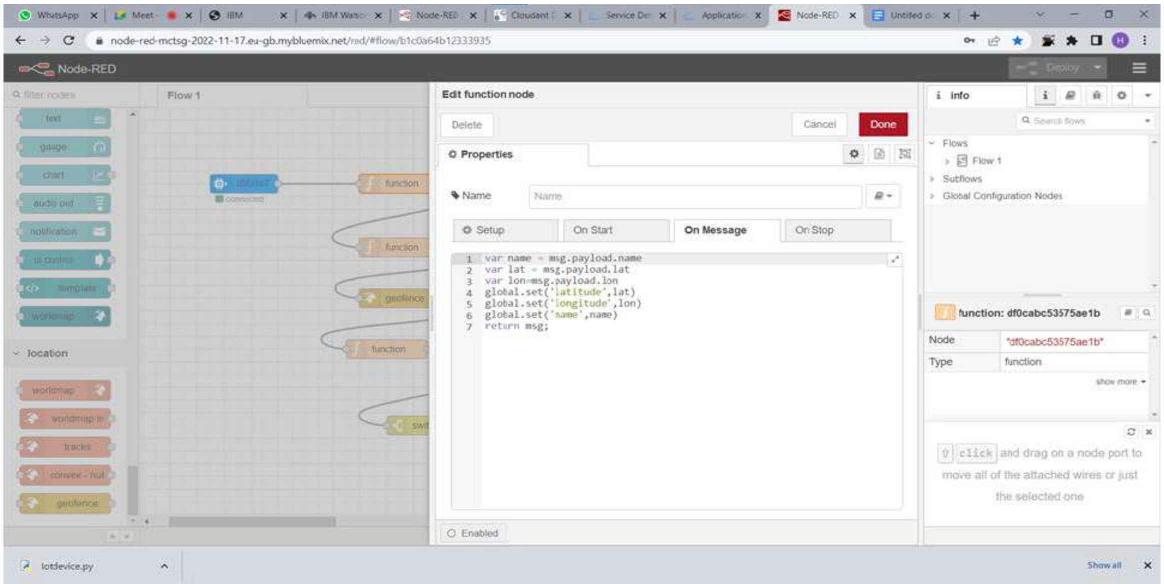
IBM Iot:



Edit ibmiot in node

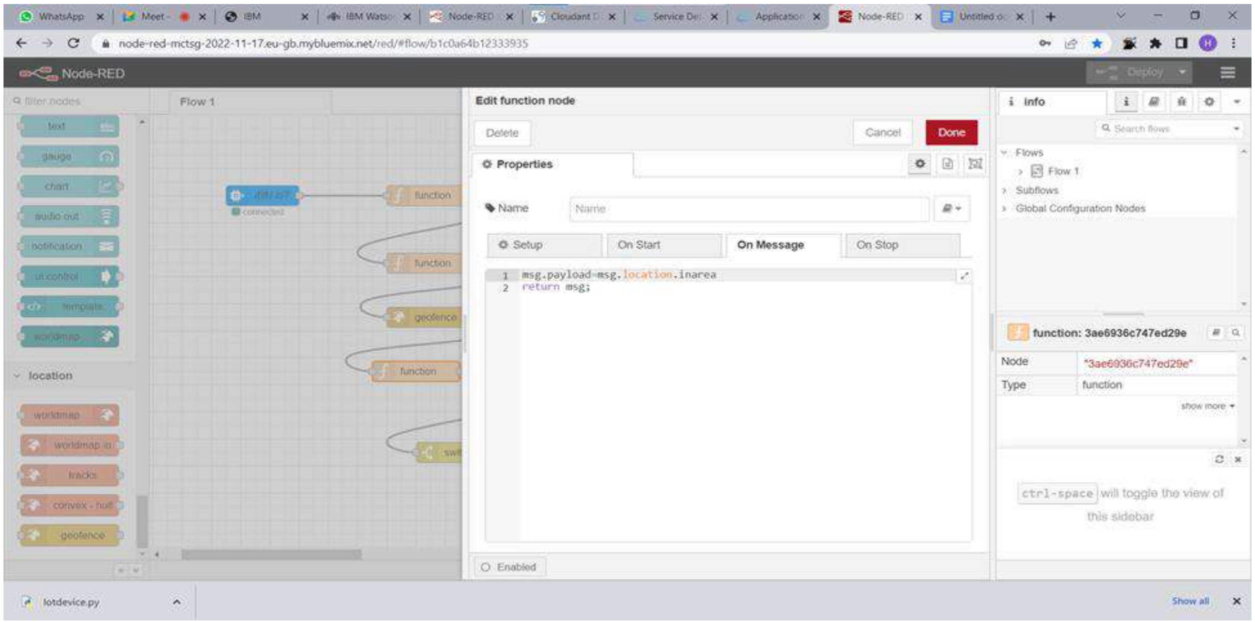
Authentication: API Key
API Key: IBMIOT APIKEY
Input Type: Device Event
Device Type: ☑ All or
Device Id: □ All or device id e.g. ab12cd231a21
Event: ☑ All or
Format: □ All or json
QoS: 0
Name: IBM IoT
Service: registered

Function nodes:

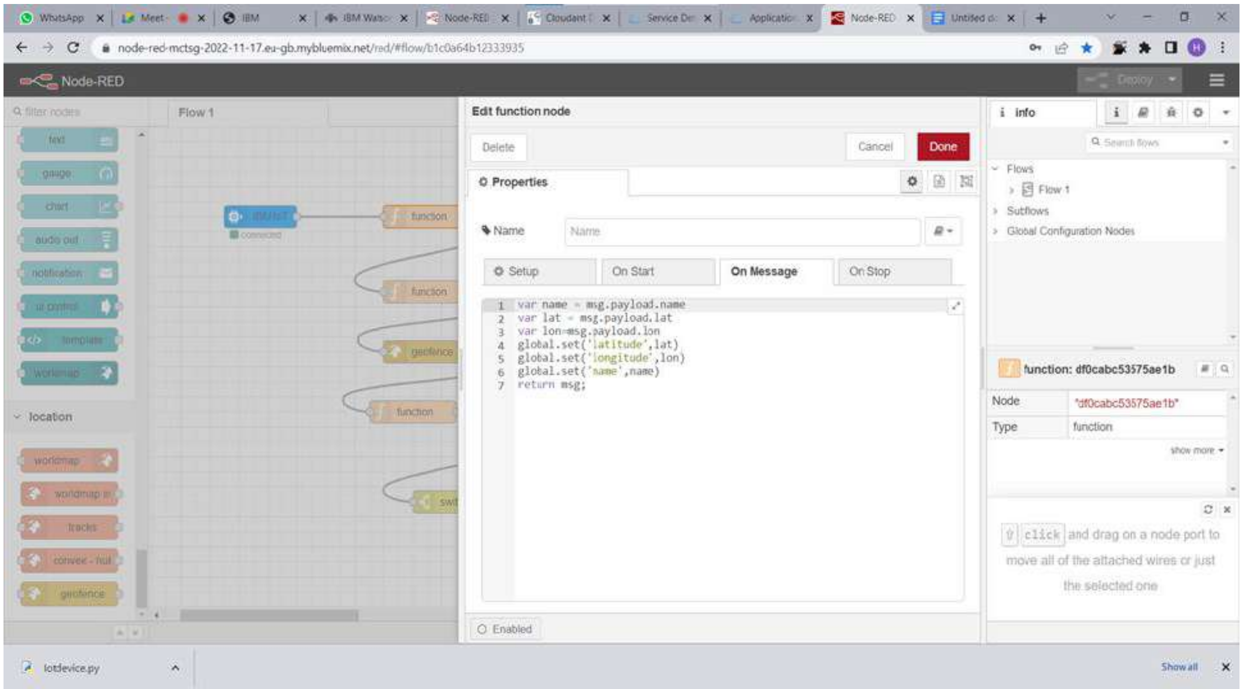


Edit function node

```
var name = msg.payload.name
var lat = msg.payload.lat
var lon=msg.payload.lon
global.set('latitude',lat)
global.set('longitude',lon)
global.set('name',name)
return msg;
```

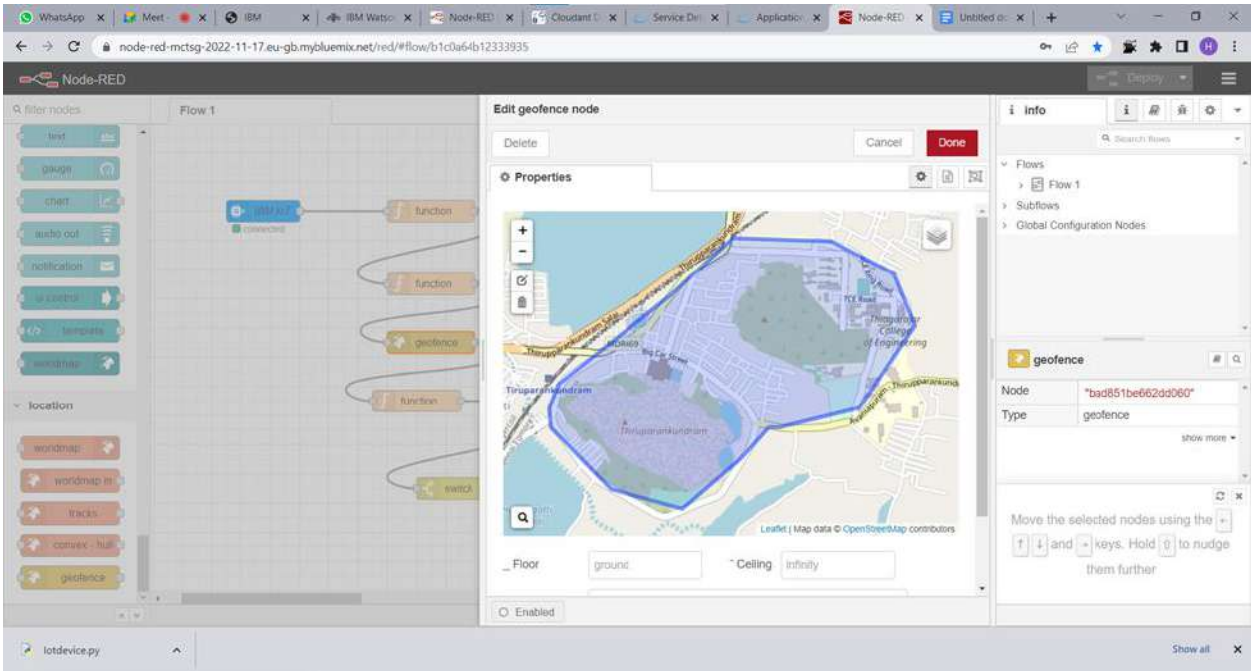


```
msg.payload=msg.location.inarea
return msg;
```


node-red-mctsg-2022-11-17.eu-gb.mybluemix.net/red/#flow/b1c0a64b12333935
Node-RED
Deploy
Flow 1
Edit function node
Delete
Cancel
Done
Properties
Name
Setup
On Start
On Message
On Stop
Enabled
info
Flows
Flow 1
Subflows
Global Configuration Nodes
function: df0cabc53575ae1b
Node "df0cabc53575ae1b"
Type function
click and drag on a node port to move all of the attached wires or just the selected one
location
Iotdevice.py
Show all

```
var name = msg.payload.name
var lat = msg.payload.lat
var lon=msg.payload.lon
global.set('latitude',lat)
global.set('longitude',lon)
global.set('name',name)
return msg;
```

Geofence node:



node-red-mctsg-2022-11-17.eu-gb.mybluemix.net/red/#flow/b1c0a64b12333935

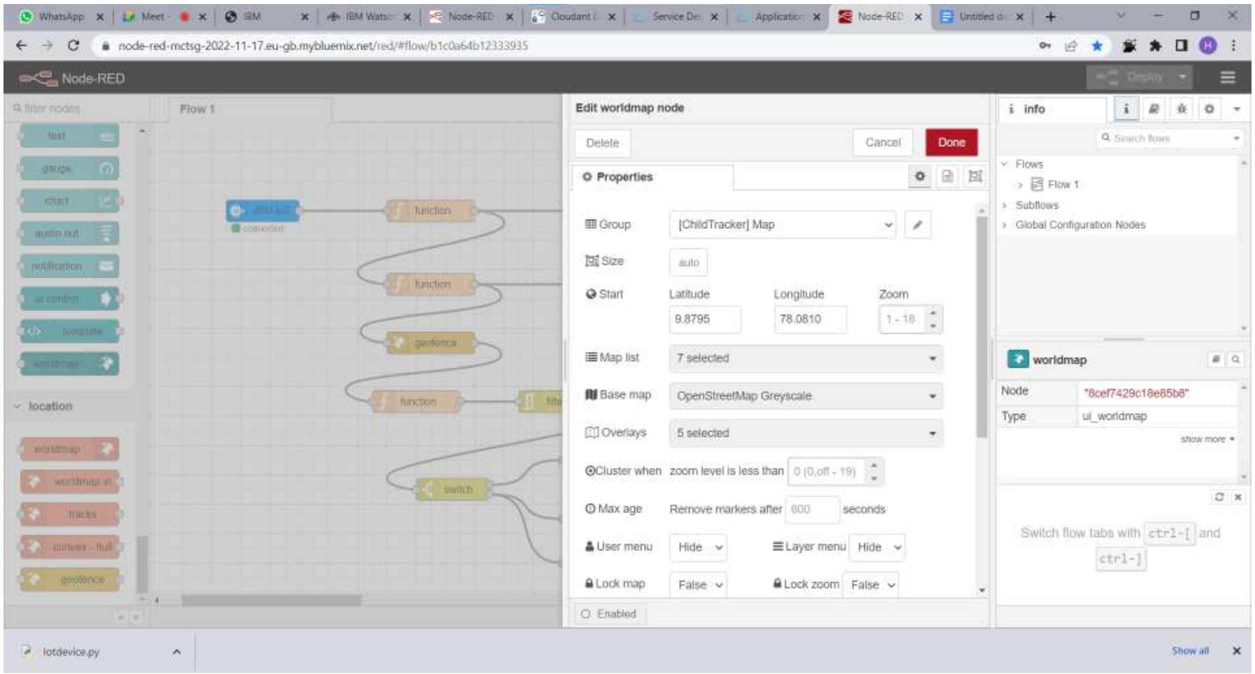
Edit geofence node

Floor: ground
Ceiling: infinity

Node: "bad851be662dd060"
Type: geofence

Node-RED
Flow 1
Delete
Cancel
Done
Properties
Leaflet | Map data © OpenStreetMap contributors
Enabled
info
Flows
Subflows
Global Configuration Nodes
geofence
Move the selected nodes using the ← ↑ ↓ and → keys. Hold ⇧ to nudge them further
Iotdevice.py
Show all

Worldmap node:



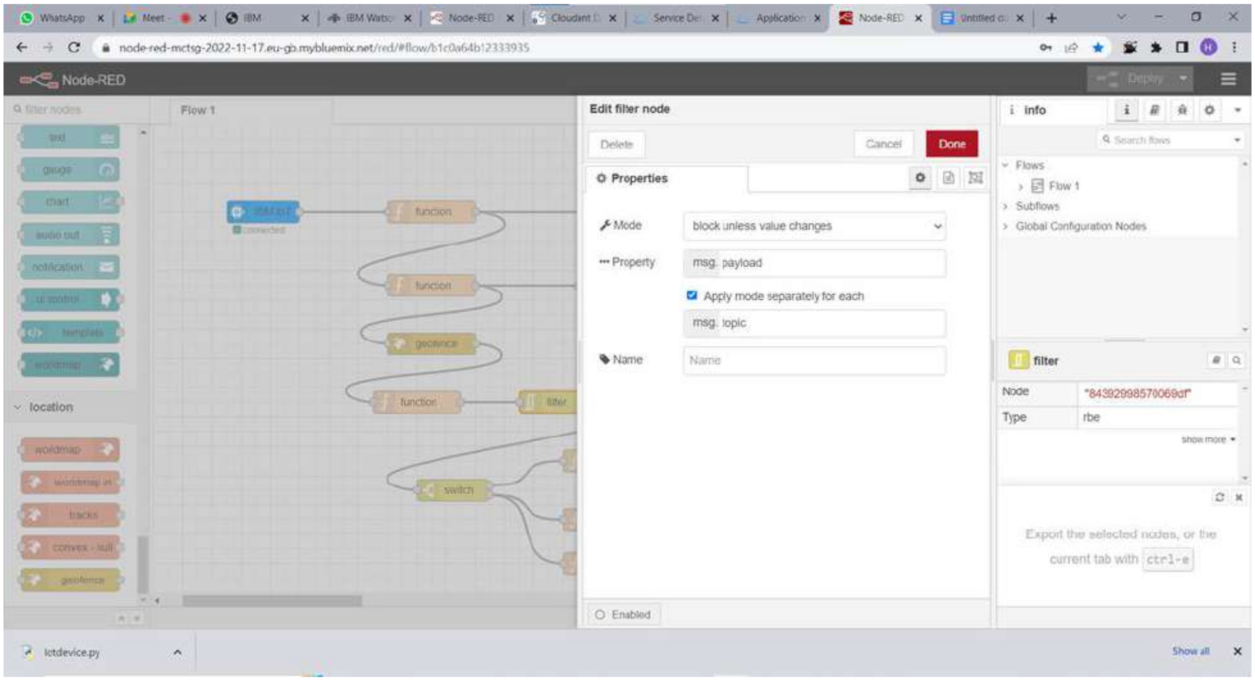
node-red-mctsg-2022-11-17.eu-gb.mybluemix.net/red/#flow/b1c0a64b12333935

Edit worldmap node

Group: [ChildTracker] Map
Size: auto
Start: Latitude 9.8795, Longitude 78.0810, Zoom 1 - 18
Map list: 7 selected
Base map: OpenStreetMap Greyscale
Overlays: 5 selected
Cluster when: zoom level is less than 0 (0,off - 19)
Max age: Remove markers after 600 seconds
User menu: Hide
Layer menu: Hide
Lock map: False
Lock zoom: False

Node: "8cef7429c18e85b8"
Type: ui_worldmap

Filter node:



Edit filter node

Mode: block unless value changes

Property: msg. payload

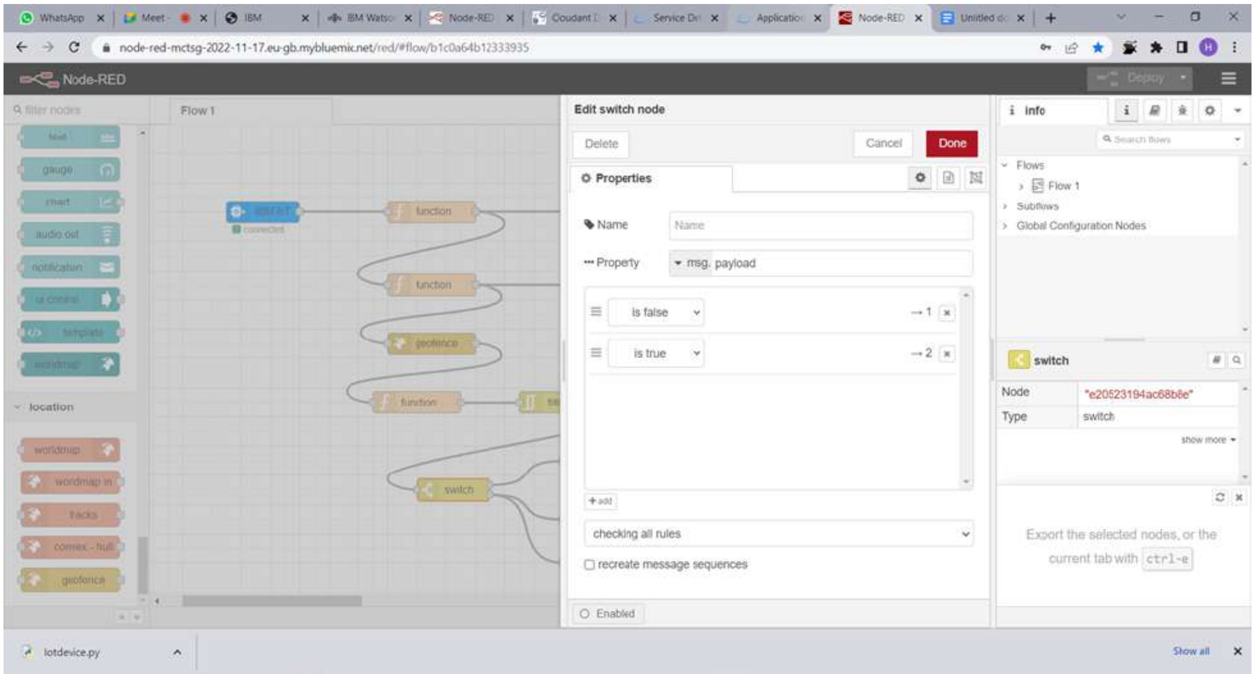
☑ Apply mode separately for each msg. topic

Name: Name

○ Enabled

Node	"84392998570069df"
Type	rbe

Switch node:



Edit switch node

Name: Name

Property: msg. payload

is false → 1

is true → 2

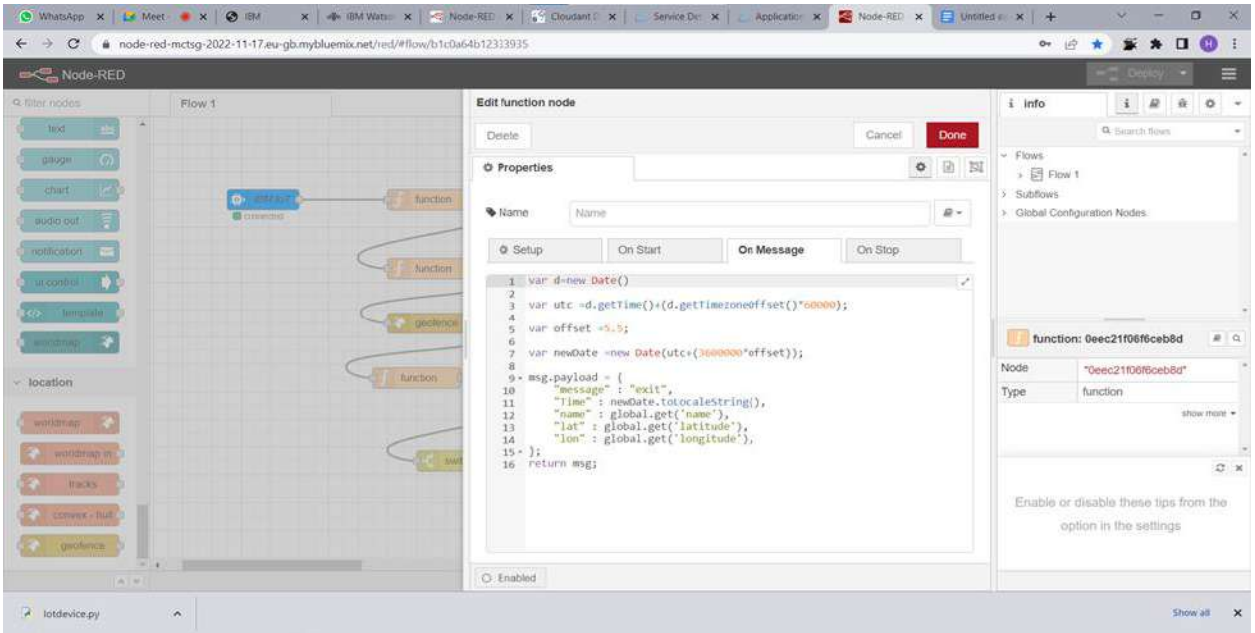
checking all rules

☐ recreate message sequences

○ Enabled

Node	"e20523194ac68b8e"
Type	switch

Function node:

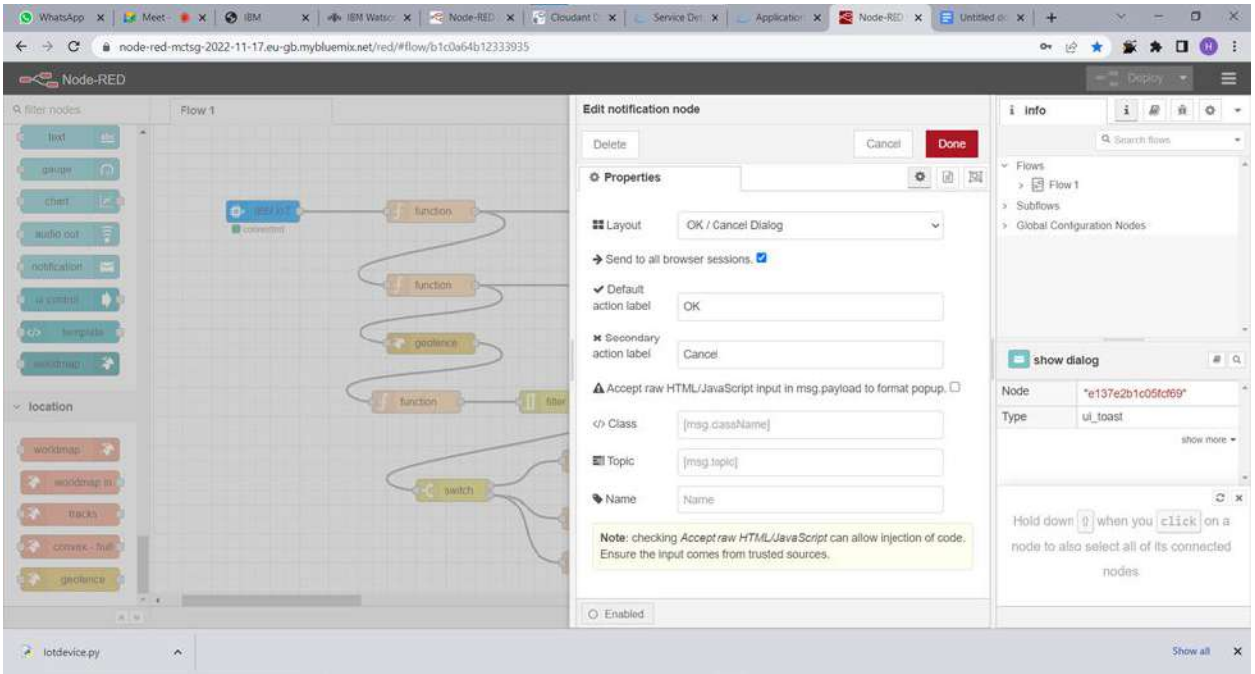


Name: Name

```
var d=new Date()

var utc =d.getTime()+(d.getTimezoneOffset()*60000);

var offset =5.5;

var newDate =new Date(utc+(3600000*offset));

msg.payload = {
    "message" : "exit",
    "Time" : newDate.toLocaleString(),
    "name" : global.get('name'),
    "lat" : global.get('latitude'),
    "lon" : global.get('longitude'),
};
return msg;
```

function: 0eec21f06f6ceb8d

Node	"0eec21f06f6ceb8d"
Type	function



Layout: OK / Cancel Dialog

Send to all browser sessions. ☑

Default action label: OK

Secondary action label: Cancel

Accept raw HTML/JavaScript input in msg.payload to format popup. □

Class: [msg.className]

Topic: [msg.topic]

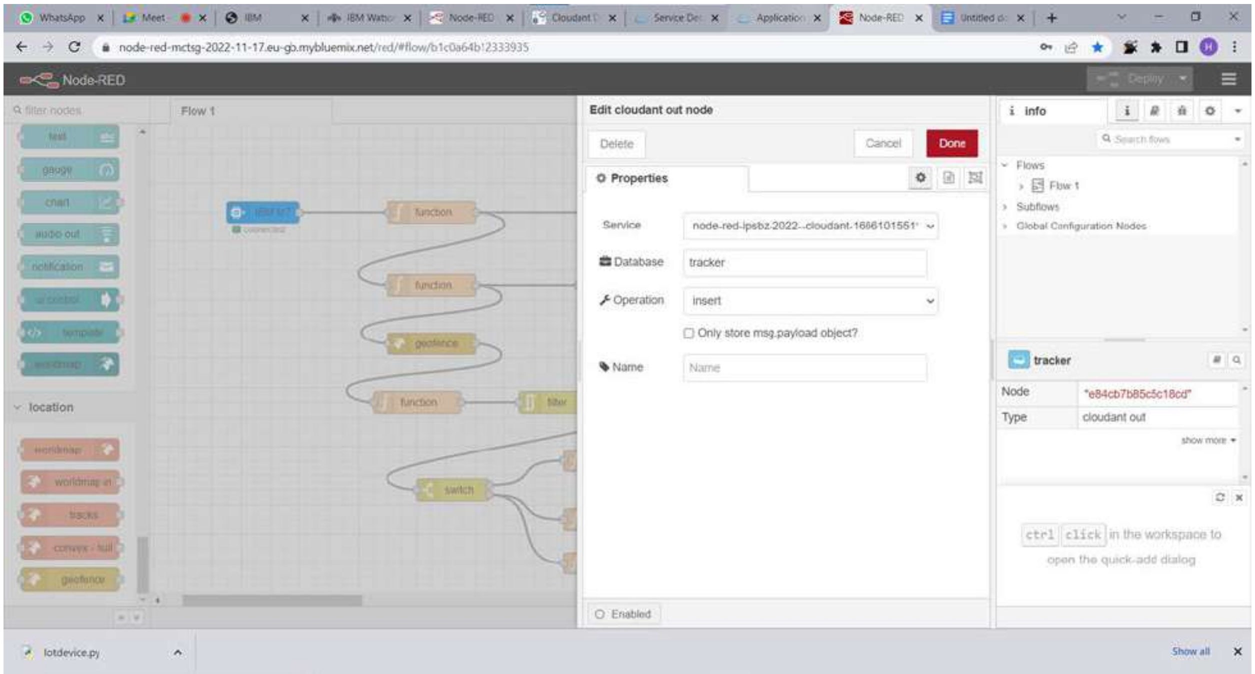
Name: Name

Note: checking *Accept raw HTML/JavaScript* can allow injection of code. Ensure the input comes from trusted sources.

show dialog

Node	"e137e2b1c05fcf69"
Type	ui_toast

Cloudant out node:



Edit cloudant out node

Service: node-red-ipsbz.2022...cloudant-1666101551

Database: tracker

Operation: insert

□ Only store msg.payload object?

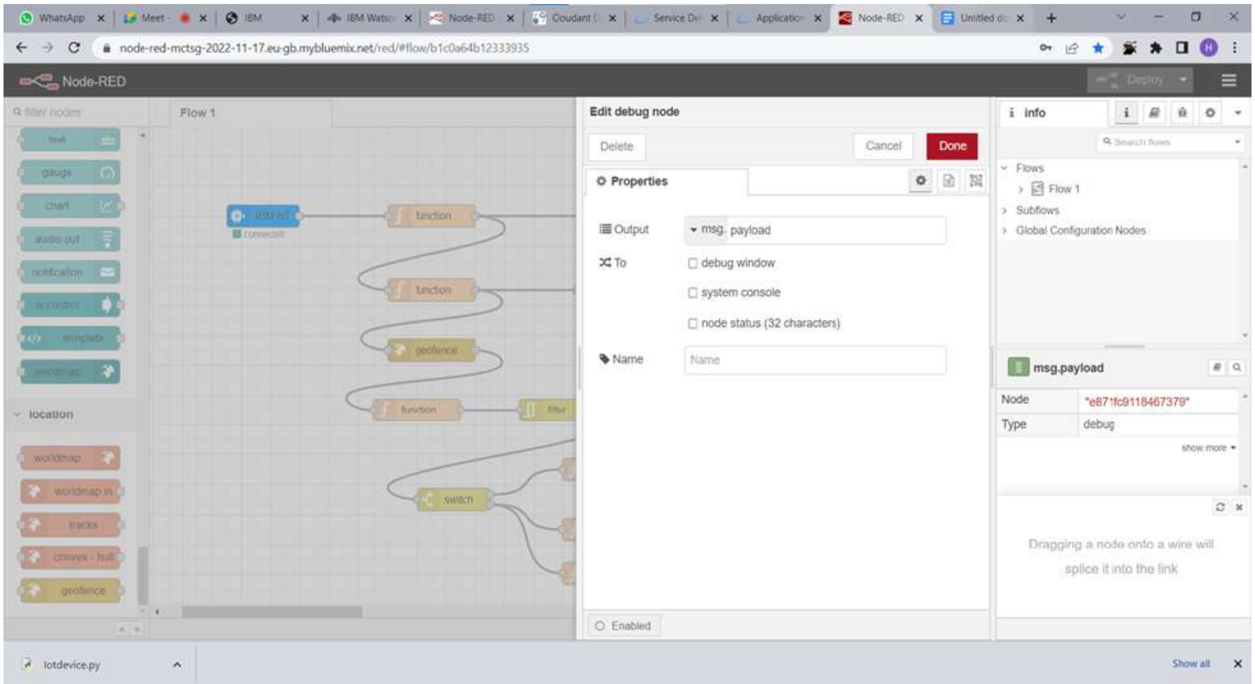
Name: Name

tracker

Node	"e84cb7b85c5c18cd"
Type	cloudant out

Node-RED
node-red-mctsg-2022-11-17.eu-gb.mybluemix.net/red/#flow/b1c0a64b12333935
Flow 1
Delete
Cancel
Done
Properties
Enabled
info
Flows
Flow 1
Subflows
Global Configuration Nodes
show more
ctrl click in the workspace to open the quick-add dialog
location
function
geofence
filter
switch
Iotdevice.py
Show all

Debug node:



Edit debug node

Output: msg. payload

To: □ debug window
□ system console
□ node status (32 characters)

Name: Name

msg.payload

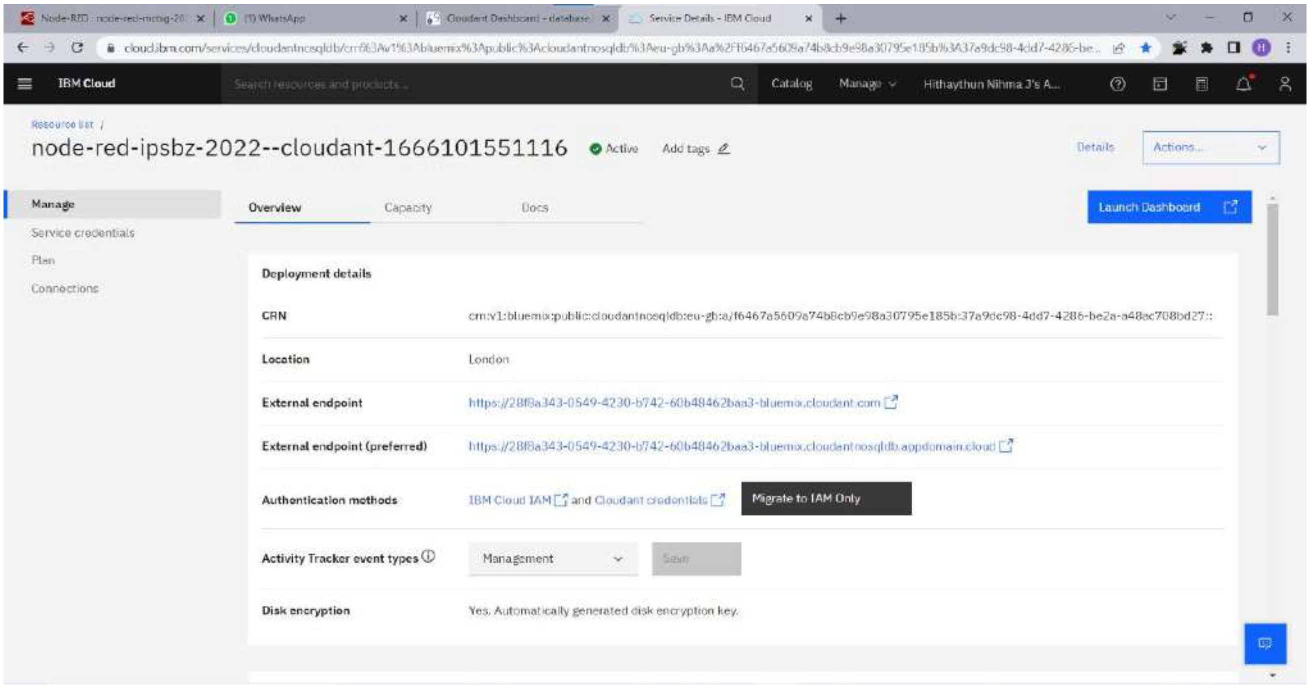
Node	"e871fc9118467379"
Type	debug

Node-RED
node-red-mctsg-2022-11-17.eu-gb.mybluemix.net/red/#flow/b1c0a64b12333935
Flow 1
Delete
Cancel
Done
Properties
Enabled
info
Flows
Flow 1
Subflows
Global Configuration Nodes
show more
Dragging a node onto a wire will splice it into the link
location
function
geofence
filter
switch
Iotdevice.py
Show all

7.4 Feature 4- CLOUDANT DB

Sprint 4 :OUTPUT IMAGES AND DATA STORED IN CLOUDANT

Create Cloudant DB:

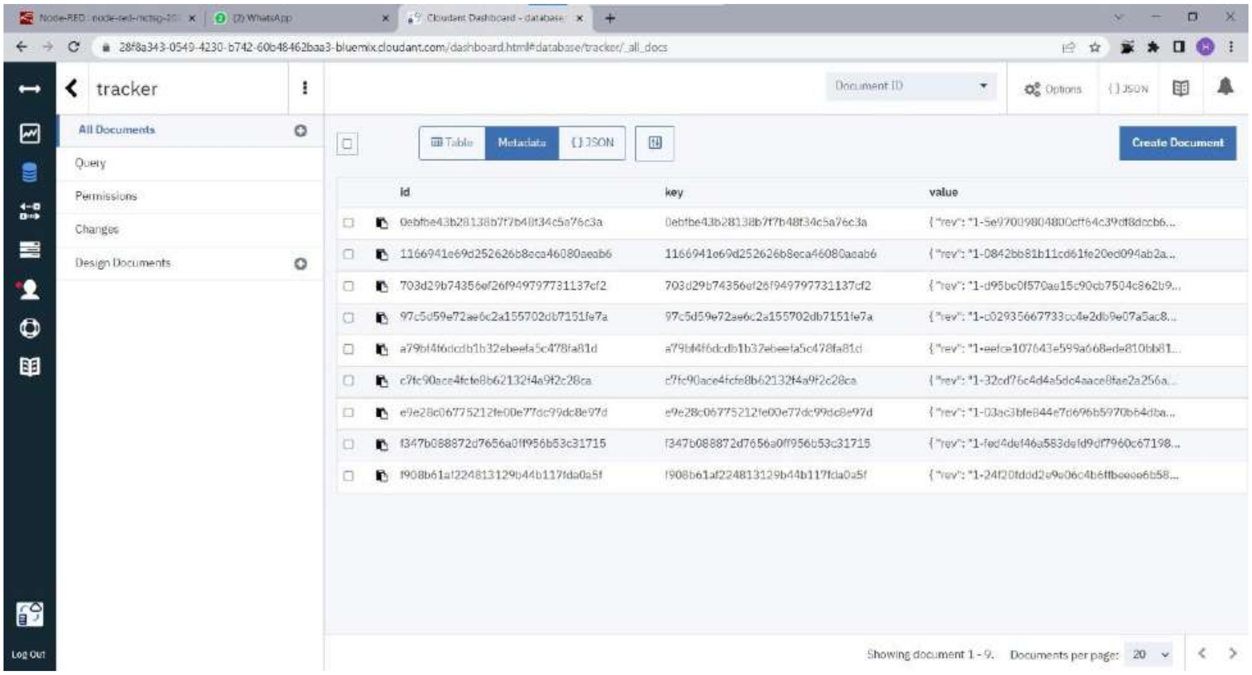


Node-RED : node-red-ncbg-20
(1) WhatsApp
Cloudant Dashboard - database
Service Details - IBM Cloud
cloud.ibm.com/services/cloudantnosqldb/crn%3Av1%3Abluemix%3Apublic%3Acloudantnosqldb%3Aeu-gb%3Aa%2Ff6467a5609a74b8cb9e98a30795e185b%3A37a9dc98-4dd7-4286-be...
IBM Cloud
Search resources and products...
Catalog
Manage
Hithaythun Nihma J's A...
Resource list /
Details
Actions...
Manage
Service credentials
Plan
Connections
Overview
Capacity
Docs
Launch Dashboard

node-red-ipsbz-2022--cloudant-1666101551116 ● Active Add tags

Deployment details

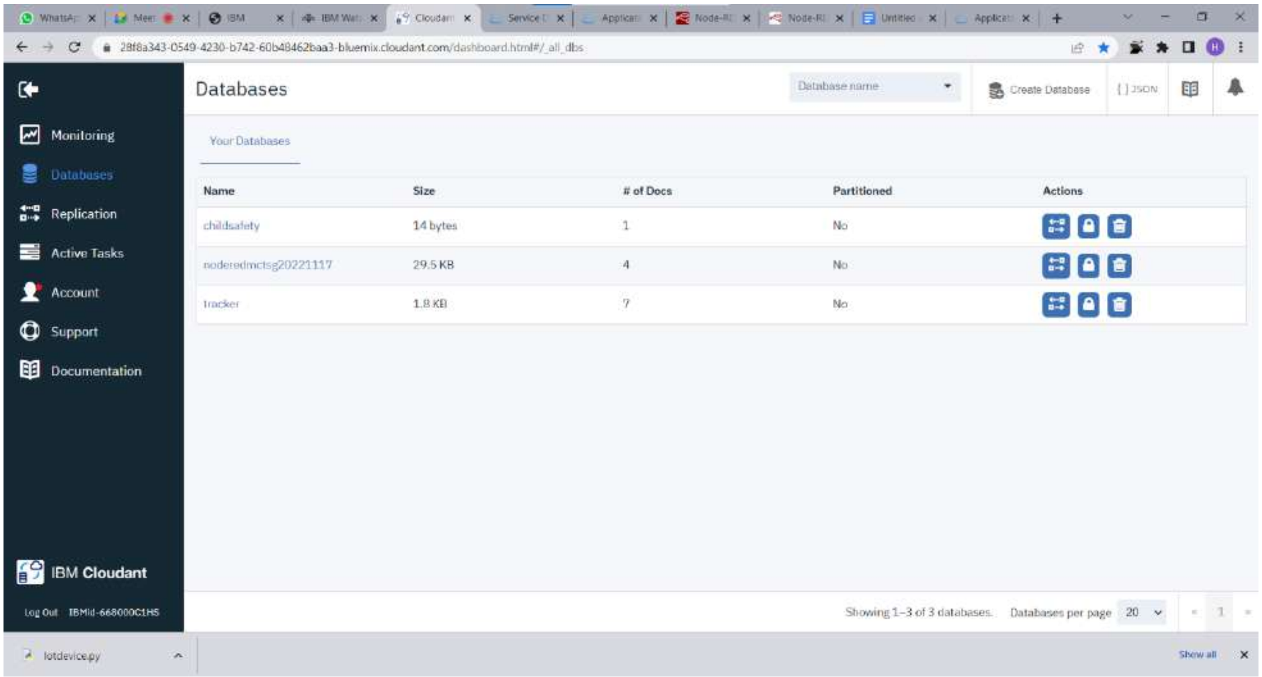
CRN	crn:v1:bluemix:public:cloudantnosqldb:eu-gb:a/f6467a5609a74b8cb9e98a30795e185b:37a9dc98-4dd7-4286-be2a-a48ac708bd27::
Location	London
External endpoint	https://28f8a343-0549-4230-b742-60b48462baa3-bluemix.cloudant.com
External endpoint (preferred)	https://28f8a343-0549-4230-b742-60b48462baa3-bluemix.cloudantnosqldb.appdomain.cloud
Authentication methods	IBM Cloud IAM and Cloudant credentials Migrate to IAM Only
Activity Tracker event types	Management Save
Disk encryption	Yes. Automatically generated disk encryption key.



Node-RED : node-red-ncbg-20
(2) WhatsApp
Cloudant Dashboard - database
28f8a343-0549-4230-b742-60b48462baa3-bluemix.cloudant.com/dashboard.html#database/tracker/_all_docs
tracker
All Documents
Query
Permissions
Changes
Design Documents
Log Out
Document ID
Options
{ } JSON
Table
Metadata
{ } JSON
Create Document

	id	key	value
□	0ebfbe43b28138b7f7b48f34c5a76c3a	0ebfbe43b28138b7f7b48f34c5a76c3a	{ "rev": "1-5e97009804800cff64c39df8dccb6...
□	1166941e69d252626b8eca46080aeab6	1166941e69d252626b8eca46080aeab6	{ "rev": "1-0842bb81b11cd61fe20ed094ab2a...
□	703d29b74356ef26f949797731137cf2	703d29b74356ef26f949797731137cf2	{ "rev": "1-d95bc0f570ae15c90cb7504c862b9...
□	97c5d59e72ae6c2a155702db7151fe7a	97c5d59e72ae6c2a155702db7151fe7a	{ "rev": "1-c02935667733cc4e2db9e07a5ac8...
□	a79bf4f6dcdb1b32ebeefa5c478fa81d	a79bf4f6dcdb1b32ebeefa5c478fa81d	{ "rev": "1-eefce107643e599a668ede810bb81...
□	c7fc90ace4fcfe8b62132f4a9f2c28ca	c7fc90ace4fcfe8b62132f4a9f2c28ca	{ "rev": "1-32cd76c4d4a5dc4aace8fae2a256a...
□	e9e28c06775212fe00e77dc99dc8e97d	e9e28c06775212fe00e77dc99dc8e97d	{ "rev": "1-03ac3bfe844e7d696b5970b64dba...
□	f347b088872d7656a0ff956b53c31715	f347b088872d7656a0ff956b53c31715	{ "rev": "1-fed4def46a583defd9df7960c67198...
□	f908b61af224813129b44b117fda0a5f	f908b61af224813129b44b117fda0a5f	{ "rev": "1-24f20fddd2e9e06c4b6ffbeeee6b58...

Showing document 1 - 9. Documents per page: 20

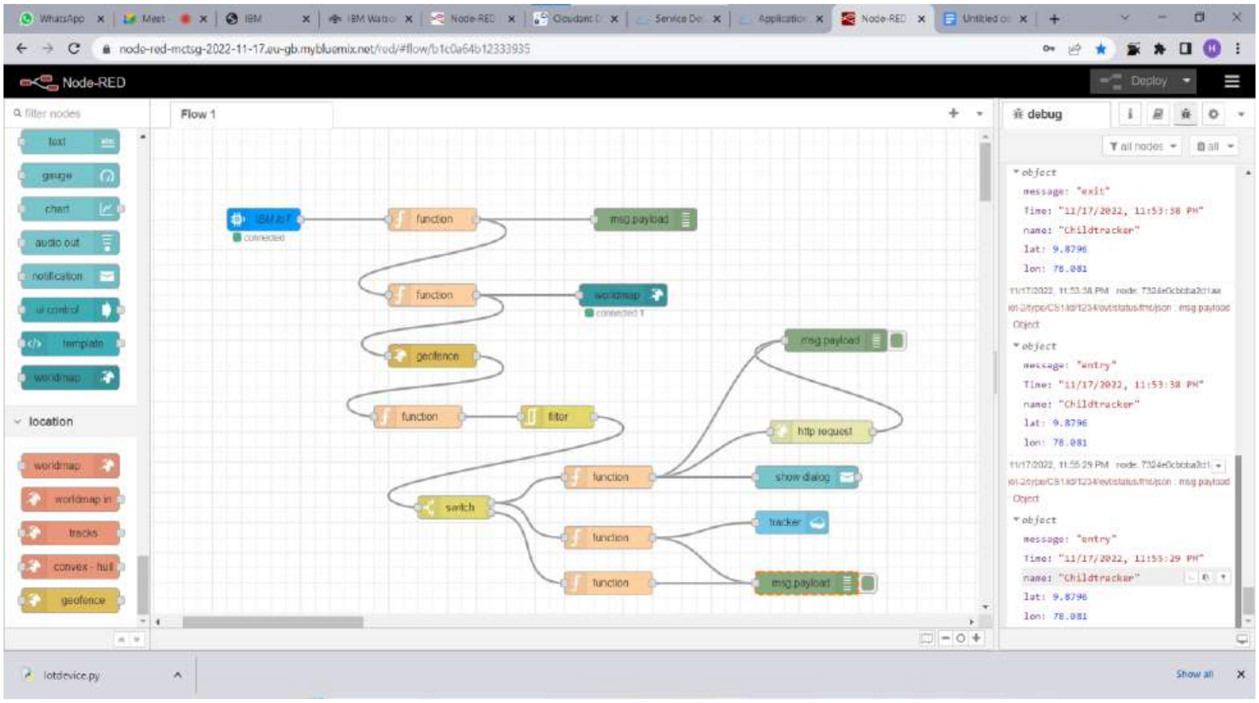


Databases

Name	Size	# of Docs	Partitioned	Actions
childsafety	14 bytes	1	No	
noderedmctsg20221117	29.5 KB	4	No	
tracker	1.8 KB	7	No	

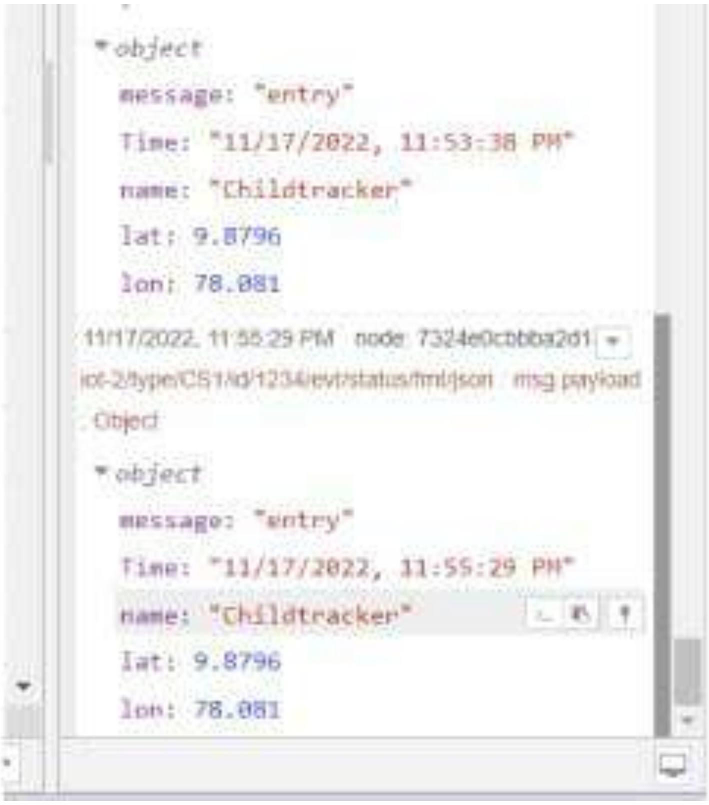
Showing 1–3 of 3 databases. Databases per page 20

NODE RED – In Area



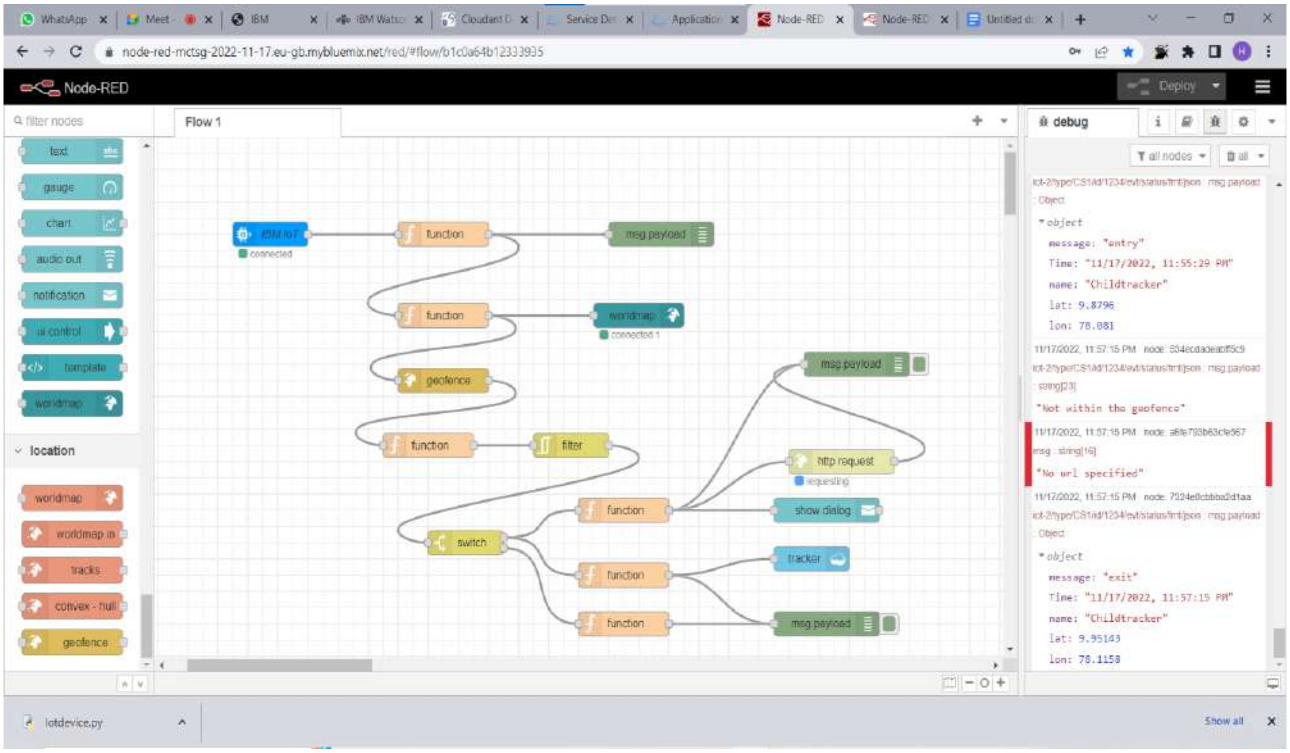
```
▾object
  message: "exit"
  Time: "11/17/2022, 11:53:38 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
11/17/2022, 11:53:38 PM  node: 7324e0cbdba2d1aa
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload : Object
▾object
  message: "entry"
  Time: "11/17/2022, 11:53:38 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
11/17/2022, 11:55:29 PM  node: 7324e0cbdba2d1
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload : Object
▾object
  message: "entry"
  Time: "11/17/2022, 11:55:29 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
```

In Area Output



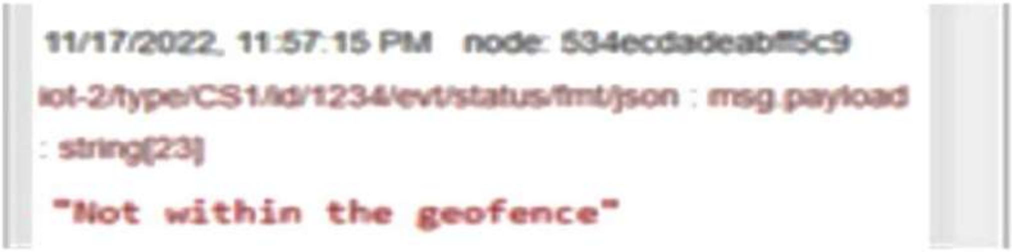
```
▾object
  message: "entry"
  Time: "11/17/2022, 11:53:38 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
11/17/2022, 11:55:29 PM  node: 7324e0cbbba2d1
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: Object
▾object
  message: "entry"
  Time: "11/17/2022, 11:55:29 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
```

NODE RED – Out Area

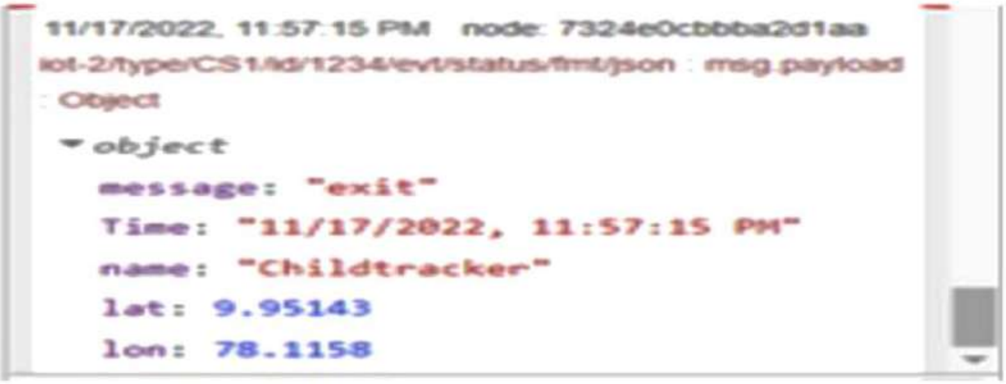


```
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: Object
▾object
  message: "entry"
  Time: "11/17/2022, 11:55:29 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
11/17/2022, 11:57:15 PM  node: 534ecdadeadf5c9
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: string[23]
"Not within the geofence"
11/17/2022, 11:57:15 PM  node: a6fe793b63c1e567
msg : string[16]
"No url specified"
11/17/2022, 11:57:15 PM  node: 7324e0cbbba2d1aa
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: Object
▾object
  message: "exit"
  Time: "11/17/2022, 11:57:15 PM"
  name: "Childtracker"
  lat: 9.95143
  lon: 78.1158
```

Out Area Output

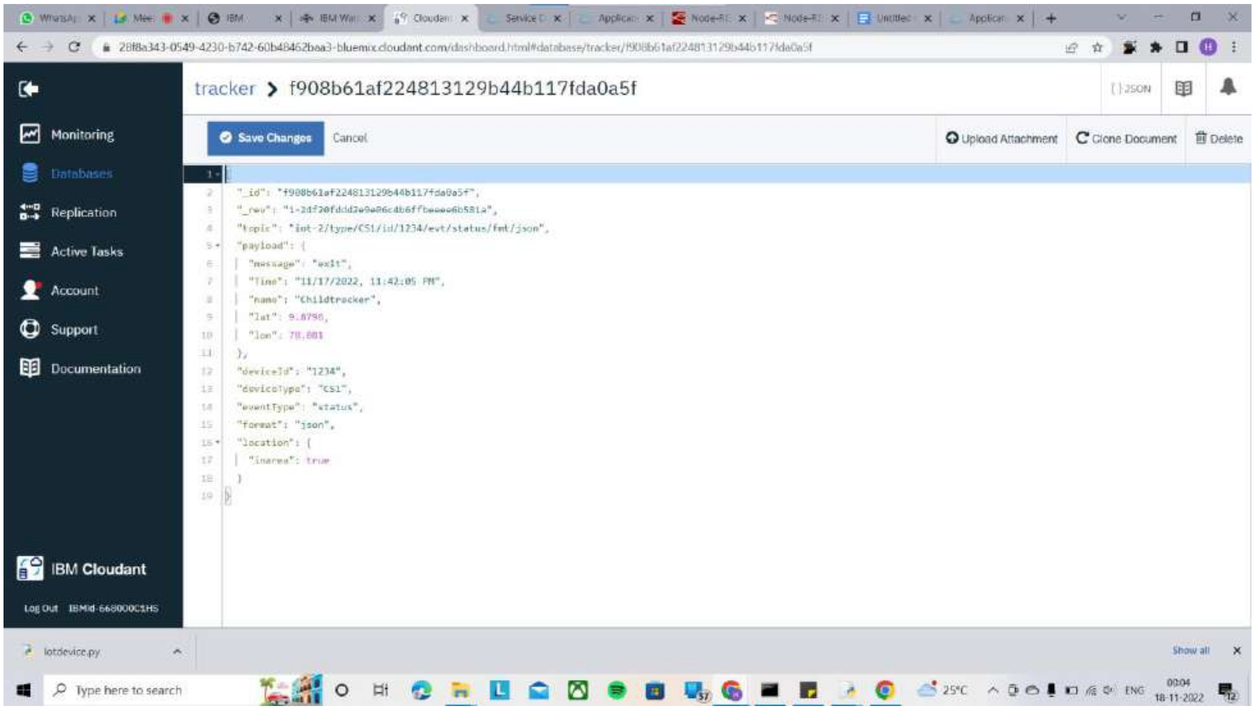


```
11/17/2022, 11:57:15 PM   node: 534ecdadeabff5c9
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: string[23]
"Not within the geofence"
```



```
11/17/2022, 11:57:15 PM   node: 7324e0cbbba2d1aa
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: Object
▾object
  message: "exit"
  Time: "11/17/2022, 11:57:15 PM"
  name: "Childtracker"
  lat: 9.95143
  lon: 78.1158
```

Data received in Cloudant

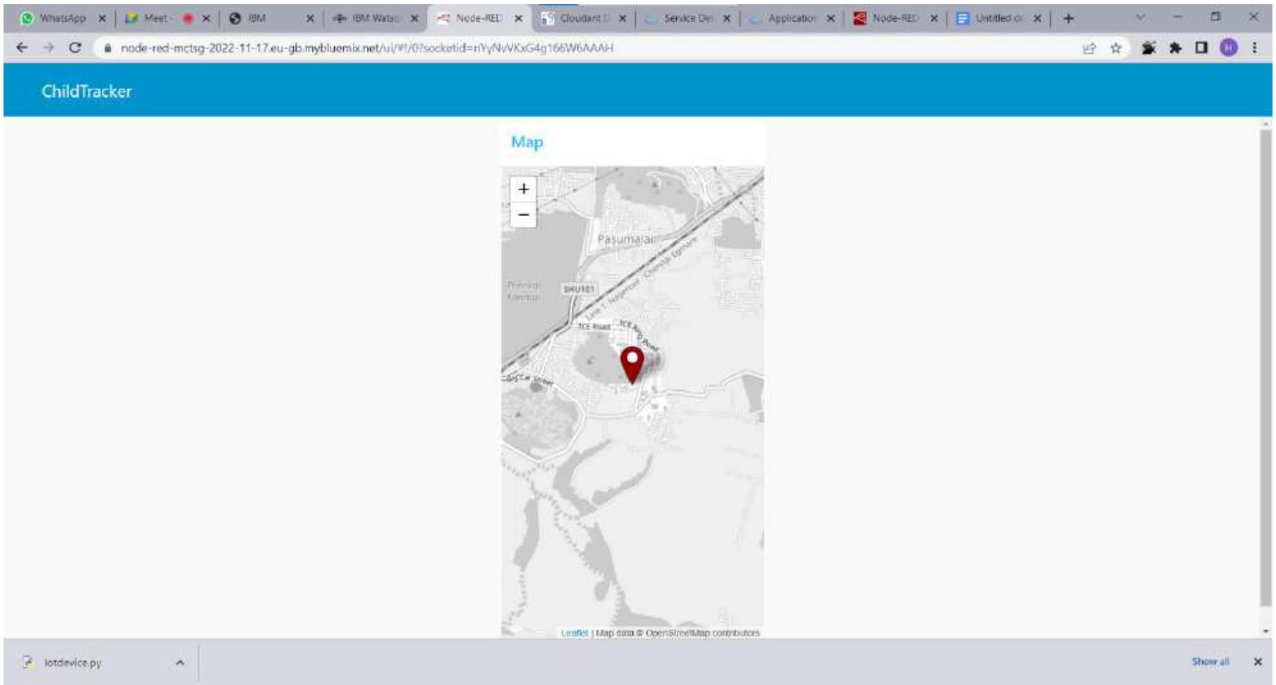


tracker > f908b61af224813129b44b117fda0a5f

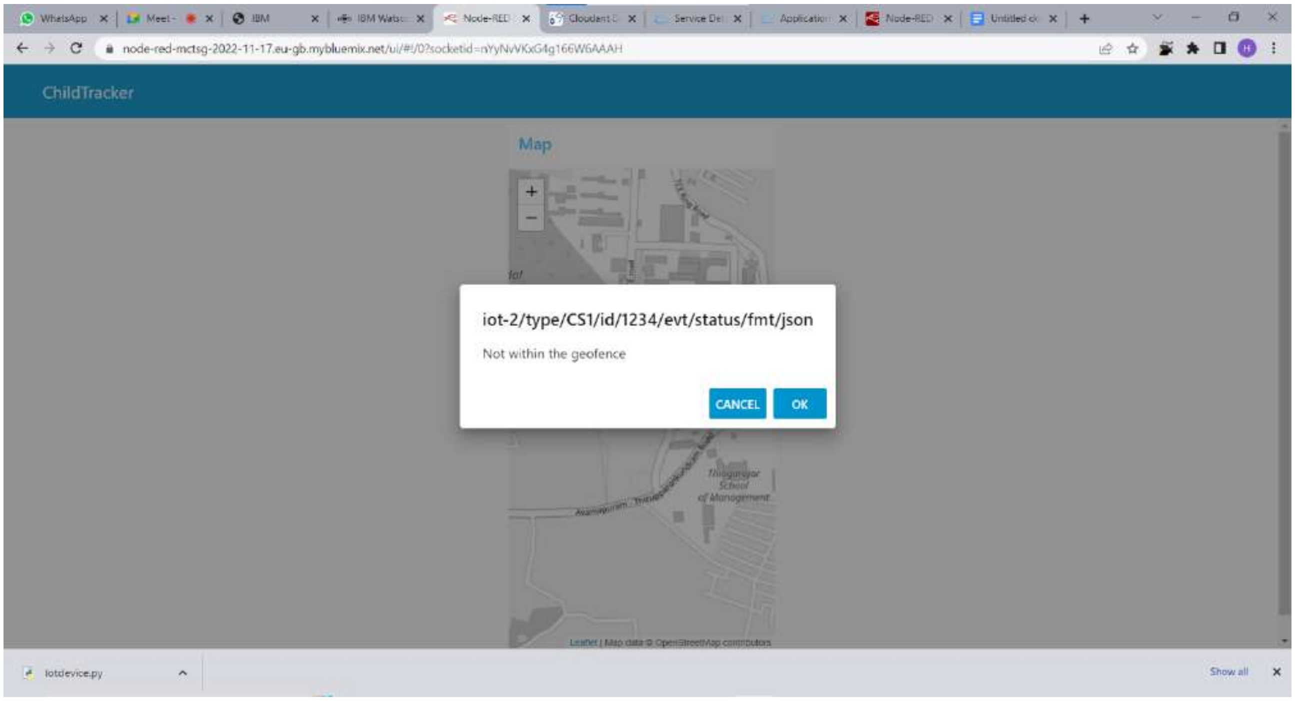
```
{
  "_id": "f908b61af224813129b44b117fda0a5f",
  "_rev": "1-2df20fddd2e9e06cdb6ffbeeee6b581a",
  "topic": "iot-2/type/CS1/id/1234/evt/status/fmt/json",
  "payload": {
    "message": "exit",
    "Time": "11/17/2022, 11:42:05 PM",
    "name": "Childtracker",
    "lat": 9.8798,
    "lon": 78.081
  },
  "deviceId": "1234",
  "deviceType": "CS1",
  "eventType": "status",
  "format": "json",
  "location": {
    "inarea": true
  }
}
```

NODE RED DASHBOARD(WEB UI)

In Area:



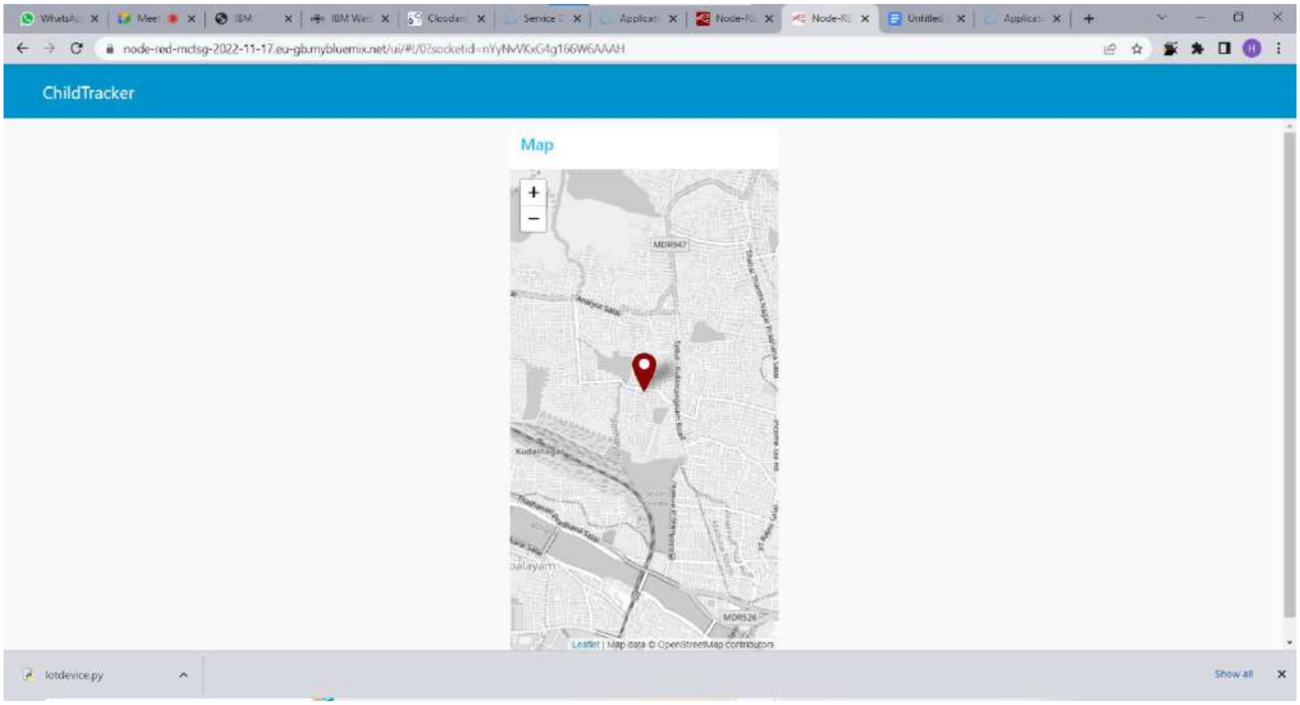
Out Area:


WhatsApp
Meet
IBM
IBM Watson
Node-RED
Cloudant
Service Det
Application
Node-RED
Untitled
node-red-mctsg-2022-11-17.eu-gb.mybluemix.net/ui/#!/0?socketid=nYyNvVKxG4g166W6AAAH
ChildTracker
Map
+
−
Thiagarajar School of Management
Leaflet | Map data © OpenStreetMap contributors
Iotdevice.py
Show all

iot-2/type/CS1/id/1234/evt/status/fmt/json

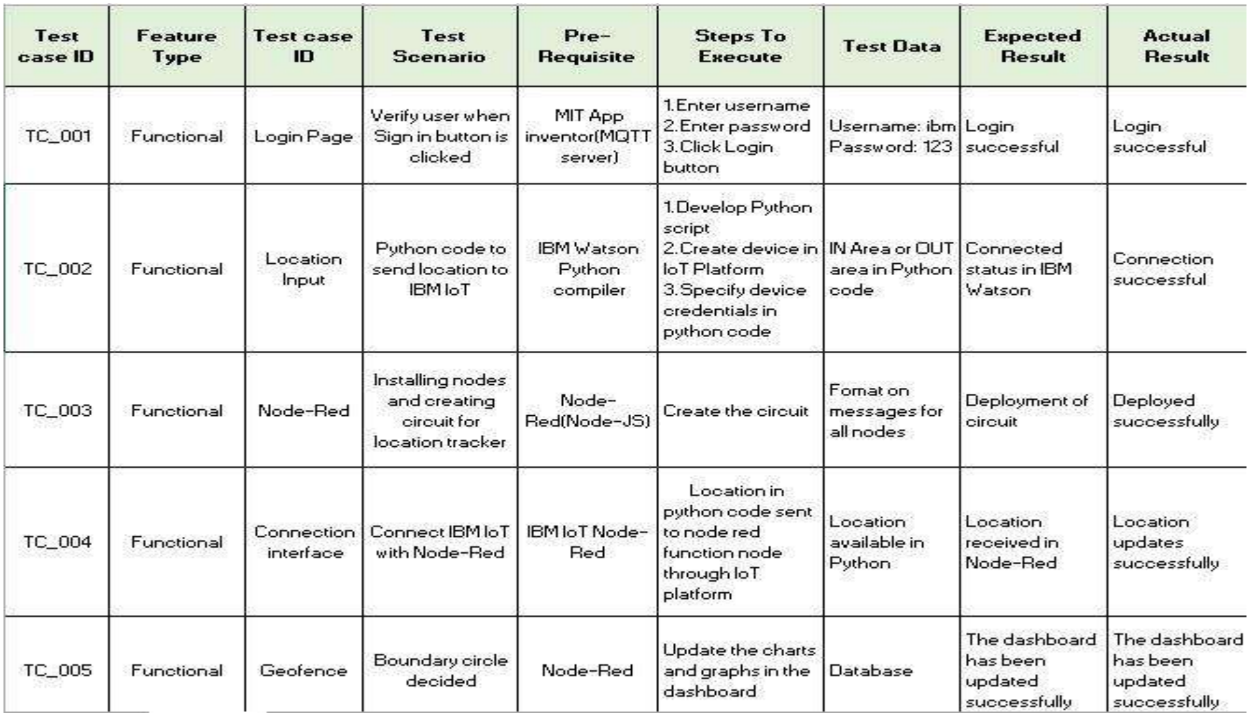
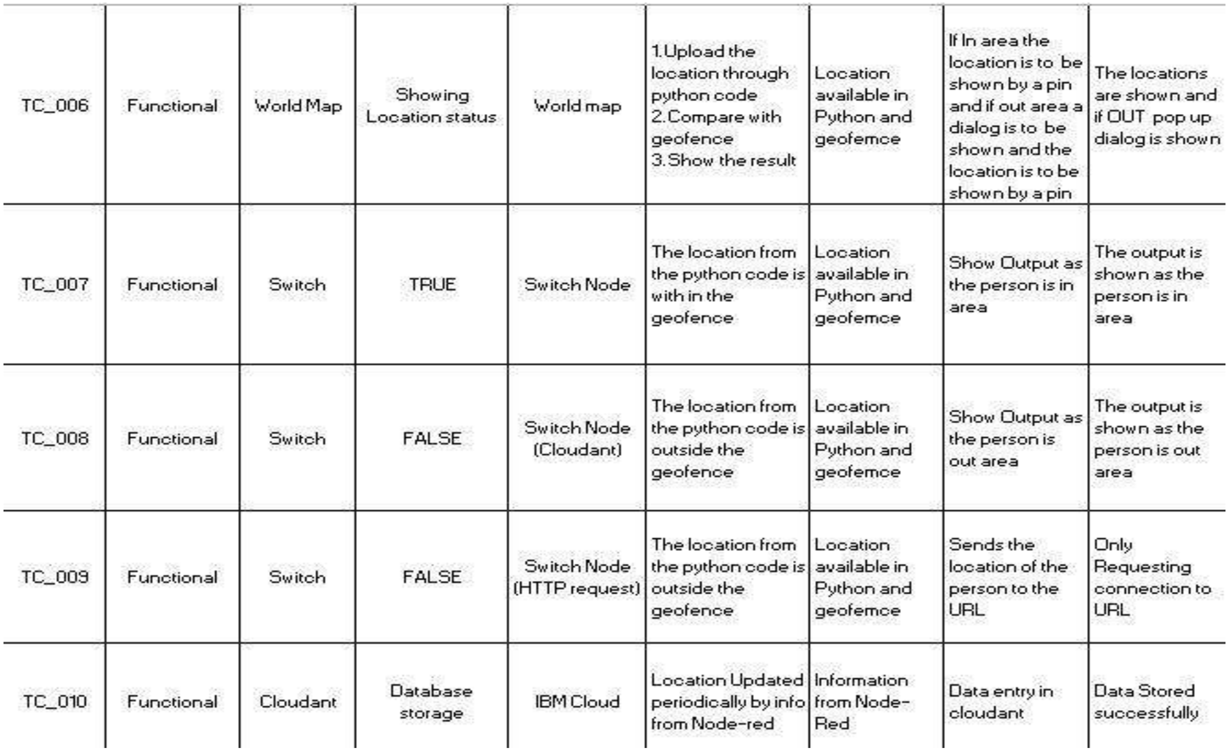
Not within the geofence

Out Area Location:



8.TESTING

8.1Test Cases

Test case ID	Feature Type	Test case ID	Test Scenario	Pre-Requisite	Steps To Execute	Test Data	Expected Result	Actual Result
TC_001	Functional	Login Page	Verify user when Sign in button is clicked	MIT App inventor(MQTT server)	1.Enter username 2.Enter password 3.Click Login button	Username: ibm Password: 123	Login successful	Login successful
TC_002	Functional	Location Input	Python code to send location to IBM IoT	IBM Watson Python compiler	1.Develop Python script 2.Create device in IoT Platform 3.Specify device credentials in python code	IN Area or OUT area in Python code	Connected status in IBM Watson	Connection successful
TC_003	Functional	Node-Red	Installing nodes and creating circuit for location tracker	Node-Red(Node-JS)	Create the circuit	Fomat on messages for all nodes	Deployment of circuit	Deployed successfully
TC_004	Functional	Connection interface	Connect IBM IoT with Node-Red	IBM IoT Node-Red	Location in python code sent to node red function node through IoT platform	Location available in Python	Location received in Node-Red	Location updates successfully
TC_005	Functional	Geofence	Boundary circle decided	Node-Red	Update the charts and graphs in the dashboard	Database	The dashboard has been updated successfully	The dashboard has been updated successfully
TC_006	Functional	World Map	Showing Location status	World map	1.Upload the location through python code 2.Compare with geofence 3.Show the result	Location available in Python and geofemce	If In area the location is to be shown by a pin and if out area a dialog is to be shown and the location is to be shown by a pin	The locations are shown and if OUT pop up dialog is shown
TC_007	Functional	Switch	TRUE	Switch Node	The location from the python code is with in the geofence	Location available in Python and geofemce	Show Output as the person is in area	The output is shown as the person is in area
TC_008	Functional	Switch	FALSE	Switch Node (Cloudant)	The location from the python code is outside the geofence	Location available in Python and geofemce	Show Output as the person is out area	The output is shown as the person is out area
TC_009	Functional	Switch	FALSE	Switch Node (HTTP request)	The location from the python code is outside the geofence	Location available in Python and geofemce	Sends the location of the person to the URL	Only Requesting connection to URL
TC_010	Functional	Cloudant	Database storage	IBM Cloud	Location Updated periodically by info from Node-red	Information from Node-Red	Data entry in cloudant	Data Stored successfully

8.2 User Acceptance Testing

Defect Analysis

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	2	0	5	0	7
Duplicate	0	0	3	0	3
External	0	0	0	7	7
Fixed	1	1	2	0	4
Not Reproduced	1	2	0	0	3
Skipped	0	0	0	3	3
Won't Fix	0	0	0	6	6
Totals	4	3	10	16	33

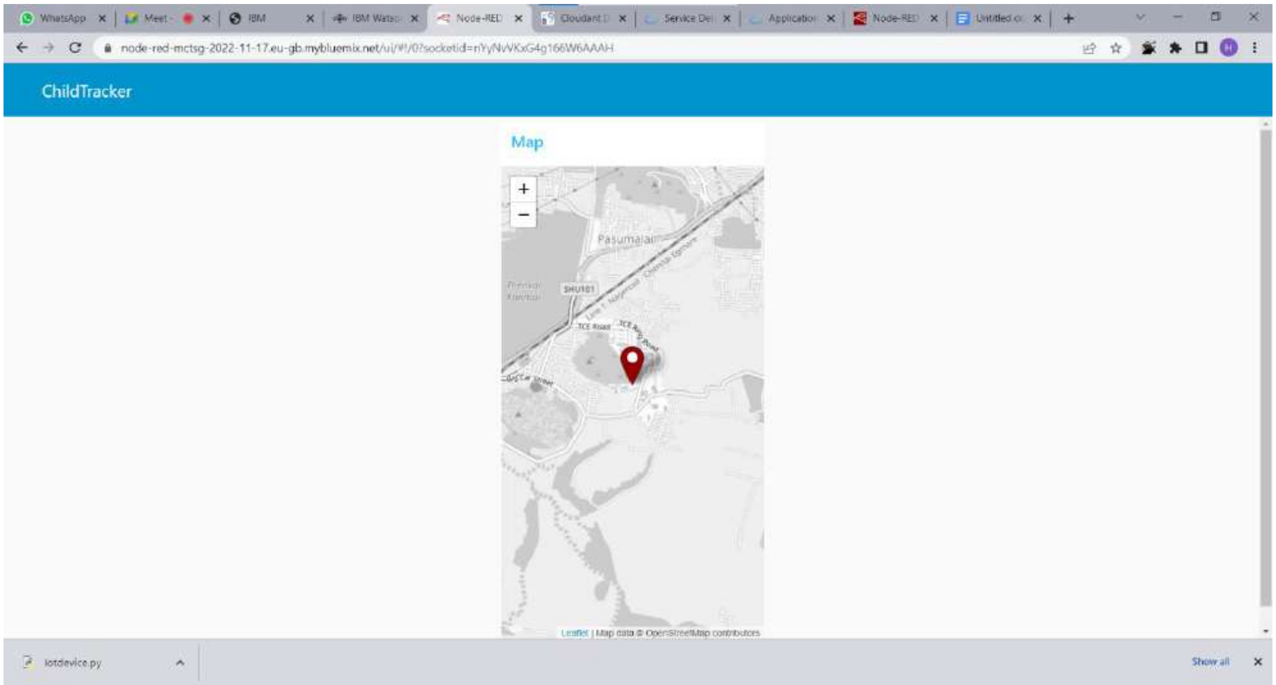
Test Case Analysis

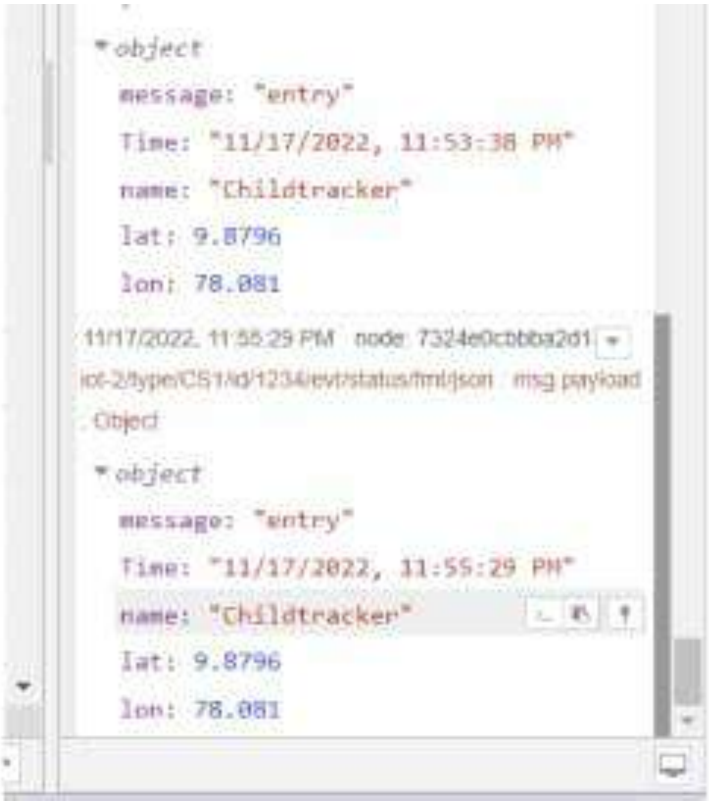
Section	Total Cases	Not Tested	Fail	Pass
Login page	6	0	0	6
Geofencing	20	0	0	20
Security	4	4	0	0
Visual popup notification	5	0	0	5
Fast2sms	3	0	3	0
Cloudant database	8	0	0	8
Plotting world map	5	0	0	5

9. Result:

9.1 Outputs and Performance Matrix:

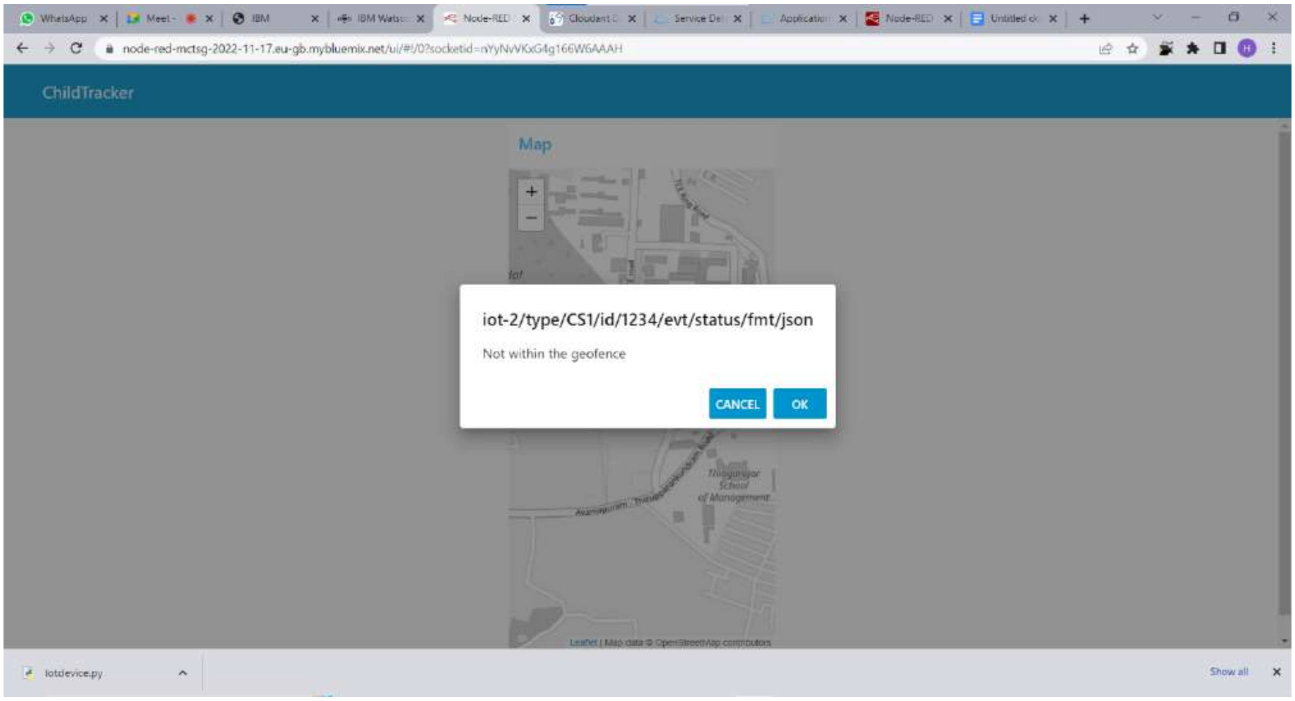
In Area Output:



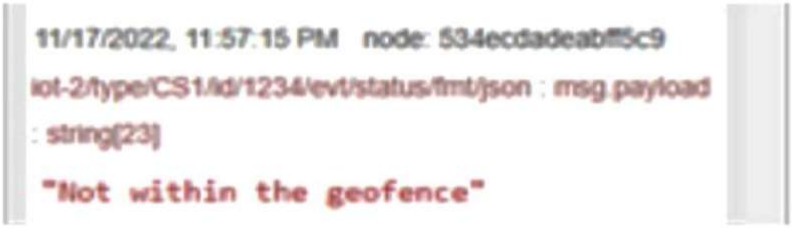


```
▾object
  message: "entry"
  Time: "11/17/2022, 11:53:38 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
11/17/2022, 11:55:29 PM  node: 7324e0cbbba2d1
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: Object
▾object
  message: "entry"
  Time: "11/17/2022, 11:55:29 PM"
  name: "Childtracker"
  lat: 9.8796
  lon: 78.081
```

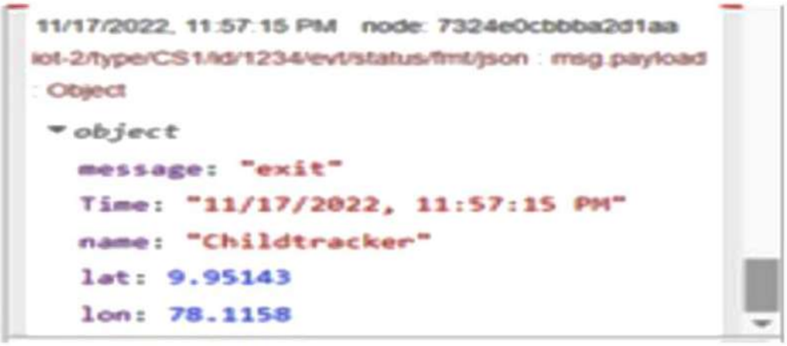
Out Area Output:


node-red-mctsg-2022-11-17.eu-gb.mybluemix.net/ui/#!/0?socketid=nYyNvVKxG4g166W6AAAH
ChildTracker
Map
Leaflet | Map data © OpenStreetMap contributors
iotdevice.py
Show all

```
iot-2/type/CS1/id/1234/evt/status/fmt/json
Not within the geofence
CANCEL   OK
```

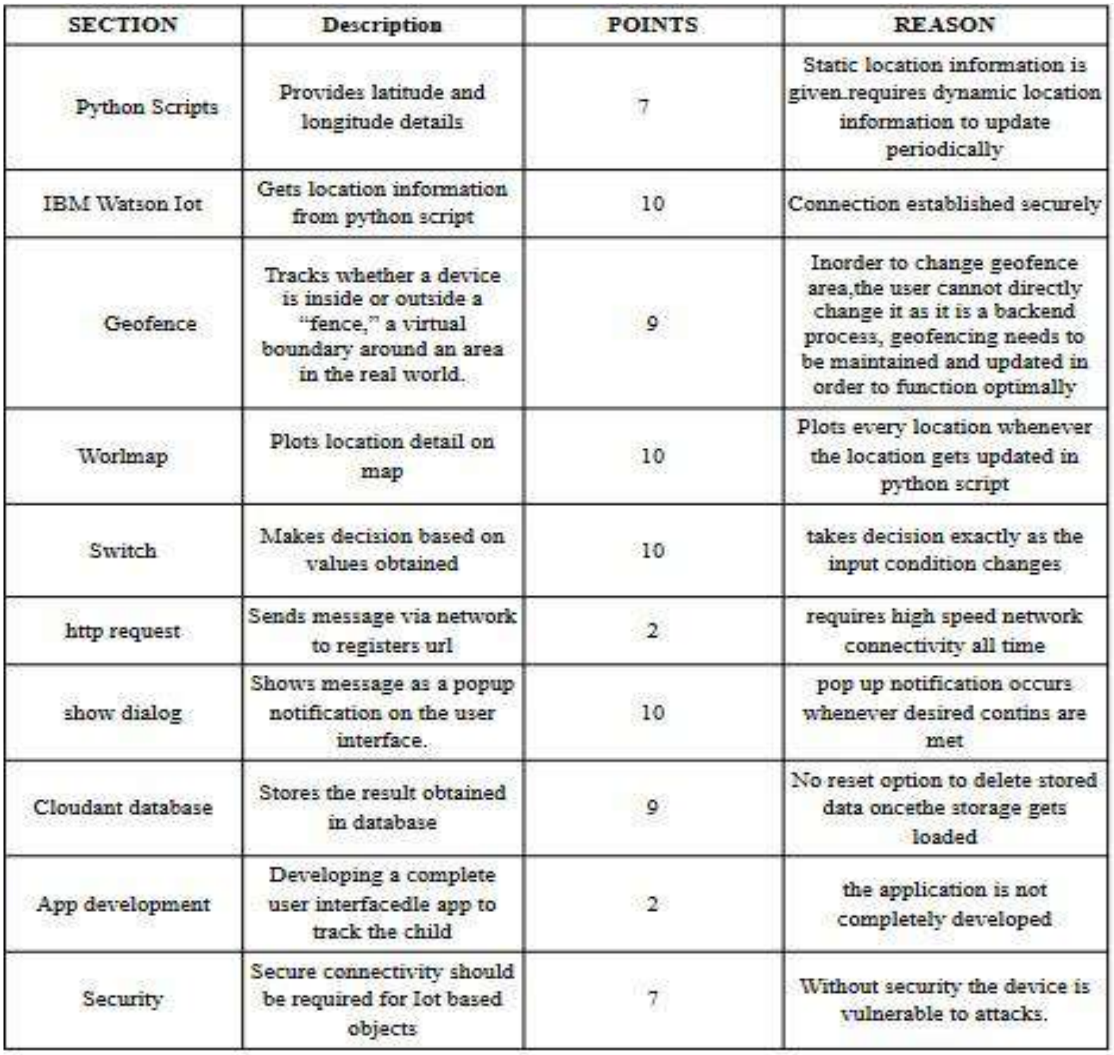


```
11/17/2022, 11:57:15 PM   node: 534ecdadeabff5c9
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: string[23]
"Not within the geofence"
```

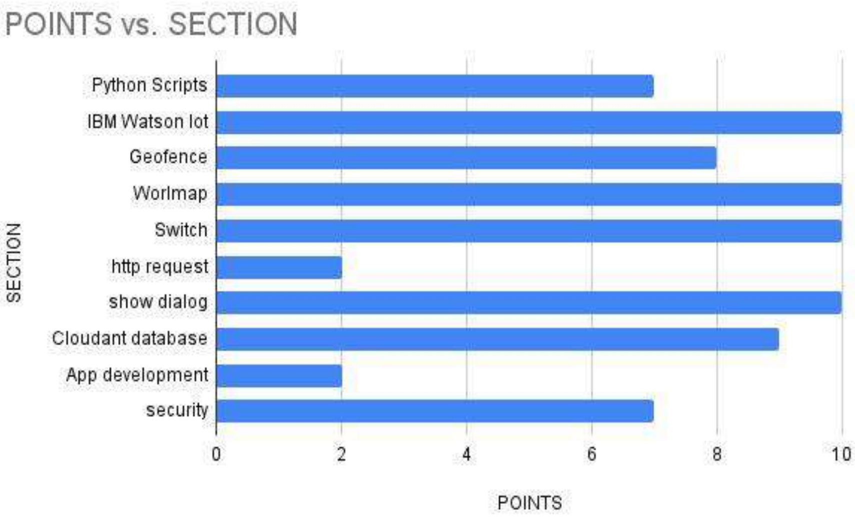


```
11/17/2022, 11:57:15 PM   node: 7324e0cbbba2d1aa
iot-2/type/CS1/id/1234/evt/status/fmt/json : msg.payload
: Object
 ▾object
   message: "exit"
   Time: "11/17/2022, 11:57:15 PM"
   name: "Childtracker"
   lat: 9.95143
   lon: 78.1158
```

Performance Analysis



SECTION	Description	POINTS	REASON
Python Scripts	Provides latitude and longitude details	7	Static location information is given.requires dynamic location information to update periodically
IBM Watson Iot	Gets location information from python script	10	Connection established securely
Geofence	Tracks whether a device is inside or outside a "fence," a virtual boundary around an area in the real world.	9	Inorder to change geofence area,the user cannot directly change it as it is a backend process, geofencing needs to be maintained and updated in order to function optimally
Worlmap	Plots location detail on map	10	Plots every location whenever the location gets updated in python script
Switch	Makes decision based on values obtained	10	takes decision exactly as the input condition changes
http request	Sends message via network to registers url	2	requires high speed network connectivity all time
show dialog	Shows message as a popup notification on the user interface.	10	pop up notification occurs whenever desired contins are met
Cloudant database	Stores the result obtained in database	9	No reset option to delete stored data oncethe storage gets loaded
App development	Developing a complete user interfacedle app to track the child	2	the application is not completely developed
Security	Secure connectivity should be required for Iot based objects	7	Without security the device is vulnerable to attacks.



10. ADVANTAGES & DISADVANTAGES:

Advantages:

1.Tracking Made Easy:

Fueled by IOT, the GPS enabled Wearable Device act as a saviour for parents who are always clouded with worries about their kids. Tracking a child was never this easy. These Wearable Devices allow parents to track their children in crowded/public places or when they are out of sight, say at school, picnic or an outing. Parents can use these gadgets to track the location of their kids.

2. Guarantees Peace of Mind to Parents:

Parents, whether at home or office, are always worried about the safety of their kids. The fear of losing your child to avoidable circumstances is the concern area for all mommies and daddies. On the other hand, a smart gadget equipped kid is always traceable and reachable in case of contingencies and emergencies. This in fact, offers great solace for parents, who are relieved at the thought of maintaining an uninterrupted connectivity with their children, anytime, anywhere. Enough to of course, guarantee the much-needed peace of mind

Disadvantages:

Keeping the data gathered and transmitted by IoT devices safe is challenging, as they evolve and expand in use. Although cybersecurity is a high priority, IoT devices aren't always included in the strategy. Devices must be protected from physical tampering, internet-based software attacks, network-based attacks and hardware-based attacks.the device has dependencies on power and internet for continuous function.when either goes down,so as the device.Requires active internet connection. Children need to login once into the application. System will provide inaccurate results if data not entered correctly

11.CONCLUSION:

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 abusements, 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 guarantees peace of mind.

This report cleary explaines the IoT concept, child safety issues and the need of using child security system. It assists parents to monitor their children remotely. In case situations when the kid steps out of geofence, notifications will be sent to parents so that actions can be taken. Through this, child safety can be ensured and safety will be reduced.

12.FUTURE SCOPE:

The proposed solution only checks whether the kid is within geofence or not and sends notification to parent if the kid crosses the fence area. This requires high speed network connectivity and satellite communication inorder to send notification to the parent. However it is difficult to ensure secure internet connectivity and communication. This problem can be overcome by using zigbee technology or LoRa technology. LoRa is the de facto wireless platform of Internet of Things (IoT). Semtech's LoRa chipsets connect sensors to the Cloud and enable real-time communication of data and analytics that can be utilized to enhance efficiency and productivity. Also the proposed system can be upgraded by adding additional features like SOS emergency button,sensing sudden vibration/high sound and send a call alert,pulse rate detection etc.

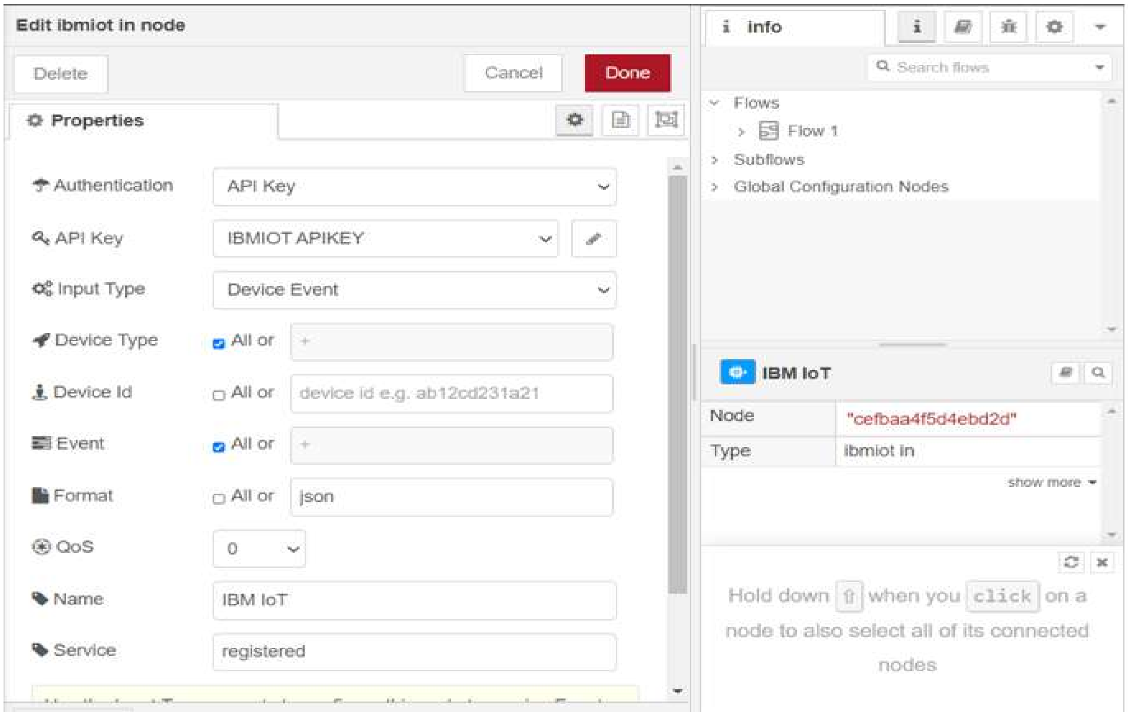
13.Appendix

PYTHON SCRIPT:

```
import time

import random

#import ibmiotf.application

import ibmiotf.device

import sys


config={

   "org":"phsvnq",

   "type"  :"CS1",

   "id":"1234",

   "auth-method":"token",

   "auth-token":"123456789"

}

client= ibmiotf.device.Client (config)

client.connect()

def myCommandCallback (cmd):

   a=cmd.data

   if len(a["command"])==0:

      pass

   else:

      print(a["command"])

def pub (data):

   client.publishEvent  (event="status",  msgFormat="json",data=data,  qos=0)

   print("Published  data Successfully: %s",data)

while True:
   name= "Childtracker"

   #in area
   #latitude= 9.8796
   #longitude= 78.0810

   #out area
   latitude= 9.95143
   longitude= 78.1158

   data={'name': name, 'lat':latitude,'lon':longitude}
   pub(data)
   client.commandCallback = myCommandCallback
   time.sleep(2)
client.disconnect()
```

NODERED:

IBM IOT:



Edit ibmiot in node

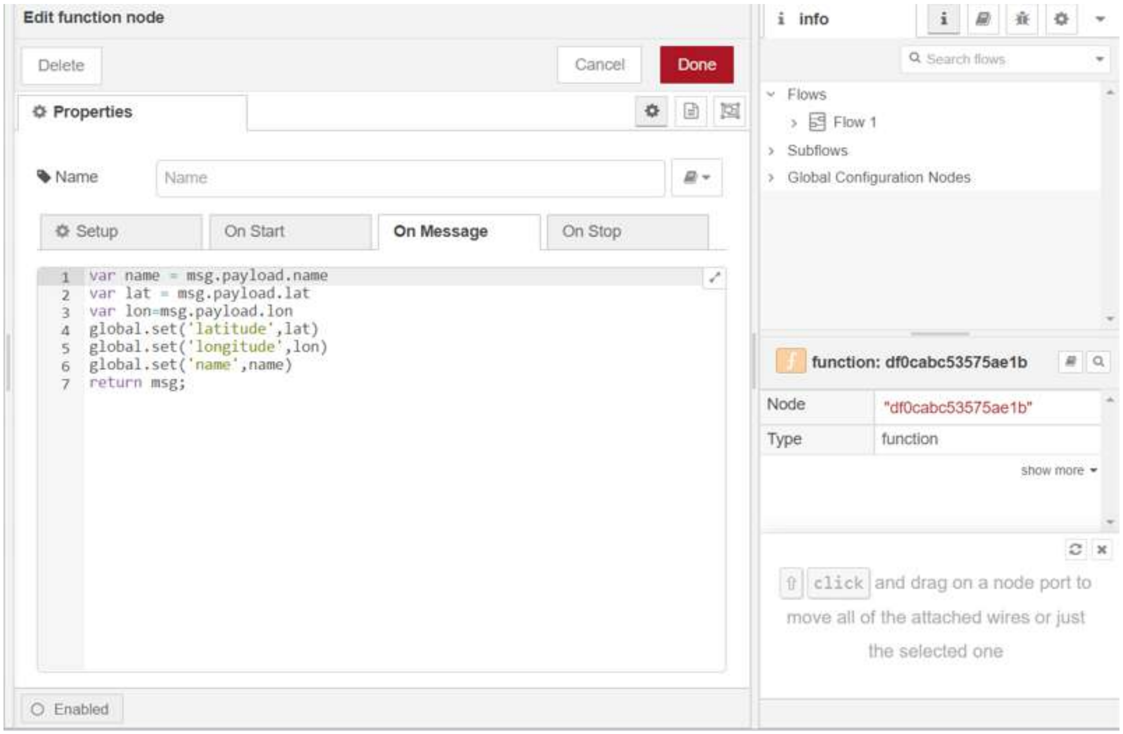
Authentication: API Key
API Key: IBMIOT APIKEY
Input Type: Device Event
Device Type: ☑ All or +
Device Id: □ All or device id e.g. ab12cd231a21
Event: ☑ All or +
Format: □ All or json
QoS: 0
Name: IBM IoT
Service: registered

IBM IoT

Node	"cefbaa4f5d4ebd2d"
Type	ibmiot in

Hold down ⇧ when you click on a node to also select all of its connected nodes

FUNCTIONS:



Edit function node

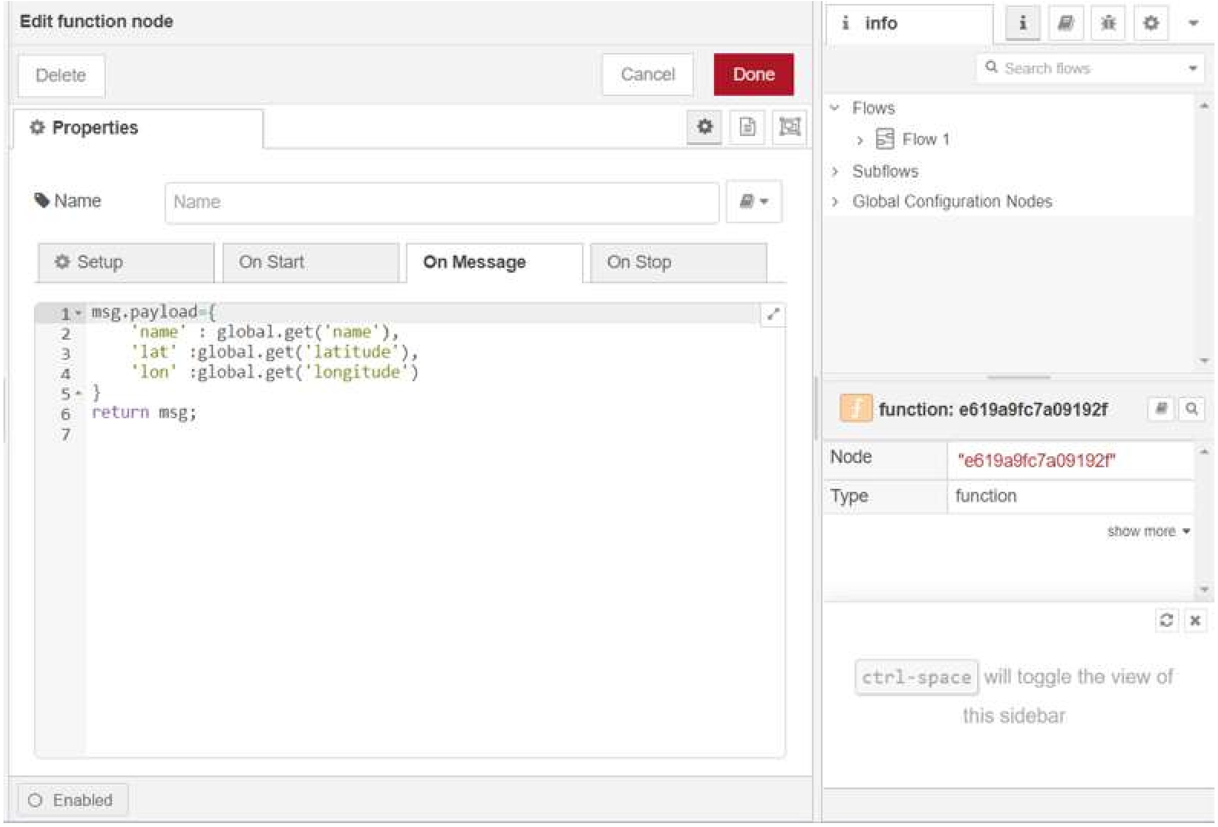
Name: Name

```
var name = msg.payload.name
var lat = msg.payload.lat
var lon=msg.payload.lon
global.set('latitude',lat)
global.set('longitude',lon)
global.set('name',name)
return msg;
```

function: df0cabc53575ae1b

Node	"df0cabc53575ae1b"
Type	function

⇧ click and drag on a node port to move all of the attached wires or just the selected one

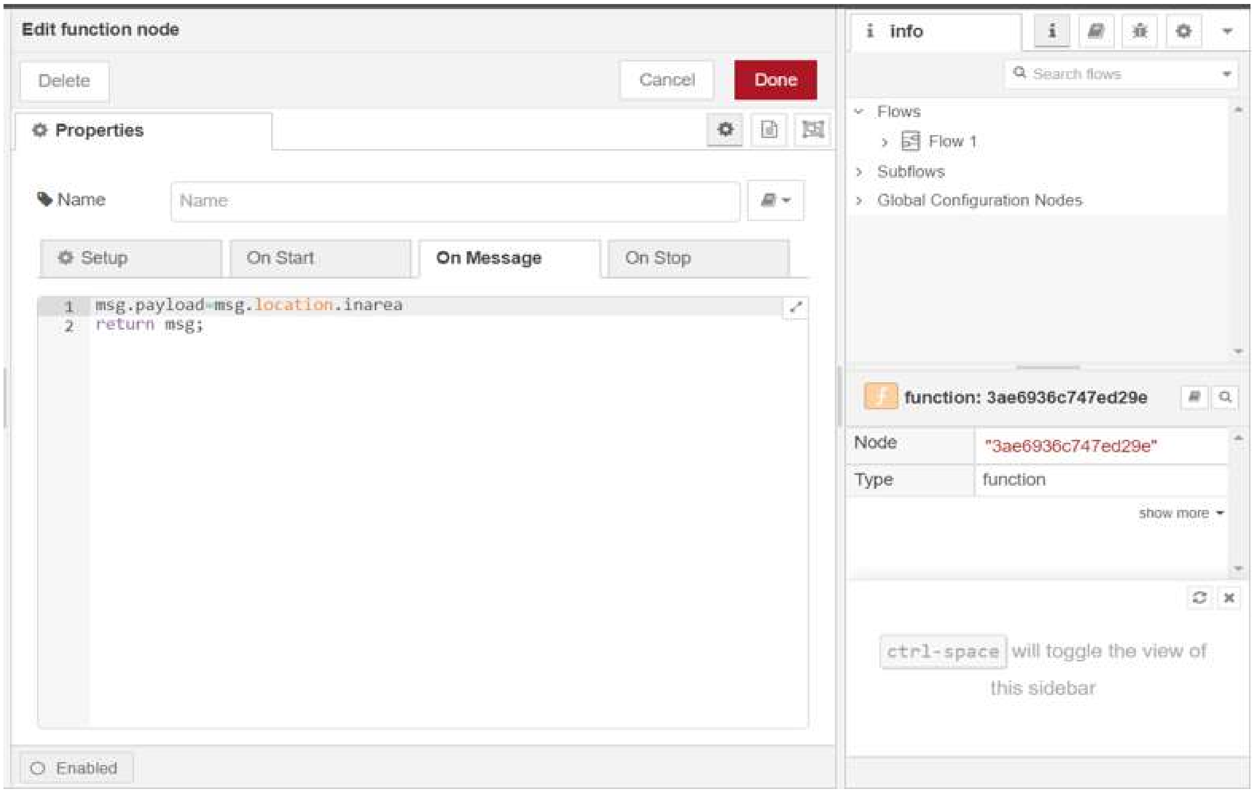


```
msg.payload={
    'name' : global.get('name'),
    'lat' :global.get('latitude'),
    'lon' :global.get('longitude')
}
return msg;
```

function: e619a9fc7a09192f

Node	"e619a9fc7a09192f"
Type	function

ctrl-space will toggle the view of this sidebar

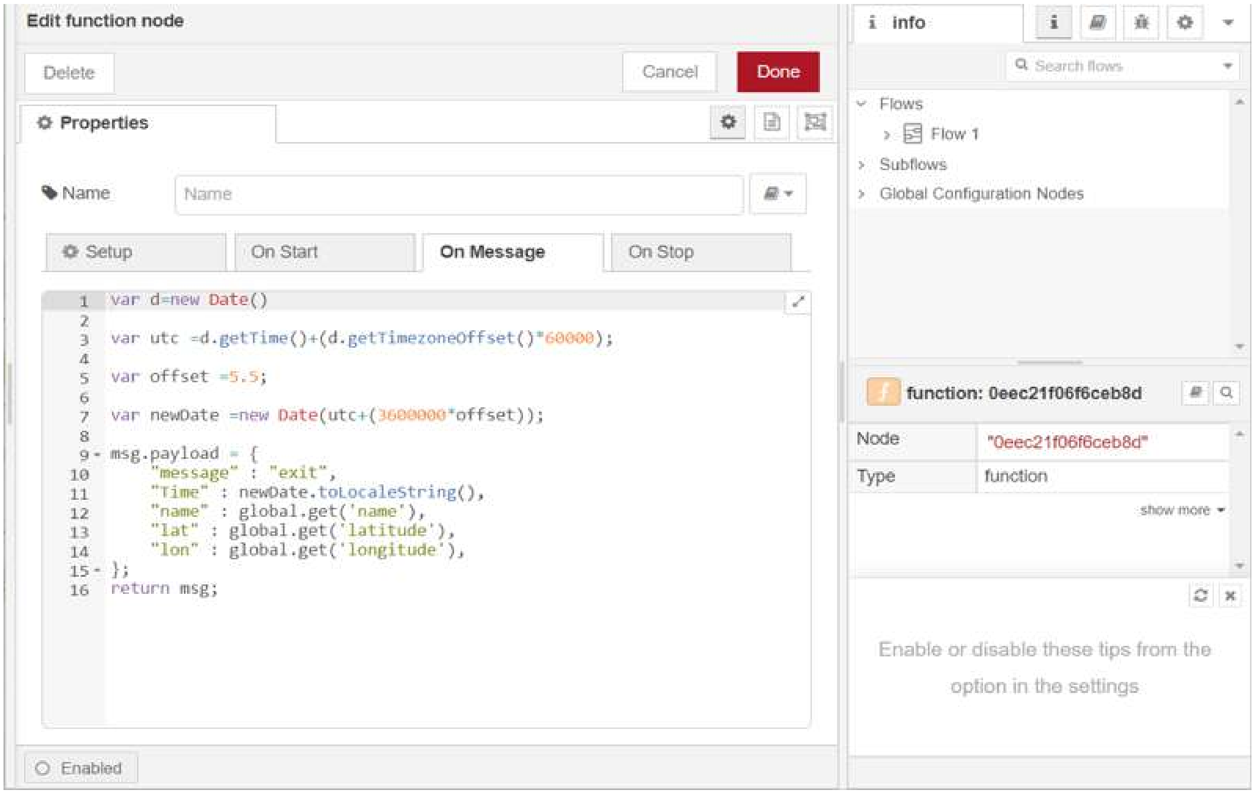


```
msg.payload=msg.location.inarea
return msg;
```

function: 3ae6936c747ed29e

Node	"3ae6936c747ed29e"
Type	function

ctrl-space will toggle the view of this sidebar



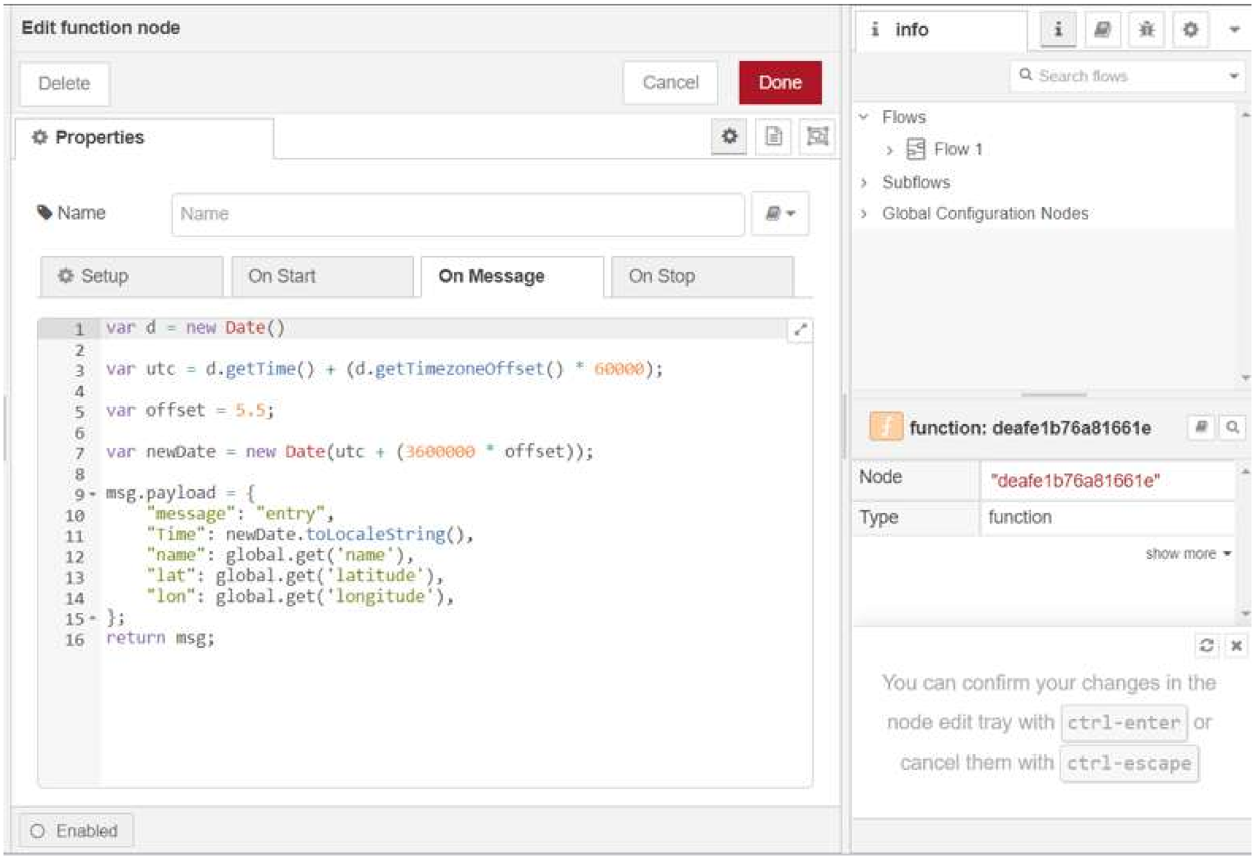
Name: Name

```
var d=new Date()

var utc =d.getTime()+(d.getTimezoneOffset()*60000);

var offset =5.5;

var newDate =new Date(utc+(3600000*offset));

msg.payload = {
    "message" : "exit",
    "Time" : newDate.toLocaleString(),
    "name" : global.get('name'),
    "lat" : global.get('latitude'),
    "lon" : global.get('longitude'),
};
return msg;
```

function: 0eec21f06f6ceb8d

Node	"0eec21f06f6ceb8d"
Type	function

Enable or disable these tips from the option in the settings



Name: Name

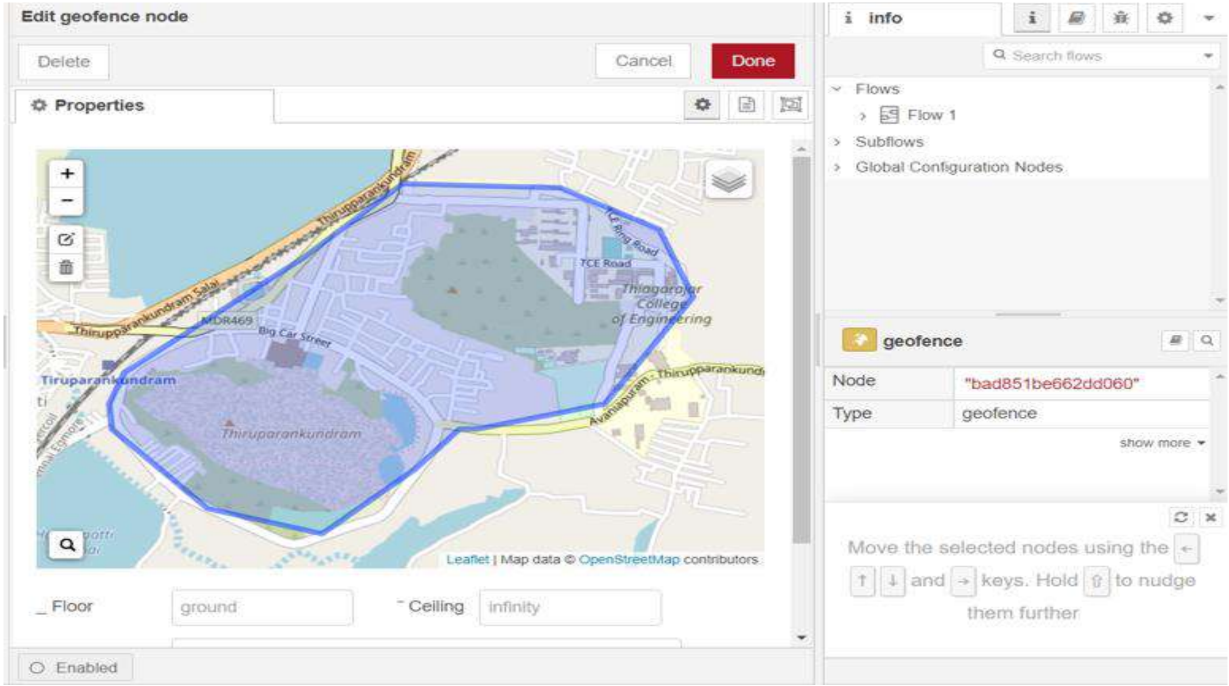
```
var d = new Date()

var utc = d.getTime() + (d.getTimezoneOffset() * 60000);

var offset = 5.5;

var newDate = new Date(utc + (3600000 * offset));

msg.payload = {
    "message": "entry",
    "Time": newDate.toLocaleString(),
    "name": global.get('name'),
    "lat": global.get('latitude'),
    "lon": global.get('longitude'),
};
return msg;
```

function: deafe1b76a81661e

Node	"deafe1b76a81661e"
Type	function

You can confirm your changes in the node edit tray with ctrl-enter or cancel them with ctrl-escape

GEOFENCE:



Edit geofence node

Floor: ground

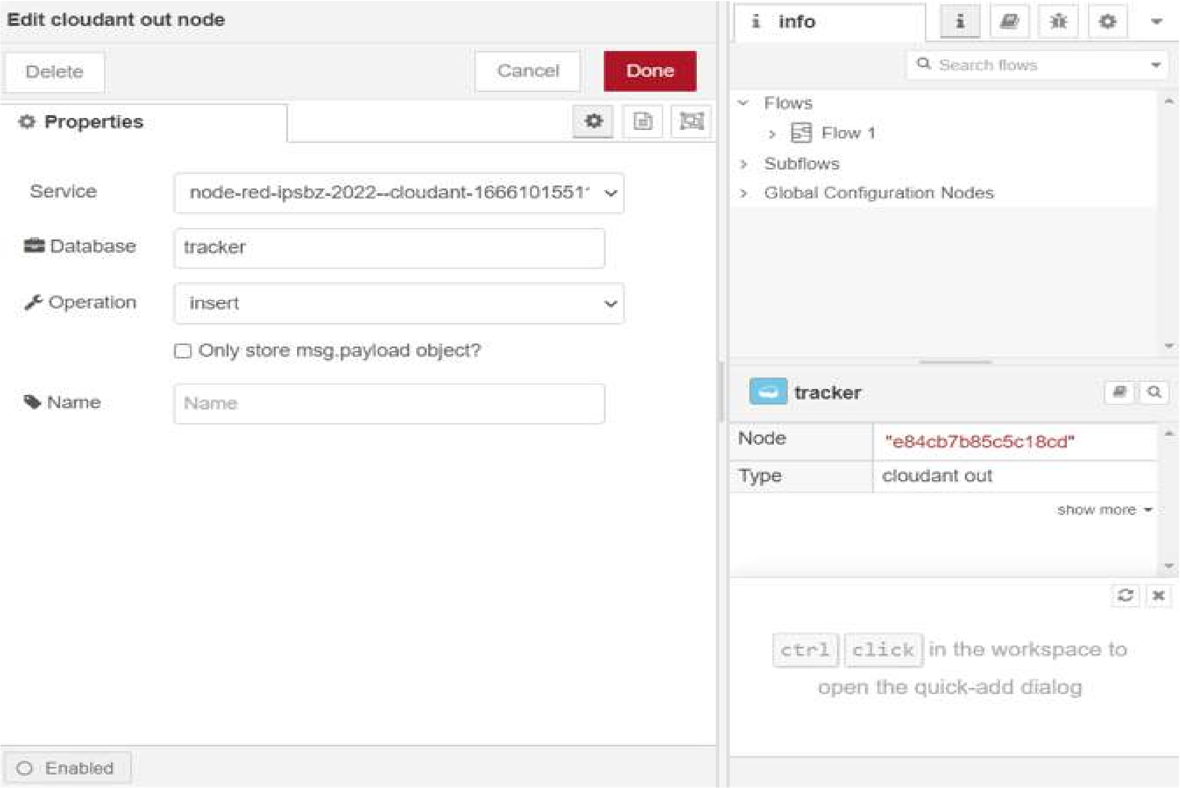
Ceiling: infinity

Node: "bad851be662dd060"

Type: geofence

Move the selected nodes using the ← ↑ ↓ and → keys. Hold ⇧ to nudge them further

CLOUDANT:



Edit cloudant out node

Service: node-red-ipsbz-2022--cloudant-1666101551

Database: tracker

Operation: insert

□ Only store msg.payload object?

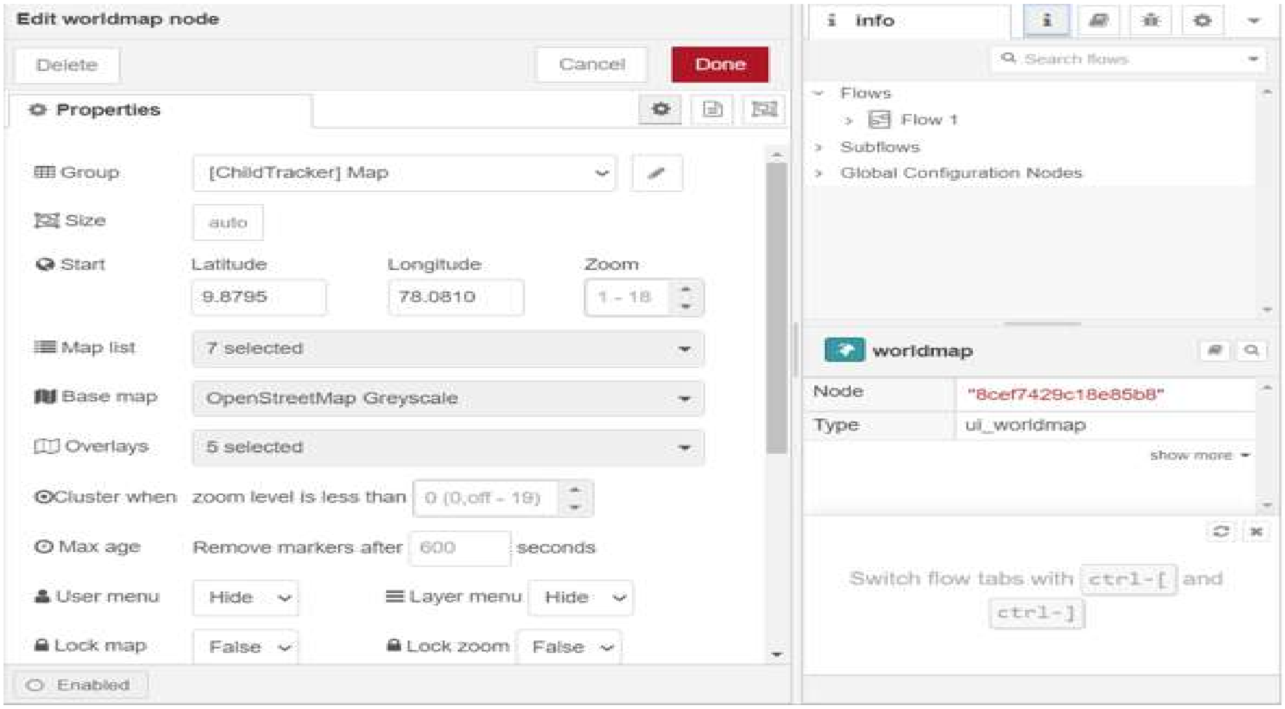
Name: Name

Node: "e84cb7b85c5c18cd"

Type: cloudant out

ctrl click in the workspace to open the quick-add dialog

WORLDMAP NODE:



Edit worldmap node

Delete Cancel Done

Properties

Group [ChildTracker] Map

Size auto

Start Latitude 9.8795 Longitude 78.0810 Zoom 1 - 18

Map list 7 selected

Base map OpenStreetMap Greyscale

Overlays 5 selected

Cluster when zoom level is less than 0 (0,off - 19)

Max age Remove markers after 600 seconds

User menu Hide Layer menu Hide

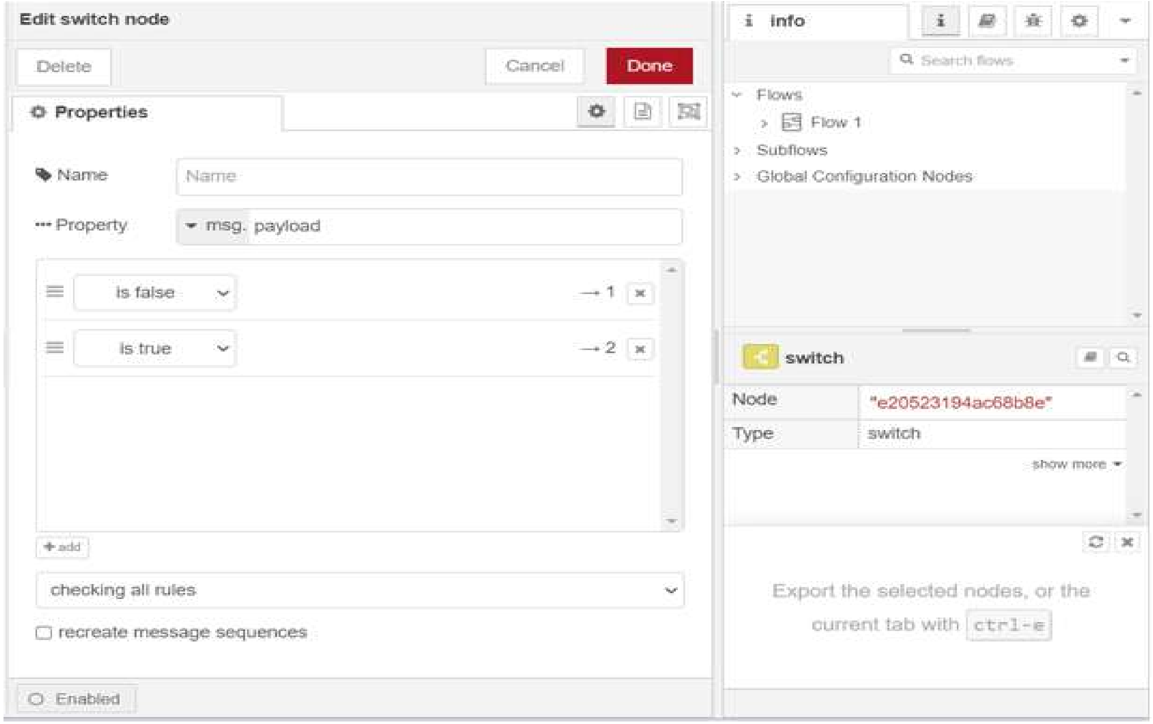
Lock map False Lock zoom False

Enabled

worldmap

Node	"8cef7429c18e85b8"
Type	ui_worldmap

SWITCH NODE:



Edit switch node

Delete Cancel Done

Properties

Name Name

Property msg. payload

is false → 1

is true → 2

+add

checking all rules

☐ recreate message sequences

Enabled

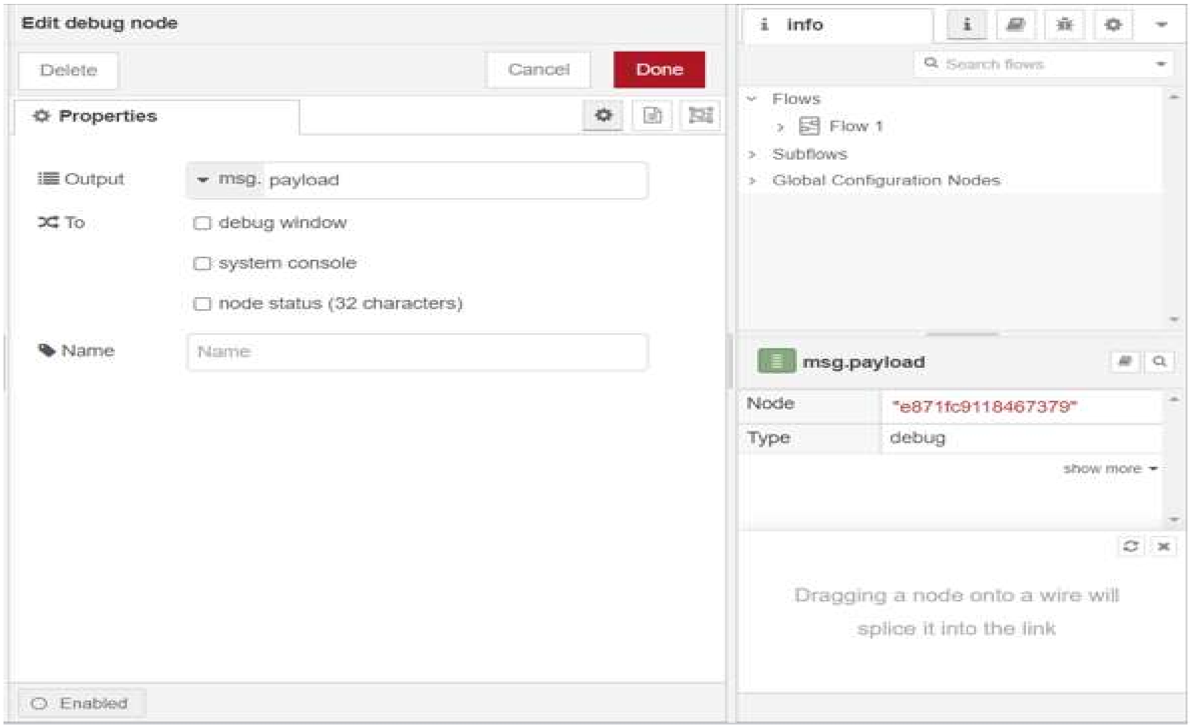
info
Search flows
Flows
Flow 1
Subflows
Global Configuration Nodes

switch

Node	"e20523194ac68b8e"
Type	switch

show more
Export the selected nodes, or the current tab with ctrl-e

DEBUG NODE:



Edit debug node

Output ▾ msg. payload

To
□ debug window
□ system console
□ node status (32 characters)

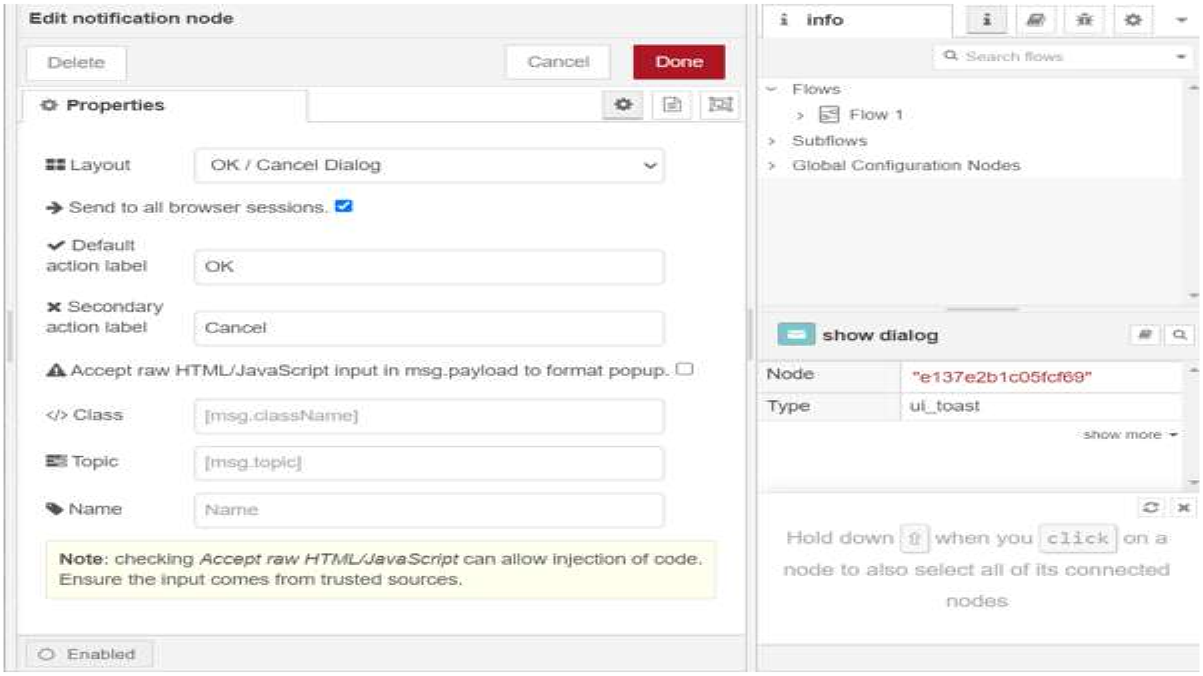
Name: Name

msg.payload

Node	"e871fc9118467379"
Type	debug

Dragging a node onto a wire will splice it into the link

NOTIFICATION NODE:



Edit notification node

Layout: OK / Cancel Dialog

Send to all browser sessions. ☑

Default action label: OK

Secondary action label: Cancel

Accept raw HTML/JavaScript input in msg.payload to format popup. □

Class: [msg.className]

Topic: [msg.topic]

Name: Name

Note: checking *Accept raw HTML/JavaScript* can allow injection of code. Ensure the input comes from trusted sources.

show dialog

Node	"e137e2b1c05fcf69"
Type	ui_toast

Hold down ⇧ when you click on a node to also select all of its connected nodes

GITHUB LINK

https://github.com/IBM-EPBL/IBM-Project-27722-1660063606

DEMO VIDEO LINK

https://drive.google.com/file/d/1TOHuebob8GRzfcOhNQB70x7g4CFWB21x/view?usp=share_link