IOT Based Safety Gadget for Child Safety Monitoring and Notification

IBM NALAIYATHIRAN

TITLE : IOT Based Safety Gadget for Child Safety Monitoring and Notification

DOMAIN NAME: INTERNET OF THINGS

TEAM ID: PNT2022TMID24179

TEAM LEADER: BASINENI SANTHOSH KUMAR

TEAM MEMBERS: BELLAMKONDA VISHNU VARDHAN
AMULURU PAVAN
ALLAMPATI KARUYNA KUMAR REDDY

MENTOR NAME: PADMANABAN

CONTENT

1.INTRODUCTION:

One of the things we parents fear most for our children is that they will be unintentionally and unnecessarily hurt.

Skinned knees and incidental boo-boos aside, the environment we create for them is the one they live in, and though we can't wrap them in bubble wrap till they're 21, there are many things we can do to help protect them from preventable injuries.

Some of these things are educational, others observational, and still others involve safety equipment or choices about furniture, or positions of items in the household.

1.1 Project Overview:

Basically, children cannot complain about abusements 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.

Since to prevent children before being attacked, an autonomous real- time monitoring system is necessary for every child out there.

In this system, the collected values from every sensor like temperature sensor, pulse rate detection sensor, metal detection sensor, and the location value from GPS are used to detect the status of the child and alerts the respective guardians using GSM accordingly.

1.2 Purpose:

It assists parents to monitor their children remotely .In case situations happen, notifications will be sent to parents so that actions can be taken. Through this child safety can be ensured and crime rate will be reduced. Parent’s concentrate to their works without worrying about their children.

2.LITERATURE SURVEY:

2.1 Existing problem:

Real-Time Child Abuse and Reporting System

In the existing system, we use a voice recognition module in which the alert commands from the child are stored and kept for further reference. If the same child delivers the same command, it will compare with the alert command which was previously stored and sets an emergency level according to the alert command. The GSM has a SIM which is used to send an alert message or an alert call to the trusted peoples. GPS is used to track the live location and it is used when needed. The server will search the respective device ID from the database and search for respective contacts according to that device ID and helps in alerting the registered guardians.

The disadvantage of this project are,

i. The child could not produce the exact alert command during a panic condition. ii. The command produced may not match with the previously stored command. iii. This project requires manual intervention.

2.2 References:

[1] Authors: M Nandini Priyanka, S Murugan, K. N. H. Srinivas, T. D. S. Sarveswararao, E. Kusuma Kumari.

Title: Smart IoT Device for Child Safety and Tracking. Published in: 2019 IEEE.

The system is developed using Link-It 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 emergency.

Merits: The parameters such as touch, temperature & heartbeat of the child are used for parametric analysis and results are plotted for the same.

Demerits: To implement the IoT device which ensures the complete solution for child safety problems.

[2] Authors: Akash Moodbidri, Hamid Shahnasser Title: Child safety wearable device. Published in: 2017 IEEE.

The purpose of this device is to help the parents to locate their children with ease. At the moment there are many wearables in the market which helps to track the

daily activity of children and also helps to find the child using Wi-Fi and Bluetooth services present on the device.
Merits: This wearable over other wearable is that it can be used in any phone and it is not necessary that an expensive smartphone is required and doesn’t want to be very tech savvy individual to operate.
Demerits: As, this device’s battery gives short life-time. High power efficient model will have to be used which can be IoT Based Smart Gadget for Child Safety and Tracking
[3] Authors: Aditi Gupta, Vibhor Harit. Published in: 2016 IEEE.
Title: Child Safety & Tracking Management System by using GPS.
This paper proposed a model for child safety through smart phones that provides the option to track the location of their children as well as in case of emergency children is able to send a quick message and its current location via Short Message services.
Merits: The advantages of smart phones which offers rich features like Google maps, GPS, SMS etc.
Demerits: This system is unable to sense human behavior of child.
[4] Authors: Dheeraj Sunehera, Pottabhatini Laxmi Priya.
Title: Children Location Monitoring on Google Maps Using GPS and GSM.
Published in: 2016 IEEE.
This paper provides an Android based solution for the parents to track their children in real time. Different devices are connected with a single device through channels of internet. The concerned device is connected to server via internet. The device can be used by parents to track their children in real time or for women safety. The proposed solution takes the location services provided by GSM module. It allows the parents to get their child’s current-location via SMS. Merits: A child tracking system using android terminal and hoc networks. Demerits: This device cannot be used in rural areas

2.3 Problem Statement Definition:

Child tracker helps the parents in continuously monitoring the child's location.

Child They can simply leave their children in school or parks and create a geofence around the particular location.

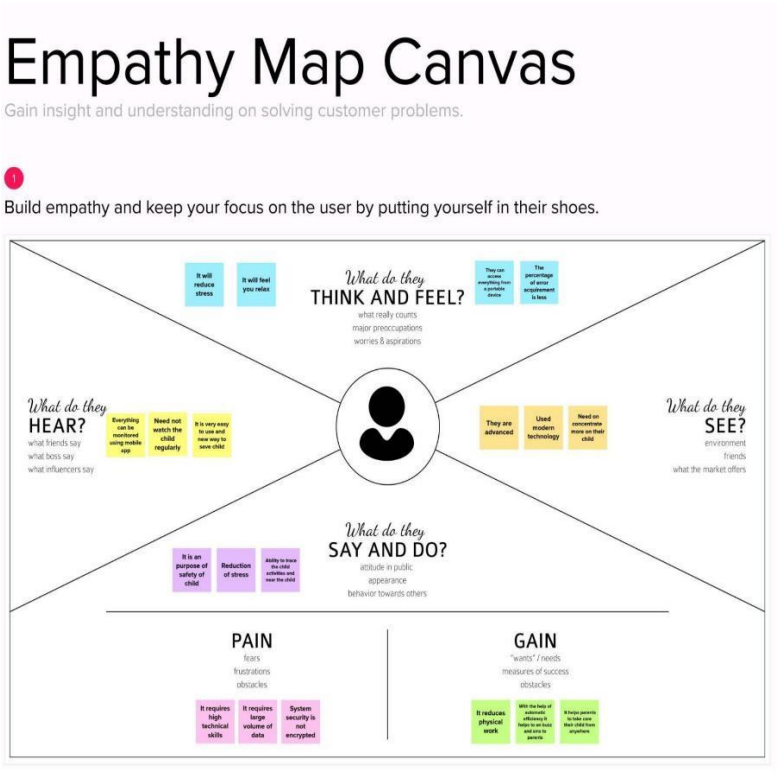
By continuously checking the child's location notifications will be generated if the child crosses the geofence.

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.

3. IDEATION AND PROPOSED SOLUTION:

3.1 Empathy Map Canvas



3.2 Ideation and Brainstorming

In this present era, most of the wearable devices today are designed based on the location, activity, temperature, pressure, etc of the child and inform the parents via GPS. Therefore it is intended to use voice call as the way of communication between the parent mobile and child’s wearable device. The system operates on the microcontroller board and the functions of sending and receiving notifications, calls, voice messages via GPS.

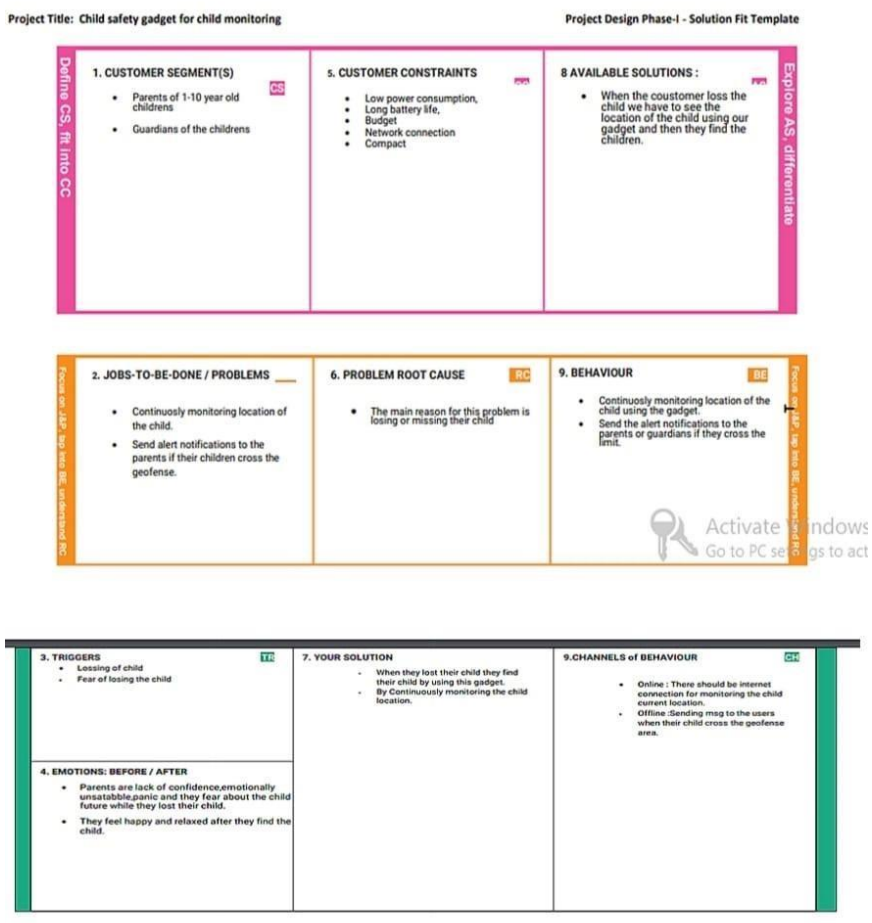
3.3 Proposed Solution

1.	Problem Statement(Problem to be solved)	1. Child tracker helps the parents in continuously monitoring the child's location. 2. They can simply leave their children in school or parks and create a geofence around the particular location. 3. By continuously checking the child's location notifications will be generated if the child crosses the geofence. 4. 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
2.	Idea/Solution description	1. Without any Interruption of miscellaneous signal for better performance. 2. It should be compact and it is mostly a wearable one.
3.	Novelty/ Uniqueness	1. If the children cross the geofence area , Snapshot of the current location of the child is notify to the parents. 2. Without internet the device should be communicate within a short range.

S.No.	Parameter	Description

4.	Social Impact/ Customer Satisfaction	1. Parents do their work properly and peacefully. 2. Without disturbing the parents work ,only the device alert when the child crosses the geofence.
5.	Business Model(Revenue Model)	1. The cost of the device is satisfactory to both Customer and the manufacturer.
6.	Scalability of the Solution	2. To made a separate device for control the gadget. 3. It transmits the messages even in a hill regions.

3.4 Problem Solution Fit



4 REQUIREMENT ANALYSIS:

4.2 Functional Requirements:

Following are the functional requirements of the proposed solution.

FR No.	**Functional Requirement (Epic)**	**Sub Requirement (Story / Sub-Task)**
FR-1	Communicate and exchange information to provide server for user	1. To monitor the children's location continuously in schools or parks. 2. Alert the parent if the child crosses the geofence through SMS.
FR-2	Continuous monitoring	1. Create the geofence around child location. 2. Continuously monitoring the child location. 3. Notifications send when child cross the geofence and chid face any issues.
FR-3	User requirement	1. Easily upgrade to any environments. 2. Easy to handle. 3. Gives more accuracy. 4. Low power consumption.

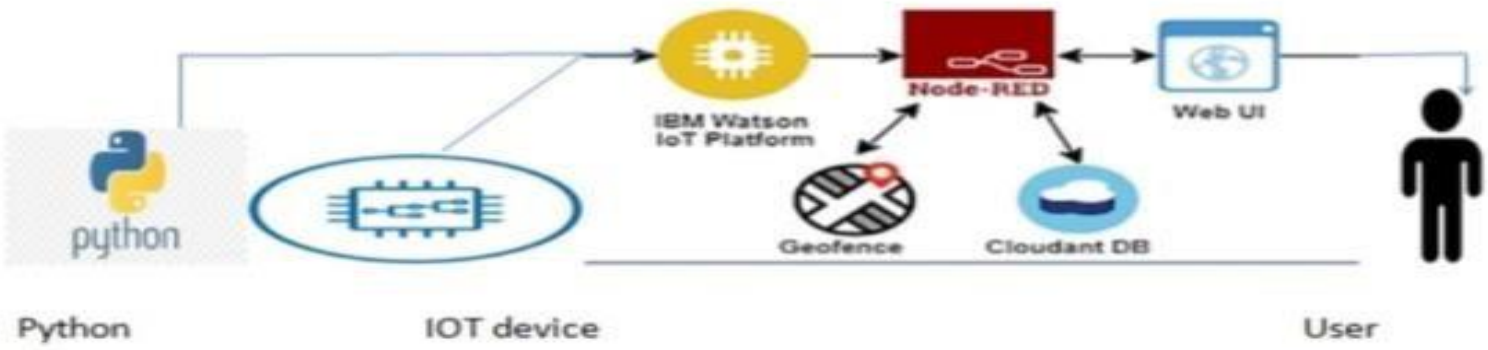
FR-4	Mandatory	1. The system will send the detail of location information the system via 3G network or Wi-Fi. 2. Accuracy of location is important. 3. The system should be scalable. 4. The entire location data will be stored
FR-5	Testing Set thegeofence.	1. The device is kept together with the children. 2. Create geofence around the child location in school or parks, if child crosses the geofence notify to the parents 3. Notifications sent in the form of SMS.

4.3 Non-Functional Requirements :

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	1. High usability of user experience design for user, 2. Which is usable for finding the children if they lost.
NFR-2	**Security**	1. The system can accessed by authorized persons only.
NFR-3	**Reliability**	1. Monitoring the location continuously and easy to upgrade the system .
NFR-4	**Performance**	1. The performance should be more effective and efficient. 2. The location data will be stored.
NFR-5	**Availability**	If we are going to upgrade the system or make any changes in the the system it will not take much time to recovery.
NFR-6	**Scalability**	1. The website traffic limit must be scalable enough to 2. support users at a time.

5 PROJECT DESIGN

5.1 Data Flow Diagrams:

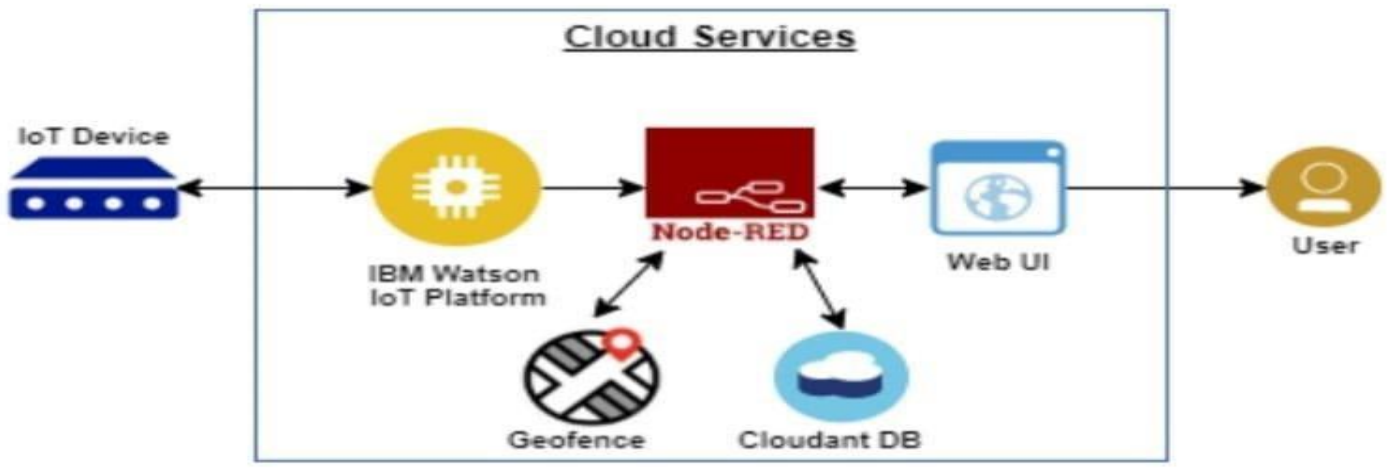


Python Script

IOT device

User

5.2 Technical Architecture:



* Feed the data from the GPS placed in the Device to the web interface.

*The data will display in the web page of the authority(user).

*The collected data is sent to the data base, where the collected location and predefined geophone location are checked and monitored if the child Cross the geophone notification send to the parents.

*The location data is provided to the cloud service and stored

*The authority monitors the web page continuous collect the

S.no	Characteristics	Description	Technology
1.	Open source Frameworks	Tracking the location of children	Random data in pythor script
2.	Security Implementation	Device ID,IBM cloud and Watson account.	Eg.SHA-256,Encryptions,IAM contro OWSAP etc.
3.	Scalable Architecture	Upgrade	IBM cloud
4.	Availability	The app contains the Location data of the children.	GPS, Python script
5.	Performance	The system continuously update the location data and if the children cross the geofence it will show alert.	Mobile app,Web UI

location data and

Send the alert to the authority

Components & Technologies

Application Charactersitics:

5.3 User Stories

Many parents don’t have an easy solution to this challenge. Cell phones are essential communication tools for adults, but in the hands of children, they may prove to be an expensive distraction. Plus, some communication tools come with potential security threats that parents want to avoid. IoT technology offers a simple solution to this dilemma: child tracking systems that are designed with communication and safety in mind.Child tracking systems and communication tools need to be completely secure from end-to-end, leaving parents with full control over who communicates with their children. Many existing smartphone apps cannot retain this level of security, in part, due to the many endpoint vulnerabilities that exist in a smartphone. While ease of communication and security are essential features in a child tracking system, these devices must also be sufficiently durable to keep up with a young child’s lifestyle. Kids run, jump and climb playground equipment all day, and they don’t want to become overburdened by bulky hardware. You can’t use a child tracking system that’s too small and delicate either, as children could lose the device or break it. Devices like Okie-talkie are designed specifically for a child’s lifestyle, featuring portable,

rugged modules that don’t break easily. The technology used to connect to the network is also designed for daily use.

6 .PROJECT PLANNING AND SCHEDULING:

6.1 Sprint Planning and Estimation:

SPRINT-1:

It indends to develop a python code to publish the location of the child.

and in command prompt: pip install wiotp-sdk pip

Step-1 install ibmiotf

Install python softwar e version 3.7.4

Step-3 Develop a python script to publish the location details to the IBM IoT platform

Step-2 Install Watso n IoT Python SDK to connec t to

IBM Watson IoT Platform using python code. Give the following

SPRINT 2:

Step-1

To create a IBM Cloud Services Step-2

Create IBM Watson IoT Platform And Device

IBM Watson IoT platform acts as the mediator to connect the web applic ation to IoTdevice, so create the IBM Watson IoT platform.

In order to connect the IoT device to the IBM cloud, create a device in the IBM Watson IoT platform and get the device credential

Step-3

Integrate the python code with IBM Watson IoT Platform. Create board in IBM Watson IoT Platform

SPRINT 3:

Creating a Web Application through which the user interacts with the dev ice & Create a Database in Cloudant DB

Step-1

Create Node-RED service Step-2

Connect the node in workflow Step-3

Install palette node-red-dashboard node-red-contrib-scx-ibmiotapp Step-4

Deploy the flow Step-5

To build a Dashboard

Step-6

Launch the cloudant DB Step-7

Create a database to store the location data

SPIRINT 4:

Develop the app and integrate with device. Step-1

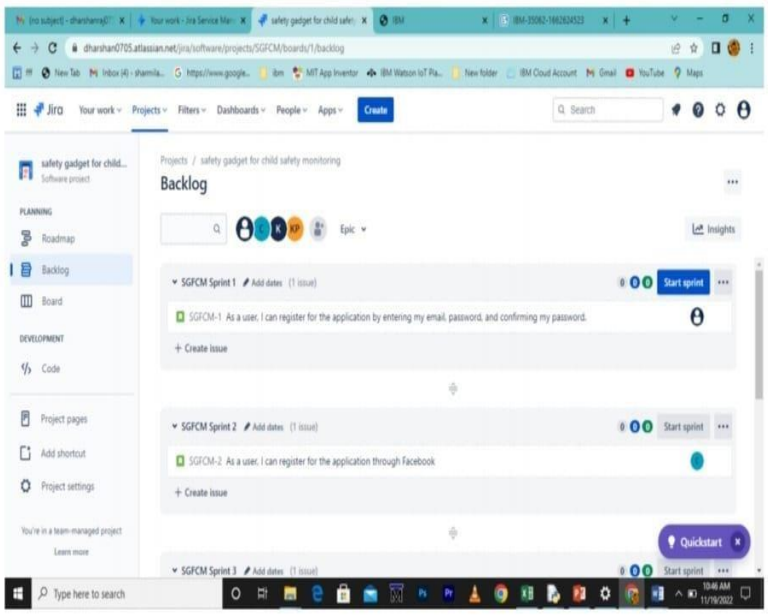
Develop the app (MIT inventor) Step-2

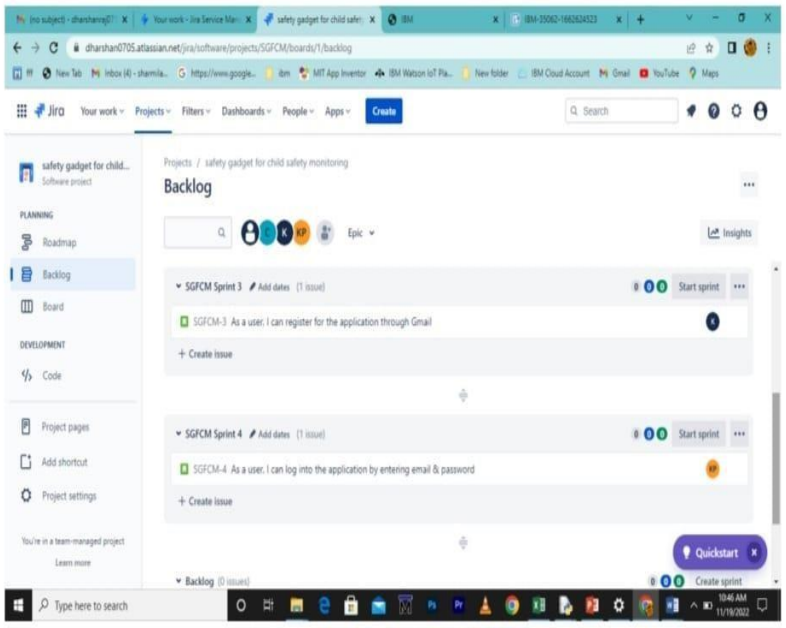
In Node-RED using http node connect the Node-RED dashboard with mit app.

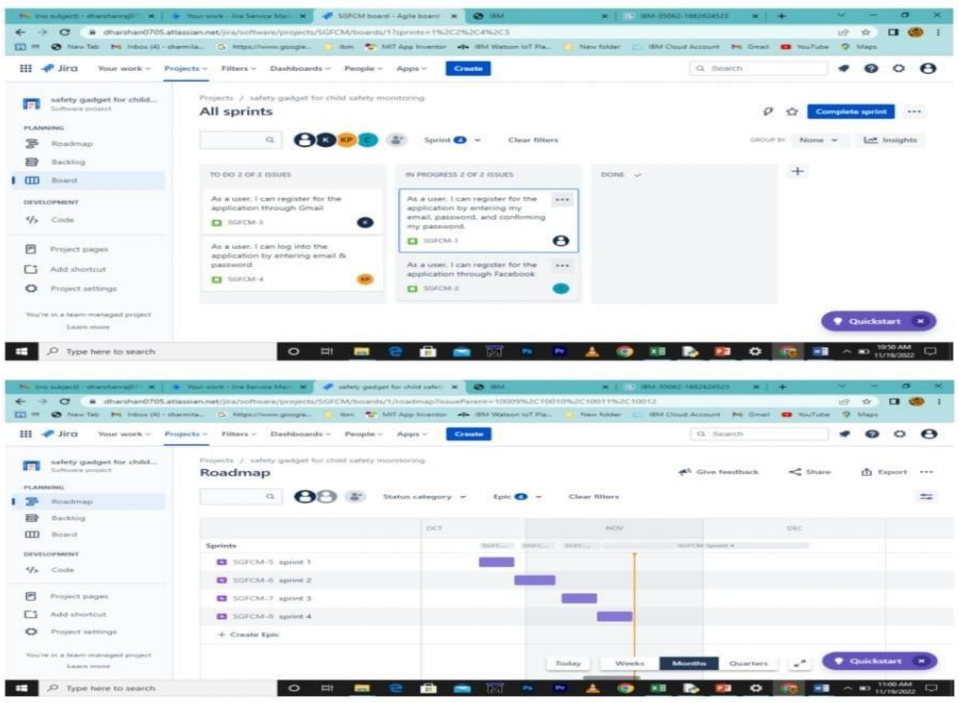
Step-3

With the help of app user get the location details and give the command through app

6.2 Report from JIRA







7.CODING AND SOLUTIONING:

```
import time  import sys

importibmiotf.applicati

on

importibmiotf.device

import random

#Provide your IBM Watson Device Credentials

  organization    =    "933n2d"    deviceType    =

  "koushik47"   deviceId  =  "07"   authMethod  =

  "token"   authToken = "87654321"

#apikey {a-illza1-mbdxqo6z0s}29

 #api token {zSYzISuAWF&F_x7GkT}

  try:

        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,

  "auth-method": authMethod, "auth-token": authToken}

      deviceCli = ibmiotf.device.Client(deviceOptions

      #.....................................
except Exception as e:

      print("Caught exception connecting device: %s" %

      str(e))

      sys.exit()
```

```
# Connect and send a datapoint "hello" with value "world" into the cloud as an
event of type "greeting" 10 times
print("power on ")
print("checking connection to wastoniot...")
time.sleep(2)
deviceCli.connect()
print("dear user ... welcome to IBM-IOT ")
print("i can provide your children live location and
temperature ")
print()
```



```
name=str(input("enter your child name:"))
while True:
temperature=random.randint(20,85)#random
temperature for your child
latitude=random.uniform(12.1295314,12.1335
137)#random latitude for your child
longitude=random.uniform(78.1955059,78.19
86357)#random
longitude for your child a="Child
   inside the geofence" b=" Child
```

```
outside the geofence" c="High
temperature" d="Low temperature"
x={'your_child_zone':a}
y={'your_child_zone':b}
z={'temp_condition':c}
w={'temp_condition':d} data = {
'temp' : temperature, 'lat':
latitude,'lon':longitude,'name':name }
  #print data
 defmyOnPublishCallback():
```

```
print ("Published Temperature = %s C" % temperature, "latitude = %s %%" %
latitude, "longitude = %s
%%" % longitude, "to IBM Watson")


 print("\n") success =

deviceCli.publishEvent("IoTSensorgpsdata", "json", data,

qos=0,on_publish=myOnPublishCallback)32

if latitude>=12.1303598 and latitude<=12.1321095 and longitude >=78.1967589
and longitude
<=78.19820833:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=x,qos=0,on_publish=

myO nPublishCallback)  print(x) print("\n")

else:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=y,qos=0,on_publish=

myO nPublishCallback)  print(y)

print("\n")
if (temperature>=40):

deviceCli.publishEvent("IoTSensorgpsdata","json",data=z,qos=0,on_publish=

myO nCallback)  print(z) print("\n")  else:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=w,qos=0,on_publish=my

OnP ublishCallback)  print(w)

print("\n")

if not success:
```

print("Not connected to

IoTF") print("\n")

time.sleep(1)

Disconnect the device and application from

the cloud deviceCli.disconnect()

7TESTING:

7.1 Test case:

SL.NO	INPUT	OUTPUT	RESULT
01.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed
02.	Latitude, Longitude Temperature	Inside the geofence, Temperature high	Passed
03.	Latitude, Longitude Temperature	Outside the geofence, Temperature low	Passed
04.	Latitude, Longitude Temperature	Outside the geofence, Temperature low	Passed

05.	Latitude,[35] Longitude Temperature	Inside the geofence, Temperature low	Passed
06.	Latitude, Longitude Temperature	Inside the geofence, Temperature high	Passed
07.	Latitude, Longitude Temperature	Outside the geofence,	Passed
08.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed
09.	Latitude, Longitude Temperature	Inside the geofence, Temperature high	Passed
10.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed
11.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed

12.	Latitude, Longitude Temperature	Outside the geofence, Temperature low	Passed
13.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed
14.	Latitude, Longitude Temperature	Inside the geofence, Temperature high	Passed

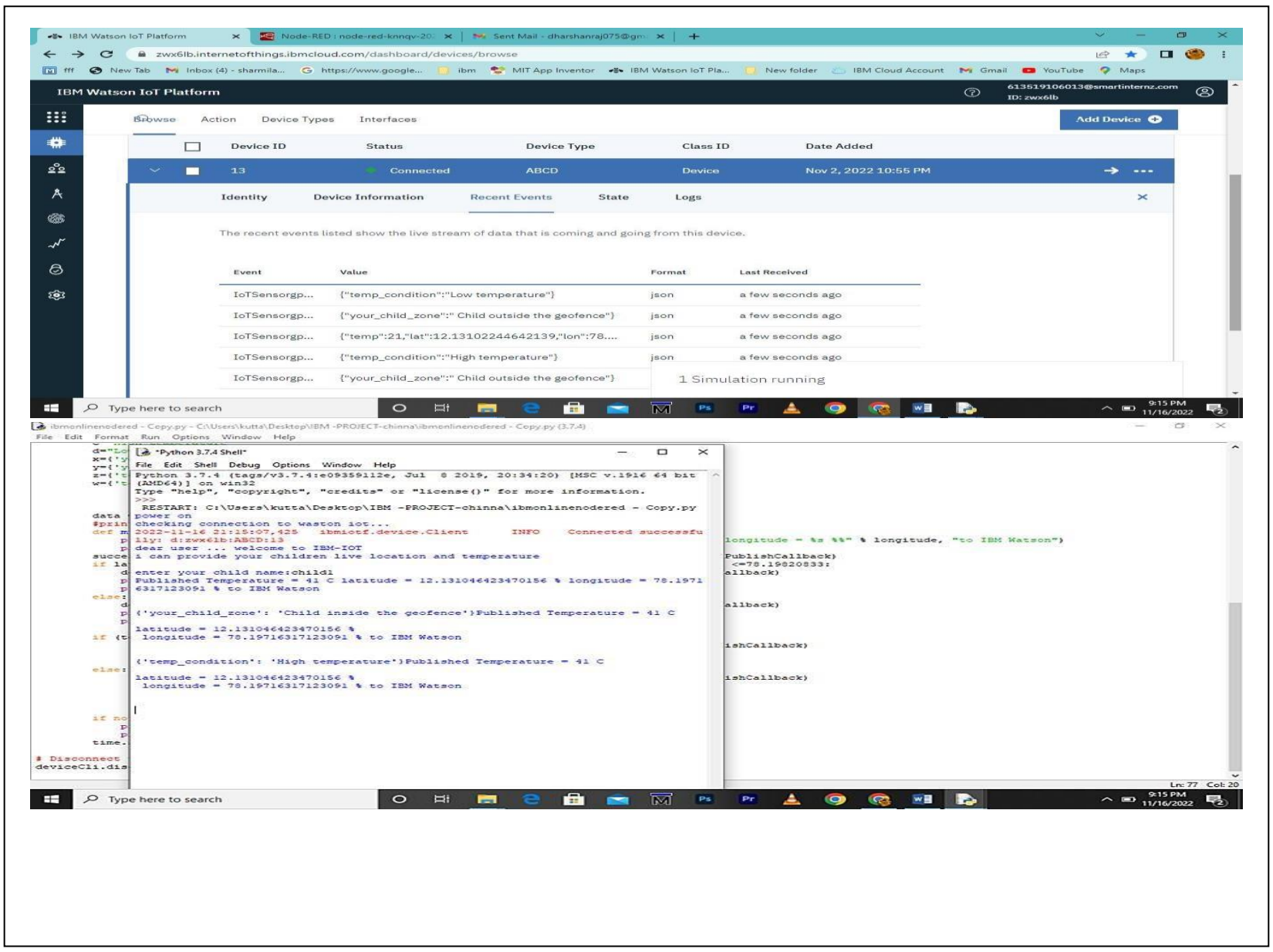
Temperature low

15.	Longitude Temperature	Outside the geofence, Temperature high	
16.	Latitude, Longitude Temperature	Outside the geofence, Temperature low	Passed
17.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed

Latitude, Passed

18.	Latitude, Longitude Temperature	Inside the geofence, Temperature high	Passed
19.	Latitude, Longitude Temperature	Outside the geofence, Temperature low	Passed
20.	Latitude, Longitude Temperature	Outside the geofence, Temperature high	Passed
21.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed

22.	Latitude, Longitude Temperature	Inside the geofence, Temperature high	Passed
23.	Latitude, Longitude Temperature	Outside the geofence, Temperature low	Passed
24.	Latitude, Longitude Temperature	Inside the geofence, Temperature low	Passed
25.	Latitude, Longitude Temperature	Outside the geofence, Temperature low	Passed



8.2 User Acceptance Testing:

Purpose of Document:

The purpose of this document is to briefly explain the test coverage and open issues of the IOT
Based Safety Gadget For Child Safety Monitoring and Notification project at the time of the release to
User Acceptance Testing (UAT).

Defect Analysis:

This report shows the number of resolved or closed bugs at each severity level, and how they were resolved

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	5	3	2	3	13
Duplicate	1	0	0	0	1
External	2	2	0	1	5
Fixed	6	5	3	10	24
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won’t Fix	0	0	2	1	3
Totals	14	10	9	16	49

Test Case Analysis

This report shows the number of test cases that have passed, failed, and untested

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	30	0	0	30
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	9	0	0	9
Final Report Output	4	0	0	4
Version Control	2	0	0	2

8.RESULTS:

The child safety wearable device can act as a smart device. It provides parents with the real-time location, surrounding temperature, SOS light along with Distress alarm buzzer for their child's surroundings and the ability to locate their child or alert bystanders in acting to rescue or comfort the child. The smart child safety

wearable can be enhanced much more in the future by using highly compact Arduino modules such as the

Lily Pad Arduino which can be sewed into fabrics. Also, a more power efficient model will have to be created which will be capable of holding the battery for a longer time.

9 ADVANTAGES AND DISADVANTAGES

Advantages:

- Save the life of the children.
- Parent’s do their work peacefully without worrying about their children.
- Continously monitoring the children.
- Saves time.
- Recovery of the children is easy,if the children lost.

Disadvantages:

1. Young Children may refuse to cooperate unless allowed to play with their gadgets.
2. Easily misusing the device.
3. No water proof.

10 CONCLUSION:

The child tracking system that helps parents track the movements of children with the help of GPS technology. The entire location data is stored in database. This proposed app can shows the whether the children inside the geofence or outside the geofence to the parent’s mobile . Even if the software is not running, the details are shown. It is because location access is available in the background and the software performs well on the mobile device. Based on the availability of the parent user, additional geofences may be required. Performance Requirements are summarized as follows: login,

Location status, temperature ,Live on map etc. The system shall allow the user to create and/or log in

to an account. The system shall allow the user to find the exact location of the children using GPS.

The system shall allow the user to track the current location of the children using GPS.

11. FUTURE SCOPE:

1. Childs surrounding can be located with the help of accurate and precise real time location.
2. Surrounding environment temperature, SOS light along with Distress buzzers are provided in this system.
3. If child crosses the geofence ,call goes to the registered mobile number’s.
4. This gadgets will be modified that has been suitable for all environments.

12. APPENDIX:

Python code:

```
import time  import sys

importibmiotf.applicati

on

importibmiotf.device

import random

 #Provide your IBM Watson Device Credentials

 organization = "933n2d" deviceType =

 "koushik47"  deviceId = "07"  authMethod =

 "token"  authToken = "87654321"

 #apikey {a-illza1-mbdxqo6z0s}
```



```
 #api token {zSYzISuAWF&F_x7GkT}

  try:

deviceOptions = {"org": organization, "type": deviceType, "id": deviceId,

"authmethod": authMethod, "auth-token": authToken} deviceCli =

ibmiotf.device.Client(deviceOptions)

      #.....................................
 except Exception as e:

      print("Caught exception connecting device: %s" %
```

```
        str(e))
        sys.exit()
# Connect and send a datapoint "hello" with value "world" into the cloud as an
event of type "greeting" 10 times
print("power on ")
print("checking connection to wastoniot...")
time.sleep(2)
deviceCli.connect()
print("dear user ... welcome to IBM-IOT ")
print("i can provide your children live location and
temperature ") print() name=str(input("enter your
child name:"))
while True:
temperature=random.randint(20,85)#random temperature for your child
latitude=random.uniform
(12.1295314,12.1335137)#random latitude for your child
longitude=random.uniform(78.1955059,78.1986357)
#random longitude for your child
    a="Child  inside  the  geofence"
    b=" Child outside the geofence"
```

```
c="High temperature" d="Low
temperature"
```



```
x={'your_child_zone':a}
y={'your_child_zone':b}
z={'temp_condition':c}
w={'temp_condition':d}
```

data = { 'temp' : temperature, 'lat':

latitude,'lon':longitude,'name':name } #print data

defmyOnPublishCallback():

print ("Published Temperature = %s C" % temperature, "latitude = %s %%" % latitude, "longitude = %s
%%" % longitude, "to IBM Watson")

print("\n") success = deviceCli.publishEvent("IoTSensorgpsdata", "json", data, qos=0, on_publish=myOnPublishCallback)

if latitude>=12.1303598 and latitude<=12.1321095 and longitude >=78.1967589 and longitude
<=78.19820833:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=x,qos=0,on_publish=

myO nPublishCallback) print(x) print("\n") else:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=y,qos=0,on_publish=my

O nPublishCallback)

print(y)

print("\n")

if (temperature>=40):

deviceCli.publishEvent("IoTSensorgpsdata","json",data=z,qos=0,on_publish=

myOnPublishCallback)

print(z)

print("\n")

else:

deviceCli.publishEvent("IoTSensorgpsdata","json",data=w,qos=0,on_publish=my

OnP ublishCallback) print(w)

print("\n")

if not success:

print("Not connected to IoTF")

print("\n")

time.sleep(1)

Disconnect the device and application from the

 cloud

device

Cli.disconnect()

GitHub &Project demo link:

GitHub Link: **https://github.com/IBM-EPBL/IBM-Project-51503-1660980053.git**