Child Tracker with Emergency Notifications Using IBM Cloud

Team:

1. S.G.SURUPA : 19R11A0489
2. VENNA KAUSHIK RAM KOTESHWAR : 19R11A0495
3. VUNRENUKA ANUSHA : 19R11A0496
4. A.R.A.PRATHYUSHA : 19R11A0497
5. SUPREETH AVULA : 19R11A0499

Introduction:

Overview:

Child tracker helps the parents in continuously monitoring the child's location. They can simply leave their children in school or parks and create a geofence around the particular location. By continuously checking the child's location notifications will be generated if the child crosses the geofence. Notifications will be sent according to the child's location to their parents or caretakers. The entire location data will be stored in the database.

Purpose:

Child Trackers help parents feel safe. Children leave their households for numerous reasons .In event that parents cannot physically monitor and assure the safety of their children, child trackers help to bridge the gap between them.
A selection of child trackers contain a feature called geo-fencing. Geo-fencing allows parents to create a safety zones and boundaries for their children. When children are outside their safety zones, parents are notified immediately and action can be taken.

Literature Survey:

Existing problem:

Increasing rate of child kidnapping.

- The total number of missing children cases has been more than a lakh in each of the years from 2015 to 2019. This number is the highest since 2015 and has increased by over 16.3% over these five years.

Lack of tracking technology for child.

- The parent is hardly to keep a watch on their child without the use of technology, especially when the child is in the outdoor. The parent even cannot avoid the negligence that will make by children in the future day.

Limited application for child monitoring.

- There are very limited application available for tracking child when they are out of parents control and let kidnapping or missing cases occurred.

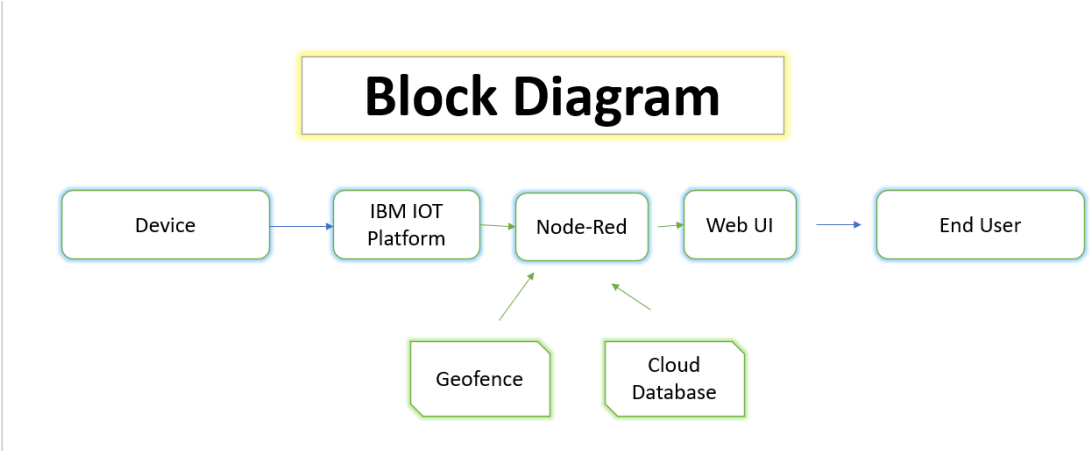
Our busy and stressful lives often mean that we can make mistakes. Despite paying attention to their activities or leaving them with vigilant babysitters, it is very much possible to make a mistake and lose track of the kids. Kids are also very energetic so they can easily sidestep you.

Proposed solution:

As a parent you need to give your child the freedom to explore the world around them, but at the same time, you may have concerns about their safety and protection.
In such cases, Child Tracking System helps you. You can define a perimeter for your kids when they go outside. As soon as they step beyond the defined area, the tracking software will alert you.
And also helps parents to access the child's location instantly without any difficulty.

Theoretical Analysis:

- The GPS coordinates of the child will be sent to IBM IoT platform
- Location can be viewed in the Web Application
- Parent can create a geofence in the web application
- Web application will check if the child is inside or outside the geofence
- Notifies the parents if the child goes out of the geofence.



Hardware:

- Smart watch with GPS connectivity

Software designing:

- Python IDLE
- IBM IOT Platform
- Node-Red Service
- Cloud Database
- Fast to SMS

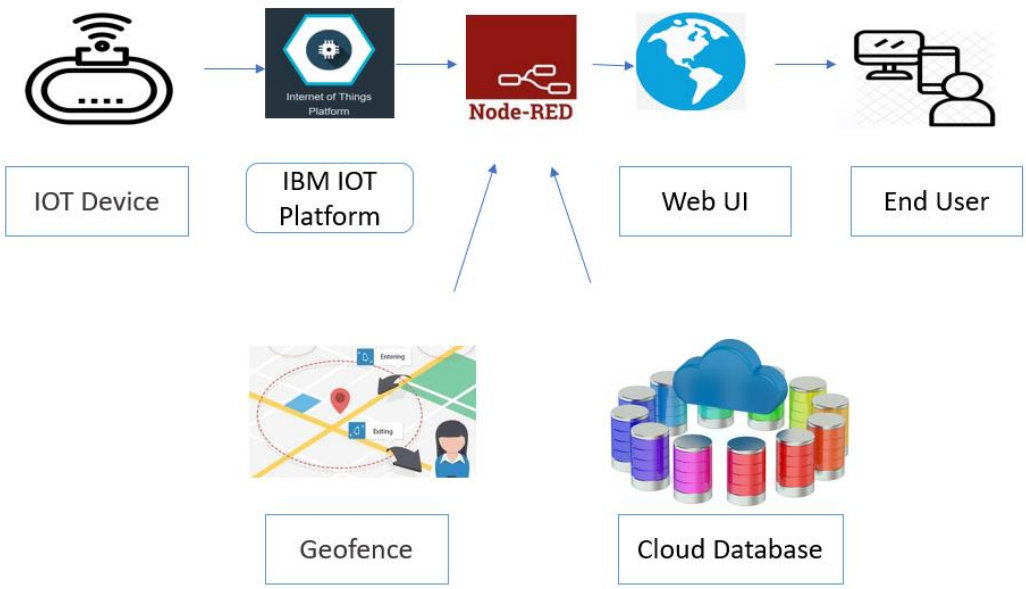
Experimental Investigations:

Co-ordinates of Location: Co-ordinates of child's location is the variable for Experimental Investigation. Example: Co-ordinates of child's location in the Geofence, and Co-ordinates of child's location out of the Geofence.

- ✓ **In area Location**: During the in area location of the Geofence, the child's location is monitored using the Web User-Interface.
- ✓ **Out area Location**: During the out area location of the Geofence, the child's parents will be alerted using a SMS.

Therefore, the project is tested with various co-ordinates of the child's location. And according to the locations I was able to track and remind the child's parent via SMS.

Flow Chart:



Result:

- ✓ I was able to create a Geofence for specified location of the child.
- ✓ I was able to track the location of the child.
- ✓ I was able to display child's location on the Web User-Interface.
- ✓ I was able to alert the parent with an SMS, whenever the child is out of the Geofence.
- ✓ And I was able to store the location in a Cloud database.

Advantages & Disadvantages:

- Application automatically operates location requests without user interaction because at times child may not have knowledge to update his location at map.
- Get Alerts (SMS), about reaching home.
- Child Tracker Makes the journey Safe.
- Alerts and notifications are an essential and beneficial feature of tracking device.

Applications:

- **Educational Institutions:** The management of an educational system will be able to track & analyse the student location. They can be notified easily whenever; the student is out of the campus premises.
- Burger King's "Whopper Detour," which involved building a 600-feet fence around McDonald's restaurants. Burger King encouraged its customers to go to McDonald's, but with a twist. As soon as they entered the "fence," they could unlock a deal for a one-cent Whopper burger on the Burger King app.
- **Social Networking**: Businesses use geofencing technology to personalize and improve the customer experience on social media platforms. Whenever a consumer enters a geofence, they can be triggered to take a photo, check-in, or engage with friends nearby. They can also be requested to leave a review, comment, or feedback when exiting the area.
- **Marketing:** Geofencing enables brands to target customers visiting particular shops and offer them incentives and better deals than the competition. Marketers can use the technology to target customers attending trade shows, events, and colleges. This approach is effective because, according to a report by TSheets, 53 percent of respondents have reported that mobile apps that send location-based notifications can sometimes be useful. These consumers respond to promotional alerts, confirming the reliability and applicability of geofencing.

Conclusion:

Child trackers & Geofence will be useful tool in future to monitor & guide children when the parents are out for work. Using these kinds of technology will be helpful in avoiding the children getting into danger while their parents are out. The rate of child accidents and tragedies will get down. This can prevent children and adults in attending malicious acts or being involved in them. This technology will be able to keep the children safe from the anonymous environment when their parents are out.

Future Scope:

Tech giants which have a say in children essentials can make way into this project. By adding this project to their range of products, they can able to provide customers safety for their children.

Bibliography:

- **https://www.lifewire.com/keep-track-of-your-kids-with-geofences-2487397**
- **https://www.grdjournals.com/uploads/conference/GRDCF/013/033/GRDCF013033.pdf**
- **Smart Bridge: Tutorials, Mentors.**

Source code:

```
import json
import wiotp.sdk.device
import time

myConfig = {
    "identity":{
        "orgId" : "ioeaum",
        "typeId": "ESP32",
        "deviceId": "54321"
    },
    "auth": {
        "token": "87654321"
    }
}
client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)
client.connect()

while True:
    name = "Thomas"

    #in area location
    #latitude = 17.4225176
    #longitude = 78.5488783

    #out area location
    latitude = 17.4219272
    longitude = 78.5488783
    myData = {'name':name, 'lat':latitude, 'lon':longitude}
    client.publishEvent(eventId="status", msgFormat="json",
data=myData, qos=0,onPublish=None)
    print("Data published to IBM IoT platform: ",myData)
    time.sleep(5)

client.disconnect()
```

project.py - C:\Users\A R A PRATHYUSHA\project.py (3.9.5)

File Edit Format Run Options Window Help

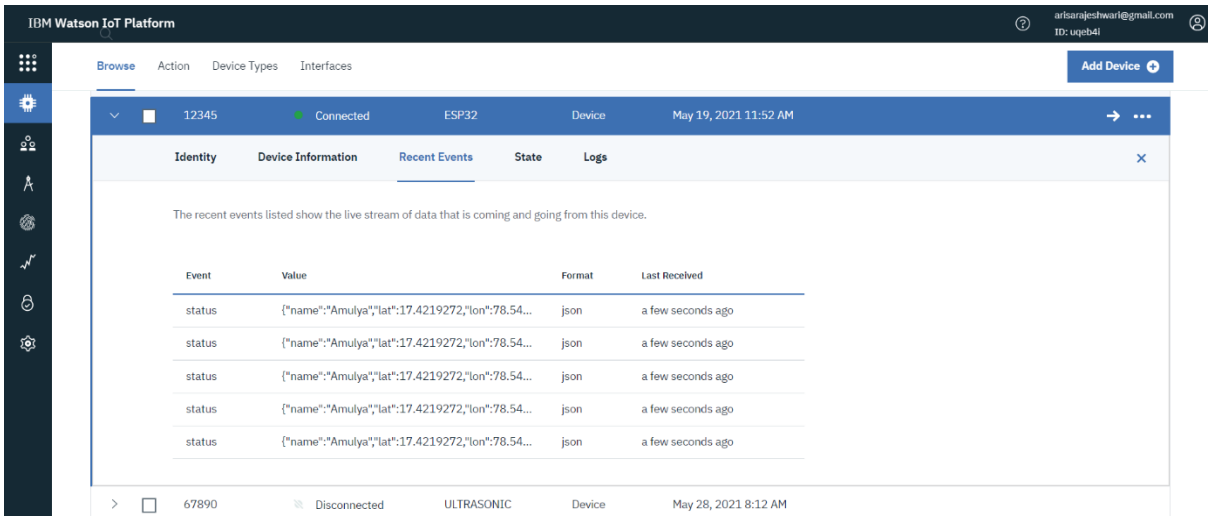
```python
import json
import wiotp.sdk.device
import time

myConfig = {
    "identity":{
        "orgId" : "uqeb4i",
        "typeId": "ESP32",
        "deviceId": "12345"
    },
    "auth": {
        "token": "9394891584"
    }
}
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
    name = "Smartbridge"

    #in area location
    #latitude = 17.4225176
    #longitude = 78.5488783

    #out area location
    latitude = 17.4219272
    longitude = 78.5488783
    myData = {'name':name, 'lat':latitude, 'lon':longitude}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,onPublish=None)
    print("Data published to IBM IoT platform: ",myData)
    time.sleep(5)

client.disconnect()
```

IDLE Shell 3.9.5

File Edit Shell Debug Options Window Help

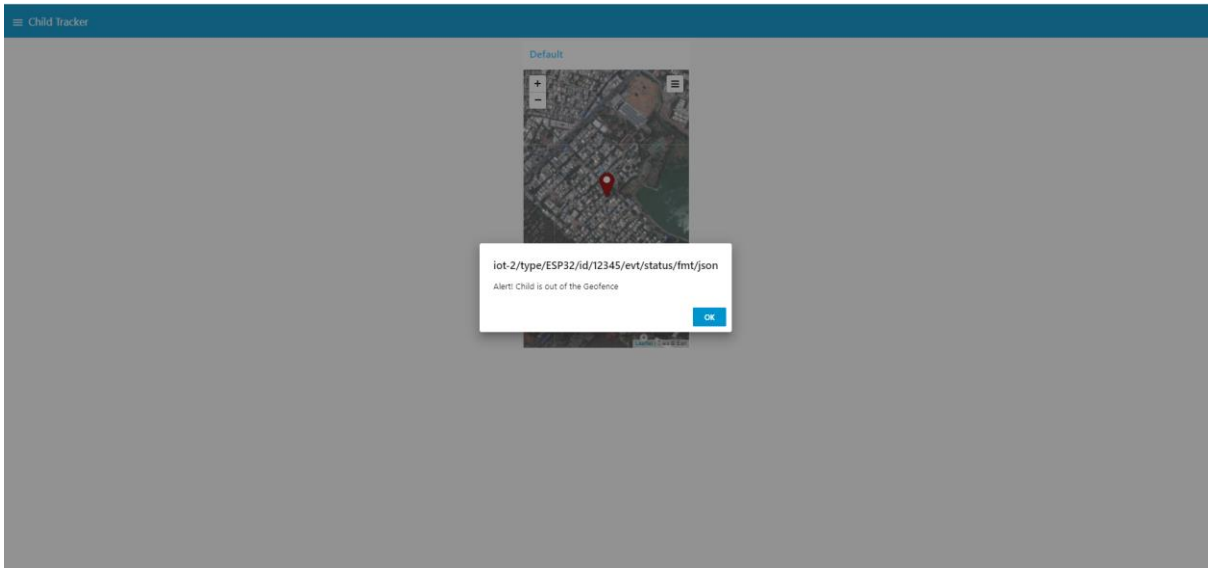
```
Python 3.9.5 (tags/v3.9.5:0a7dcbd, May  3 2021, 17:27:52) [MSC v.1928 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
================= RESTART: C:\Users\A R A PRATHYUSHA\project.py ================
2021-06-05 21:00:51,942   wiotp.sdk.device.client.DeviceClient  INFO    Connected successfully: d:uqeb4i:ESP32:12345
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
Data published to IBM IoT platform:  {'name': 'Amulya', 'lat': 17.4219272, 'lon': 78.5488783}
```

IBM IoT:

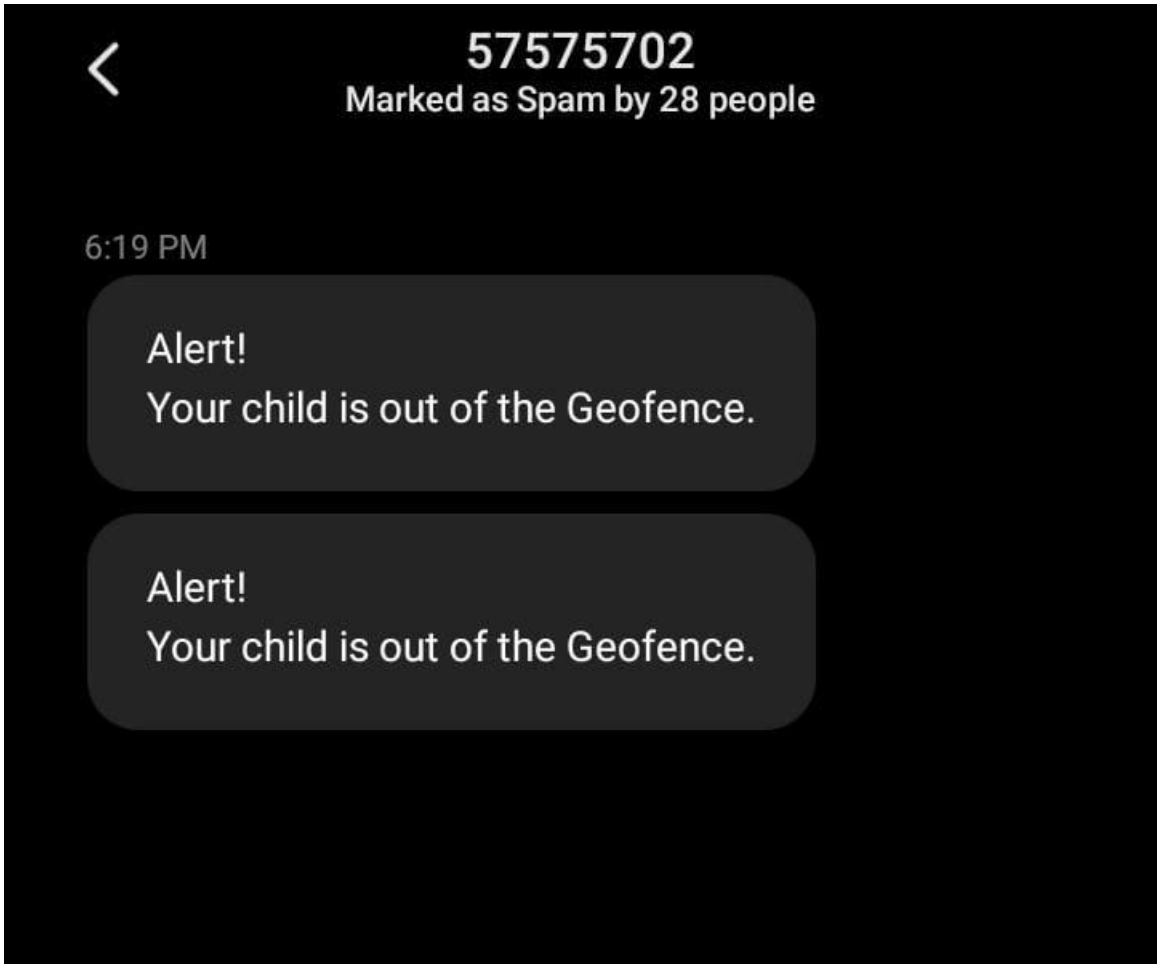


Debug Console of Node Red:

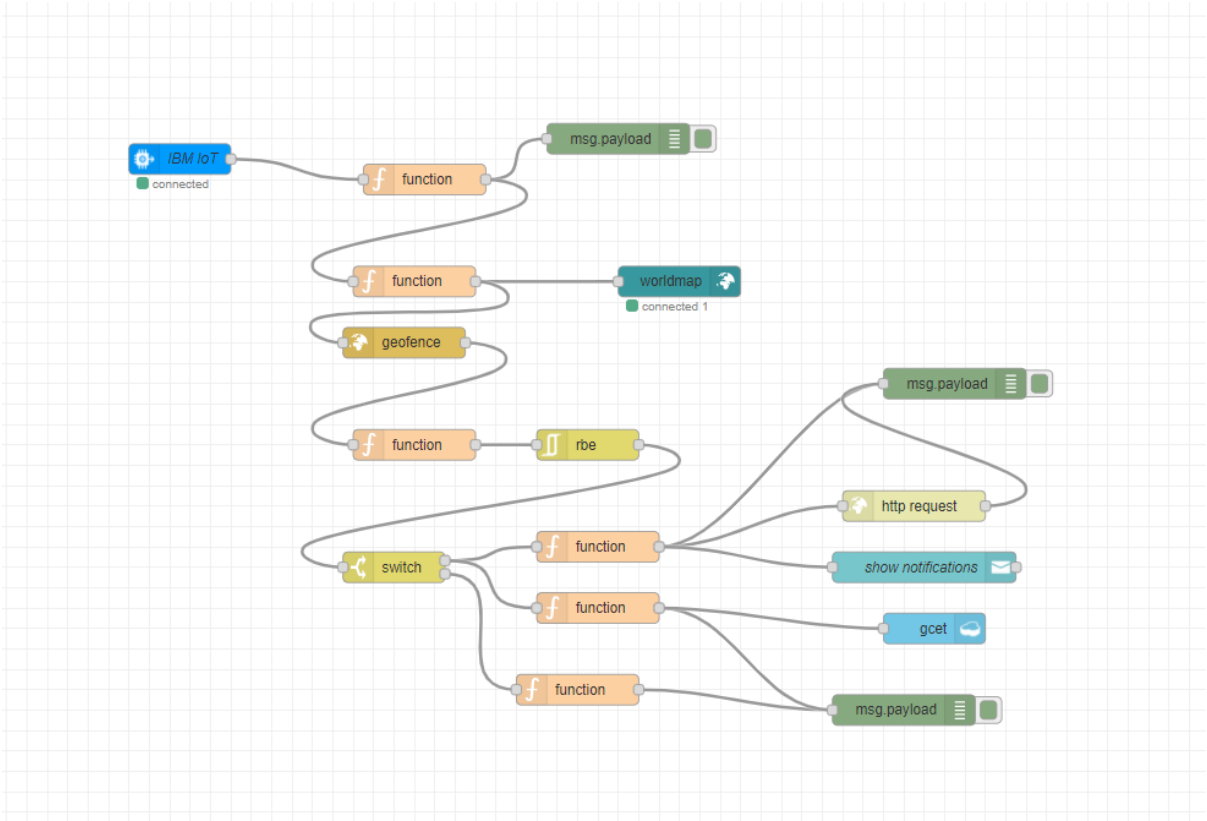
Web User-Interface:



Message Output:



Nodes:


IBM IoT
connected
function
msg.payload
function
worldmap
connected 1
geofence
function
rbe
msg.payload
http request
switch
function
show notifications
function
gcet
function
msg.payload