Smart Security And Safety Solutions Based On IoT For Large Industrial Plants

## By SAI KUMAR.D &

## A. SAINISCHAY

# INTRODUCTION

# Overview:

**What is IoT ?, IoT is creating a giant network where all the devices are connected to each other and providing them with the capability to interact with each other. This is driving the automation to a next level where devices will communicate with each other and make decisions on their own without any human interventions**

**The ‘Thing’ in IoT can be any device with any kind of built-in-sensors with the ability to collect and transfer data over a network without manual intervention. The embedded technology in the object helps them to interact with internal states and the external environment, which in turn helps in decisions making process.**

**IoT makes our lives Smarter , Easier and Quick**

**We can do lot of things with IoT for example reading the heart rate using a smart watch , Detecting the air contaminants using sensors , Reading the Temperature and humidity values using DHT senor etc**

# Purpose:

**So Now we are going to demonstrate that how that how the IoT Platform is useful and plays an important role in the case of the Emerging Large Scale Industries**

# Existing Problem:

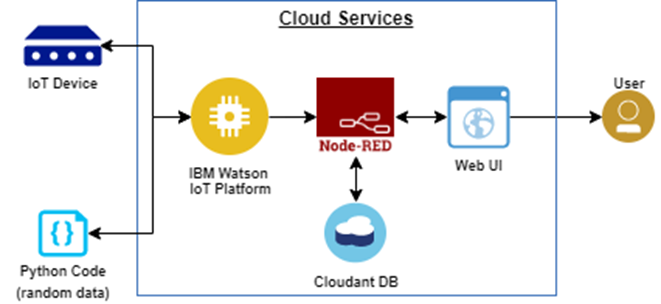
**Now the given Problem Statement is To Evacuate the Employees from the Radiation room who are staying in it for more Duration by sending an alert Notification to the Admin and to send the following Employee details to the Admin on Web ui and App**

# Proposed Solution:

**Now to the given problem statement we have solution , By using python code we are going enter employee details and These details are sent to the IBM Watson IoT Platform after compiling and from There we are going to export them to Node-Red Services in order to display the details in Web ui and in the form the Dashboard , Later we develop an app using MIT app inventor and the URLs. Finally the app is ready to extract the python details and Alert them**

# Theoritical Analysis:

# Block Diagram



# Hardware/Software Desinging

**As physical hardware is not available we take random values as input .We write a python code for Employee name, Employee id and duration in which the person has stayed in radiation room.So our project is to display alert. We then send code to the ibm cloud by giving our Device credentials that we created in the ibm cloud in the python code. The data printed in the Python will automatically get stored in ibm cloud. And we then create a node. For creating node red we have to first login in our ibm cloud account and click on cloud foundary apps and then click on the node red app servicesand then click on create new node editor. In node red we use some nodes to display the data in the url and as ui. We use some nodes like ibm cloud(in),few function nodes,ibm cloud(out),http in and out,and some text nodes,etc.After the completion of node red we check weather the data is been printed on the webpage or not by copying the url of our node red and paste it on an empty tab and replace red with data. Then we will observe our data which is been printed in python been displayed on the webpage. Similarly we check the same for visual representation of our program by replacing data to ui. So half of the process is been completed. Now the next step is to create an mobile app which is used to display the employee details and alert if a person exceeds the time. We use some of the blocks in mit app inventor for mobile screen display and then open blocks for the display to work. The main use of blocks is to configure the items displayed on the mobile screen to work. In the mobile display there will be employee details like his name ,id and duration for which he had stayed in the radiation room. If the employee exceeds the given amount of time an alert msg will be displayed in the web browser and an pop up will be appeared in themobile screens of the app which employees has. The main objective is to intimate the employee an alert message in order to be safe.**

# Experimental Investigations :

**As we know many deaths of workers have occurred due to minor mistakes by the company or industry. Workers work in different areas which are very hazardous.So the industry should provide minimum security for them. Let us take an example of a chemical factory. In that factory if there will be a leakage of any hazardous gasses and the industry is not able to recognize then it will result in many deaths of workers. But in our generationthere are many precautions taken in every industry.So in an industry if an employee stays more than the time limit in the radiation room then there should a message passed to the employee that he has to get out of the radiation room.So our project is to deliver that message to the employees.So our idea is that every employee has to have the appcreated so if the time exceeded there will be an alert message given to the employee. In this way industry provides safety for the employees. And also ensures the safety for the employee .**

# Flowchart

|  |
| --- |
| Employee Details  Through  python |

|  |
| --- |
| Creating node flow to get data |

|  |
| --- |
| Passing the details to IBM cloud account |

|  |
| --- |
| Displays alert popup if the duration has exceeded |

|  |
| --- |
| Creating a mobile app |

|  |
| --- |
| Using dash board notes for creating web app |

Result:

**Displays alert if the persons duration exceeded than the allowed time. This ensures the persons safety as well as his health. By displaying the alert the person will be able to know that he/she has to come out of the radiation room or any other room.**

# Advantages and disadvantages :

# Advantages:

**1. The person will be able to get out of the room without any injuries.**

**2. Company can assure safety to the workers.**

**3. Also the manager can see the alert and the employee details.**

**4. The best advantage of this is this the message will be sent directly to the mobile app.**

# Disadvantages:

1. **The alert message will be received by every employee which has the app which disturbs the other employees**

# Applications:

* **We can also Implement the same system in various industries like when the Toxic levels are high in the atmosphere of the any of the chamber of the industry (chemical industry) by using the sensors which check the toxic levels in that atmosphere and then using this system we can look after the safety of the employees**
* **We can also implement this system even when there is a fire accident in any of the plant of the industry the senors detect it and pushes a alert notification using this system to the employees**

# Conclusion:

**So by Making and Demonstrating this project we conclude that IoT is an Essential Part and plays an important role to run the Industrial World in Smarter , Easier and Quick**

# Bibliography:

**We have seen some reference videos for the Node red**

# Appendix:

# Source Code:

**import wiotp.sdk.device**

**import time**

**import random**

**from datetime import datetime**

**myConfig = {**

**"identity": {**

**"orgId": "uzkg5k",**

**"typeId": "ESP32",**

**"deviceId":"95505"**

**},**

**"auth": {**

**"token": "12345678"**

**}**

**}**

**def myCommandCallback(cmd):**

**print("Message received from IBM IoT Platform: %s" % cmd.data)**

**m=cmd.data**

**client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)**

**client.connect()**

**while True:**

**ID1=1234**

**ID2=5647**

**NAME1=str("Ajay")**

**delay=random.randint(0,20)**

**delay1=random.randint(0,20)**

**time.sleep(delay)**

**t1=datetime.now()**

**print("delay=",delay)**

**print("delay1=",delay1)**

**dt1 = t1.strftime("%d/%m/%Y %H:%M:%S") #ep1 entry time**

**time.sleep(delay)**

**t2=datetime.now()**

**dt2 = t2.strftime("%d/%m/%Y %H:%M:%S") #ep1 exit time**

**time.sleep(delay)**

**t0=datetime.now()**

**dt0 = t0.strftime("%d/%m/%Y %H:%M:%S") #ep2 entry time**

**time.sleep(delay1)**

**t3=datetime.now()**

**dt3 = t3.strftime("%d/%m/%Y %H:%M:%S") #ep2 exit time**

**time.sleep(delay1)**

**NAME2=str("Arun")**

**myData2={ 'ID1':ID1,'Name1':NAME1,'Entry\_dt\_of\_ep1':dt1,'Exit\_time\_of\_ep1':dt2,'Duration\_of\_ep1':delay,'ID2':ID2, 'Name2':NAME2,'Entry\_dt\_of\_ep2':dt0,'Exit\_time\_of\_ep2':dt3,'Duration\_of\_ep2':delay1}**

**client.publishEvent(eventId="status", msgFormat="json", data=myData2, qos=0, onPublish=None)**

**print("Published data Successfully: %s", myData2)**

**print("Detecting..........")**

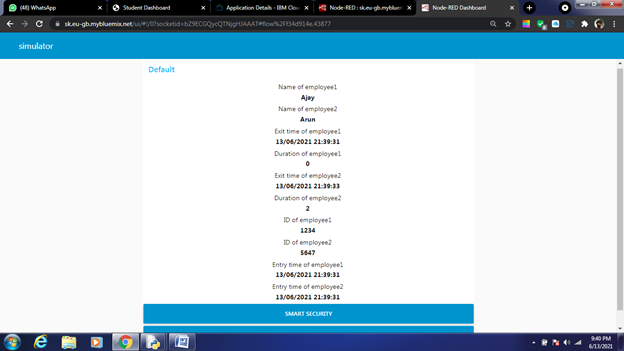
**client.commandCallback = myCommandCallback**

**time.sleep(1)**

**client.disconnect()**

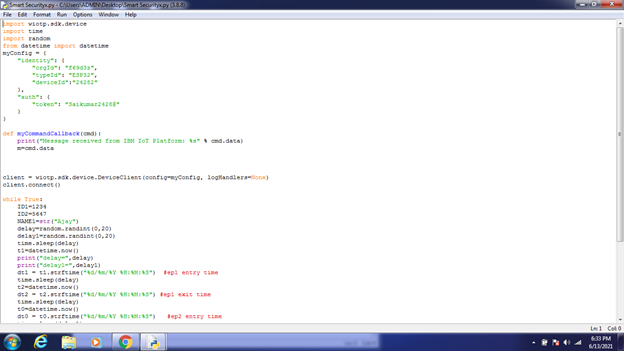
# Screeshots of the project in Order wise :

**1 UI OUTPUT:**

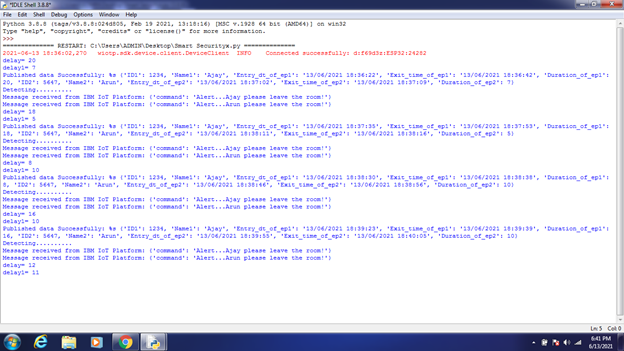


# Screeshots of the project in Order wise :

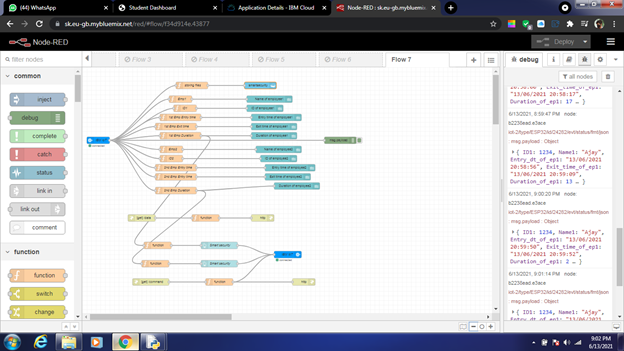
1. **Python Code**



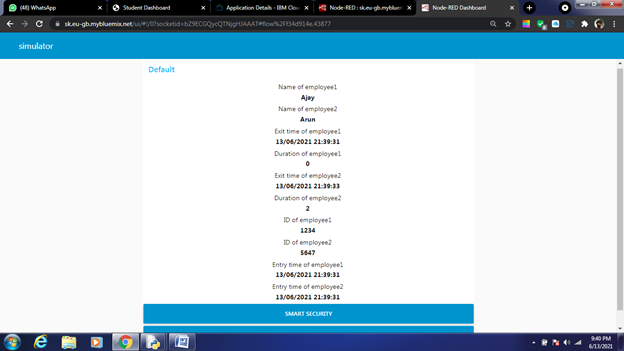
**2 PYTHON IDLE OUTPUT:**



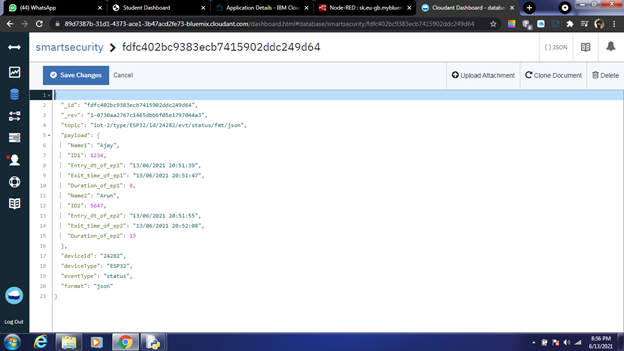
**3 . NODE RED OUTPUT:**



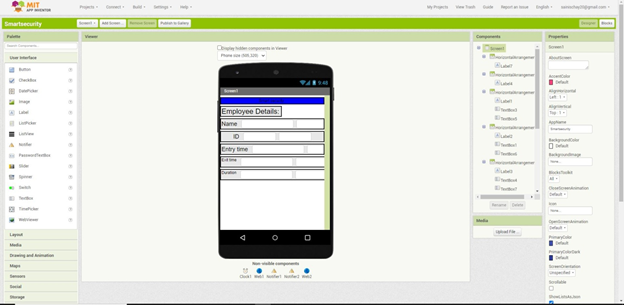
**4.WEB UI DASHBOARD OUTPUT:**



**5. STORING FILES IN CLOUDANT DB:**



**6.MIT APP DESIGN:**



**7.MIT BLOCKS SCREENSHOT:**

