

# REPORT

TEAM ID :	PNT2022TMID30898
PROJECT NAME :	PERSONAL ASSISTANCE FOR SENIORS WHO ARE SELF-RELIANT
MEMBERS:	NAVEEN KUMAR K (TEAM LEADER) VETRI VIGNESH A VISHVA R SURENDARANAND P

## 1. INTRODUCTION

### 1.1. Project Overview

Sometimes elderly people forget to take their medicine at the correct time. They also forget which medicine He / She should take at that particular time. And it is difficult for doctors/caretakers to monitor the patients around the clock. To avoid this problem, this medicine reminder system is developed. An app is built for the user (caretaker) which enables him to set the desired time and medicine. These details will be stored in the IBM Cloudant DB. If the medicine time arrives the web application will send the medicine name to the IoT Device through the IBM IoT platform. The device will receive the medicine name and notify the user with voice commands. Here Node-red connects the Web UI, Cloudant DB and IBM Watson IoT platform.

### 1.2. Purpose

The major purpose of this project is to help the elderly people who can assist themselves with the help of an voice alarm to take medicine at correct time. This avoids the irregular intake of medicine and keeps their body healthy. Some elderly people have oblivion to take medicine at right time but they can assist themselves in all other works these type of elderly people will make of this project to their best. This method of medicine alerts also avoids the need of personal care assistance expenses and man power.

## 2. LITERATURE SURVEY

### 2.1. Existing problem

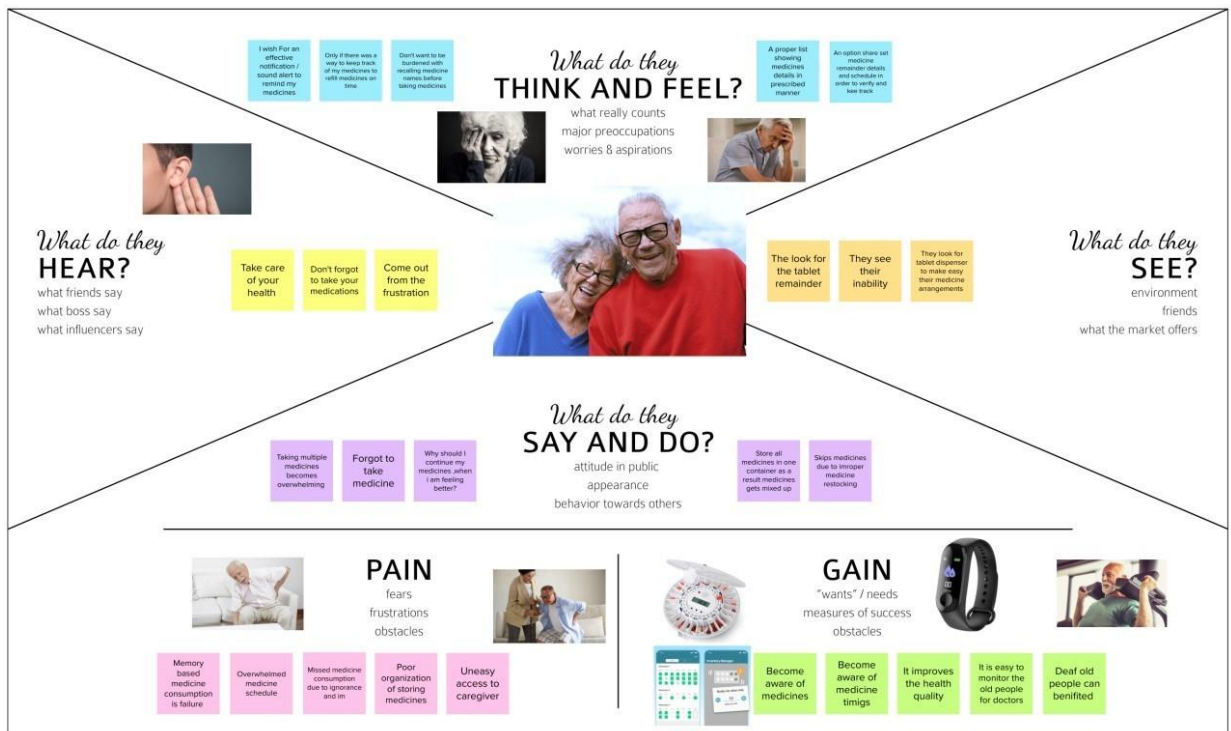
The aging of population is a global issue all over the world and as the people start aging it is a common problem of getting diseases such as cholestrol, diabetes, blood pressure, etc, . It is not possible to assist elderly people with our presence with them to take medicine at correct time, some elderly people have a problem in taking medicine at correct time and get confused to take which medicine due to forgetness, this leads to severe health issues for elderly people. To avoid this Existing problem new method is proposed to help elderly people to get voice messages to take which type of medicine at what time as an alert.

### 2.3. Problem Statement Definition

Personal Assistance for elderly people for alerting them to take medicine on time is designed for helping the seniors who are self - reliant. The alerting mechanism also includes the medicine details for specific users so that it will be easy for them to find their medicine and take it at right time.


### 3. IDEATION & PROPOSED SOLUTION

#### 3.1. Empathy Map



#### 3.2. Ideation & Brainstorming

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



## Brainstorm & idea prioritization

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

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

➔

**Before you collaborate**

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

🕒 10 minutes

---

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

[Open article](#) ➔

1

**Define your problem statement**


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

🕒 5 minutes

---

PROBLEM

How might we [your problem statement]?



**Key rules of brainstorming**

To run a smooth and productive session

- 😊 Stay in topic.
- 💡 Encourage wild ideas.
- ⏸️ Defer judgment.
- 👂 Listen to others.
- 🗣️ Go for volume.
- 👁️ If possible, be visual.

## Step-2: Brainstorm, Idea Listing and Grouping

2

**Brainstorm**

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

🕒 10 minutes

**TIP**

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

Amar

Yuktesh

Person 3

Person 4

Person 5

Person 6

Person 7

Person 8

3

**Group ideas**

Take turns sharing your ideas while clustering similar or related notes as you go. In the last 10 minutes, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you can break it up into smaller sub-groups.

🕒 20 minutes

Person 4

**TIP**

Add customizable tags to sticky notes to make it easier to find, browse, organize, and categorize important ideas as themes within your mural.

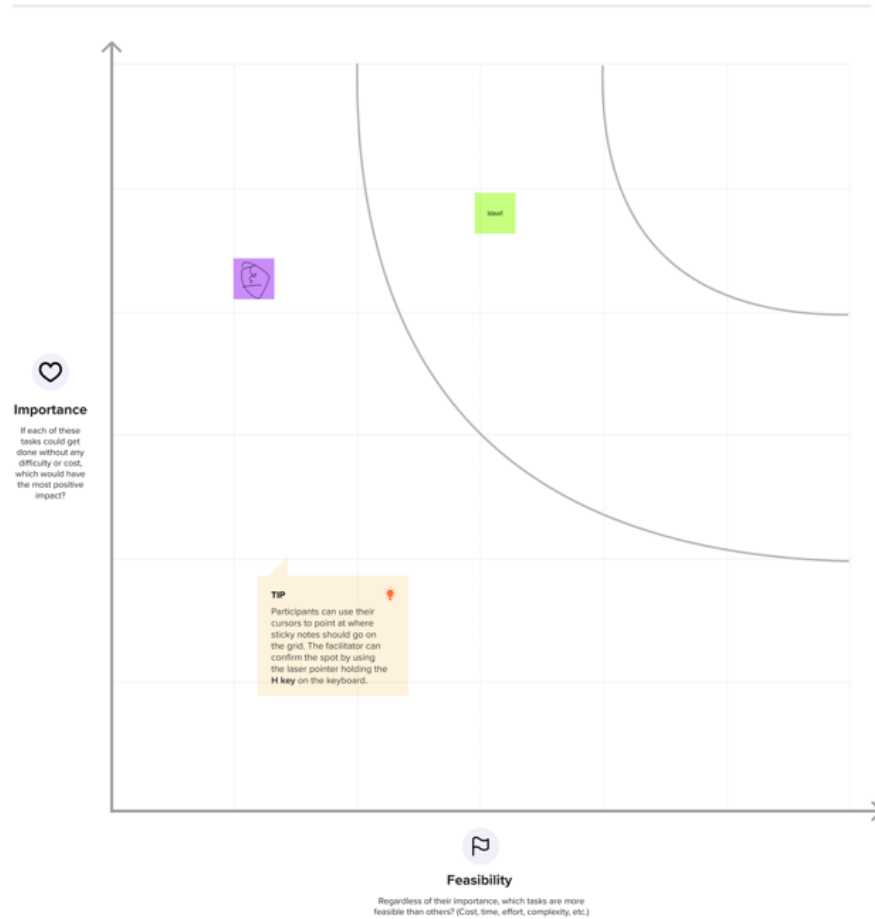
## Step-3: Idea Prioritization

4

**Prioritize**

Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which ideas are important and which are feasible.

🕒 20 minutes



### 3.3. Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Sometimes elderly people forget to take their medicine at the correct time. They also forget which medicine He / She should take at that particular time. And it is difficult for doctors/caretakers to monitor the patients around the clock. To avoid this problem, this medicine reminder system is developed. An app is built for the user (caretaker) which enables him to set the desired time and medicine.

2.	Idea / Solution description	we introduce a smart medicine reminder system based on IoT. The proposed scheme was particularly created for the Android platform. For our system, we implement a reminder system which provides an alarm when it is time for taking medicine. Along with that, there is an android application where the user can set their medicine time. In the application, there will some feature that help the user to know more details about their medicine. It keeps track for the medicine which means how much medicine they have to take they can be fixed in the application
3.	Novelty / Uniqueness	It is a user-friendly app that sends users medication and refill reminders, provides drug interaction warnings, and helps caregivers manage prescriptions for loved ones
4.	Social Impact / Customer Satisfaction	I constructed these proto-personas, or names, based on the research findings from the user interview. They would be crucial to the rest of the design process. All design decisions may be assessed and re-evaluated using these personas, keeping the user and their perspective in mind.
5.	Business Model (Revenue Model)	When it comes to the business there is no one-size-fits- all solution. The model you choose depends on your target audience, business goals, and the resources you already possess.
6.	Scalability of the Solution	where the user can set their medicine time. In the application, there will some feature that help the user to know more details about their medicine. It keeps track for the medicine which means how much medicine they have to take they can be fixed in the application

### 3.4. Problem Solution fit

Define CS, fit into CC	<b>1. CUSTOMER SEGMENT(S)</b> <span>CS</span>  Elderly people who are self - Reliant	<b>6. CUSTOMER CONSTRAINTS</b> <span>CC</span>  Elderly people should be at a distance of the alerting system and they should be awake to hear the alerting voice message such that they could take their medicine according to the instructions	<b>5. AVAILABLE SOLUTIONS</b> <span>AS</span>  Elderly people usually take their medicine by themselves or with the care of family or doctors but it is not always good for them to depend on others	Explore AS, differentiate
	<b>2. JOBS-TO-BE-DONE / PROBLEMS</b> <span>J&amp;P</span> <ul style="list-style-type: none"><li>Alerts to take medicine on correct time</li><li>Which type of medicine should be taken will be specified in the alert message</li></ul>	<b>9. PROBLEM ROOT CAUSE</b> <span>RC</span>  Elderly people have the problem of irregular intake of medicine or intake of wrong medicine due to oblivion	<b>7. BEHAVIOUR</b> <span>BE</span>  Once the voice message is sent elderly people take their correct type of medicine at correct time	
Focus on J&P, tap into BE, understand RC	<b>3. TRIGGERS</b> <span>TR</span> We can request our customer to get an experience with our product and explain them the needs of our products	<b>10. YOUR SOLUTION</b> <span>SL</span> We can get the details of the medicine and accurate time to take medicine and store the data in database and connect the IoT device to database and convert the text message to voice messages that could be listened by the elderly people	<b>8.CHANNELS of BEHAVIOUR</b> <span>CH</span> <b>Online:</b> Details of medicine and time could be fetched  <b>Offline:</b> The data fetched and converted voice message could be delivered at high sound	Identify strong TR & EM
Identify strong TR & EM	<b>4. EMOTIONS: BEFORE / AFTER</b> <span>EM</span> Before: Customer takes medicine at improper time interval and also could have possibility of taking wrong medicine After: Customer takes correct medicine at correct time			

## 4. REQUIREMENT ANALYSIS

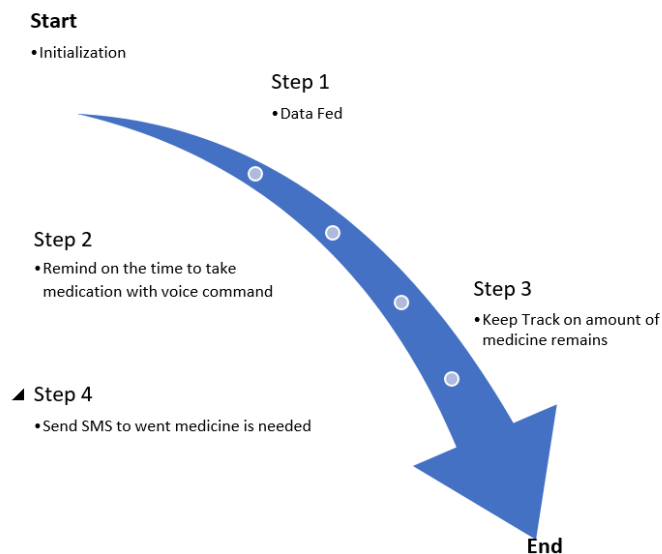
### 4.1. Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story/ Sub-Task)
FR-1	User Registration	Registration through Form Registration through Gmail Registration through LinkedIn
FR-2	User Confirmation	Confirmation via Email Confirmation via OTP
FR-3	User Login	login through User Id and Password.
FR-4	Network Connectivity	via Wi-Fi /mobile data.
FR-5	IBMIoT Platform	Access cloud storage via internet and it gives medication information.
FR-6	Node-RED	Uses to transfer the data from IOT platform to UI platform and helps in storing the data.

## 4.2. Non-Functional requirements

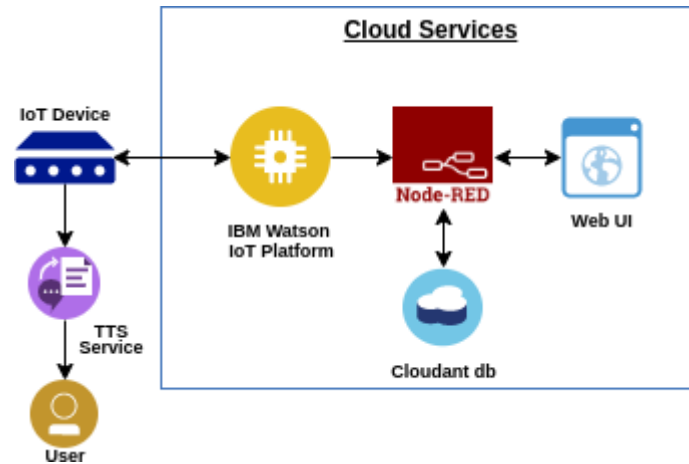
FR No.	Non-Functional Requirement	Description
NFR-1	<b>Usability</b>	It can easily track and monitor the medication time of users and share the information to the caregivers.
NFR-2	<b>Security</b>	The cloudant database is highly secured and it prevents data from hacking.
NFR-3	<b>Reliability</b>	The prescription of medication for users is assured all the time.
NFR-4	<b>Performance</b>	It reminds users to take their medications and get them refilled, warns about drug interactions, and assists caregivers in managing prescriptions.
NFR-5	<b>Availability</b>	To keep track the medication of users.
NFR-6	<b>Scalability</b>	The users can set the time for their medication and also can adjust how much medication to take within the application.

## 5. PROJECT DESIGN



### 5.1. Data Flow Diagrams

## 5.2. Solution & Technical Architecture



**Table-1: Components & Technologies:**

S. No	Component	Description	Technology
1.	User Interface	Mobile App	MIT
2.	Application Logic-1	Gathering medicine details and check the timings	Python
3.	Application Logic-2	alarms	IBM Watson
4.	Cloud Database	Database Service on Cloud	IBM Cloudant DB

**Table-2: Application Characteristics:**

S. No	Characteristics	Description	Technology
1.	Open-Source Frameworks	Mobile application development set up, programming the IoT device, text to speech service, storing details in cloud	MIT, IBM Watson, node red, IBM cloud
2.	Security Implementations	Implementing encryption for security purpose	SHA-256.
3.	Scalable Architecture	Application is able to load as many members as logged in	MIT, IBM Watson
4.	Availability	Application is available 24/7	MIT, IBM Watson, node red, IBM cloud
5.	Performance	Reminder with correct timing	IBM Watson, IBM IoT Platform



### 5.3. User Stories

User Type	Functional requirement	User story number	User story/task	Acceptance criteria	Priority	Release
Customer (Mobile user, Web user, Care executive, Administrator)	Registration	USN-1	As a user, I can register for the application by entering my mail, password, and confirming my password	I can access my account/ dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
	Dashboard	USN-3	As a user, I can register for the application through internet	I can register & access the dashboard with Internet login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail	I can confirm the registration in Gmail	Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password	I can login with my id and password	High	Sprint-1

## 6. PROJECT PLANNING & SCHEDULING

Sprint	Functional Requirement (Epic)	User Story / Task	Story Points	Priority
Sprint-1	Resources Initialization	Create and initialize accounts in various public APIs like OpenWeatherMap API.	1	LOW
Sprint-1	Local Server/Software Run	Write a Python program that outputs results given the inputs like weather and location.	1	MEDIUM
Sprint-2	Push the server/software to cloud	Push the code from Sprint 1 to cloud so it can be accessed from anywhere	2	MEDIUM
Sprint-3	Hardware initialization	Integrate the hardware to be able to access the cloud functions and provide inputs to the same.	2	HIGH
Sprint-4	UI/UX Optimization & Debugging	Optimize all the shortcomings and provide better user experience.	2	LOW

## 7. CODING & SOLUTIONING

### 7.1. Feature 1

- IoT Device
- IBM Watson platform
- Node - Red
- Cloudant DB
- Web UI
- MIT App Inventor
- Python Code

### 7.2. Feature 2

- Login
- Wokwi

## 8. TESTING

### 8.1. Test Cases

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

Section	Total Cases	Not Tested	Fail	Pass
Login Page	5	0	0	5
Node Red Dashboard	32	0	0	32
IBM Watson IOT platform	2	0	0	2
MIT App Inventor	3	0	0	3

### Test Case 1:

## 8.2. User Acceptance Testing

The main Purpose of UAT is to validate end to end business flow. It does not focus on cosmetic errors, spelling mistakes or system testing. User Acceptance Testing is carried out in a separate testing environment with production-like data setup. It is kind of black box testing where two or more end-users will be involved.

UAT is performed by:

- Client
- End user



## 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	4	3	2	1	10
Duplicate	1	0	3	0	4
External	2	2	1	1	6
Fixed	4	3	5	19	31
Not Reproduced	1	0	1	1	3
Skipped	0	0	1	1	2

## 9. RESULTS

### 9.1. Performance Metrics

An experiment is conducted on an elderly person who is in need of Personal Assistant Device and the following results are obtained, it shows the medicine reminder that gives the information regarding the intake of medicine by the person using the personal Assistant Device. The stored data in cloudant database on specified time alerts user with a voice message.



One of the greatest hindrances of a PDA is the expense. Other than paying for the gadget itself, most PDAs require the purchaser to buy in to a utilization contract. This includes a month to month bill and the chance of overage charges if the client outperforms his designated free telephone minutes or information limits.

- Interruption

PDAs may likewise turn into an interruption when they're not satisfying an authentic need. The capacity to be constantly associated can prompt sat around riding the Web, settling on telephone decisions or messing around. Some business clients whine of being "available to come in to work" when their colleagues and bosses can reach them whenever.

- Restricted in Scope

PDAs are restricted in degree. They are neither PC substitutions nor would they be able to be successfully used to supplant mobile phones. PDAs are not furnished to manage miniature preparing capacities.

- Time constraint

PDAs are not generally the best response to business arrangements. Paper-based coordinators are a more reasonable choice since PDAs are hard to utilize, information passage is abnormal, they are moderate and beginner clients discover them superfluously unpredictable.

## **11. CONCLUSION**

With the continuously increasing utilization of internet in this point in time, this assignment paintings have been engaged to execute a framework depending on web innovation which could discuss through internet for health checking of patients and for giving assist to vintage people. It utilized to apprehend the development of patient which sends these statistics to everything communicate producer to reveal the readings. During the crisis situations, a caution might be raised over the internet level telling the expert/overseer by way of the patient simply by squeezing a seize in the helpful machine. This offers a trustworthy framework which can screen the well-being reputation continuously of a patient or a vintage individual.

## **12. FUTURE SCOPE**

Whether or not the role of IoT as the best solution to provide help for the weak elderly citizens is accepted, yet these people are certainly in need of care. There are some strong and determined persons who manage to preserve their mind and body active until an old age. Still there are many who are in need of aid in their routine life as well as those people who totally depend on others. It provides an effective homecare monitoring and care support for elderly people by communication and coordination with professional helpers and thereby improving the quality for independent life of old aged. Future elder care IoT projects will also more than likely have the ability to take on medical diagnostics, as well as use facial recognition algorithms to determine how someone is feeling. But despite all of this future capability, there still exists a dichotomy of things that IoT can do way better than humans and things they simply cannot do at all. For instance, an elder care IoT based projects in the future may easily be able to find and retrieve a pill box from another room, however, without an excellent mobility system, it will be stopped dead in its tracks

should it get caught on something along the way. Collaboration and integration between researchers, private industry, investors, and the government will be key in the years to come.

## 13. APPENDIX

### 13.1. Source Code

#### 1. Python code for random medicine and time generating:

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

myConf
    ig = {
        "ident
ity": {
            "orgId": "dhhnmy",
            "typeId": "naveen",
            "deviceId": "ibm"
        },
        "auth": {
            "token": "Nanee_Depp_3112"
        }
    }
client = wiotp.sdk.device.DeviceClient(config=myConfig,
logHandlers=None)client.connect()

for i in range(0,20):
    tablet=["Paracetamol","Aspirine","Azithral","Asthalin","Sinarest"]
    medicinetime=[12.00,1.00,2.00,3.00,5.00,18.00,20.00,7.00]
    name = "mani"
    medicine=random.choice(tablet)
    medicinetime=random.choice(medici
netime)
    mydata = {'Patient Name': name, 'Medicine Name': medicine, 'Time':
    medicinetime}client.publishEvent("IoTSensor", "json", data=mydata,
    qos=0, onPublish=None) print("Data published to IBM IOT platform :",
    mydata)
    time.slee
p(5)
client.disco
```

nnect()

## WOKWI SIMULATED CODE

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQTT
#include <LiquidCrystal_I2C.h>
#include "DHT.h"// Library for dht11
#define DHTPIN 15 // what pin we're connected to
#define DHTTYPE DHT11 // define type of sensor DHT 11
#define LED 2
DHT (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht
connected
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-----credentials of IBM Accounts-----

#define ORG "64yf7x"//IBM ORGANITION ID
#define DEVICE_TYPE "b11m3edevicetype"//Device type mentioned in ibm watson IOT
Platform
#define DEVICE_ID "b11m3edeviceid"//Device ID mentioned in ibm watson IOT
Platform
#define TOKEN "-&EMtr7l-v-Gz2G))e" //Token
String data3="";
int buzz= 13;

//----- Customise the above values -----
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event
perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd REPRESENT command
type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
LiquidCrystal_I2C lcd(0x27,32,2);

//-----
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the
predefined client id by passing parameter like server id,portand wificredential

void setup()// configureing the ESP32
{

  Serial.begin(115200);
  dht.begin();
  pinMode(buzz, OUTPUT);
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
```



```

}

void loop()// Recursive Function
{
    if (!client.loop()) {
        mqttconnect();
    }
}

/*.....retrieving to
Cloud.....*/

void PublishData(float temp, float humid) {
    mqttconnect();//function call for connecting to ibm
}

void mqttconnect() {
    if (!client.connected()) {
        Serial.print("Reconnecting client to ");
        Serial.println(server);
        while (!!!client.connect(clientId, authMethod, token)) {
            Serial.print(".");
            delay(500);
        }

        initManagedDevice();
        Serial.println();
    }
}

void wificonnect() //function defination for wificonnect
{
    Serial.println();
    Serial.print("Connecting to ");

    WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish
the connection
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("");
    Serial.println("WiFi connected");
    Serial.println("IP address: ");
    Serial.println(WiFi.localIP());
}

void initManagedDevice() {
    if (client.subscribe(subscribetopic)) {
        Serial.println((subscribetopic));
        Serial.println("subscribe to cmd OK");
    } else {

```

```

        Serial.println("subscribe to cmd FAILED");
    }
}

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{
    Serial.print("callback invoked for topic: ");
    Serial.println(subscribetopic);
    for (int i = 13; i < payloadLength-2; i++) {
        //Serial.print((char)payload[i]);
        data3 += (char)payload[i];
    }

    Serial.println("Medicine Name: "+ data3);
    if(data3 != "")
    {
        lcd.init();

        lcd.print(data3);
        digitalWrite(LED,HIGH);
        tone(buzz, 100, 1000);
        delay(2000);
        digitalWrite(LED,LOW);
        noTone(buzz);
        delay(1000);

    }

    else
    {
        digitalWrite(LED,LOW);

    }
    data3="";
}

```

<https://github.com/IBM-EPBL/IBM-Project-28152-1660107362>