NALAIYA THIRAN PROJECT

HX 8001 - Professional Readiness for innovation, Employability and Entrepreneurship

Project Name : Smart Waste Management Systems for Metropolitan Cities

Team ID : PNT2022TMID49521

Project Delivery Date: 19 November 2022

1. INTRODUCTION

Project overview:

In our project is used to detected the harmful waste material in the household wastage. Because, in this waste can able to create the diseases and infected the peoples. Hence, we are creating the harmful waste detector for detect the waste and send the information to the garbage collector to collect the waste immediately.

Purpose:

It is used to detect the harmful wastage immediately in the house. Hence, people are secure from the diseases and live healthier. It is mostly useful in now a day.

2. LITERATURE SURVEY

Abstract - Waste management system includes monitoring of garbage, collection, transport, sorting, recycling or disposal, and waste material monitoring. The system usually denotes to the materials created by human activity and is generally used to decrease their influence on health, the atmosphere or aesthetics. Whenever the garbage bins are full, it also appears to overwhelm individuals and it allows waste to spill from dustbins. So, one of the critical issues that is becoming difficult to lead hygienic life is an efficient waste management system. A smart garbage monitoring framework based on IOT that uses web application PHP programming, Kodular, Node-MCU, Ultrasonic sensors and GPS for communication is proposed to ensure this problem. In the proposed method, the ultrasonic sensor tracks and tests the amount of garbage in the garbage bins, and if a person wants to throw the garbage, an android app is programmed to direct the user to the bin that is not complete using GPS and also sends a warning message to the municipal department authorities concerned about bin status using cloud (Firebase), Node-MCU. The proposed system is more effective than the current system as it simplifies the human effort, saves time and is more cost effective.

I. INTRODUCTION

In today's environment, waste management is very important concept. Due to the population rise, the production of waste is become double day by day. In turn, the rise in waste is impacting many people's lives. People who live in these areas appear to suffer a great deal from the detrimental effects of poor sanitation. Waste management is one of the biggest problems facing many countries in the world, with 2.01 billion tons of waste produced annually worldwide and rising to 3.40 billion tons by 2050 [1]. The statistical graph is shown in Fig. 1. If this waste is not properly disposed of, the ecosystem and, in turn, the human lifestyle would be severely affected. So, proper waste monitoring and collection needs to be undertaken. Most of the natural resources are being polluted by the waste disposal openly into the soil and freshwater resources which needs

to be take care. Now a Days urban garbage monitoring is being problematic. Though they are providing bins in areas and doing activities like Swachh Bharath, there are many cases where garbage cans are being over filled and people tending to dump their garbage over the bins which causes them to spill over by the air on to the roads. The main problem is that the trucks that are ordered to empty those bins need to visit each and every bin taking long journeys and huge time loss for them. In today's world most of the devices used by the people are connected to each other by using the concept of Internet of Things due to tremendous growth in the internet. Things in IOT refer to a hardware device which has sensing ability and will communicate with other devices or base stations via IP address [2]. Majority of the devices are linked with the internet servers and operate simultaneously even if they are far away from each other. The concept of IoT is used to design the garbage monitoring system for efficient collection of waste in urban areas. The proposed system using NodeMCU, Ultrasonic Sensor and Kodular app is made a realistic monitoring and cleaning the garbage bins when the level crosses the threshold in waste management system. Using the software installed on the phone, the user can track the amount of the garbage at any time and when the garbage in a specific bin reaches the threshold, then the google map automatically shows the location of the bin whose garbage is not complete and at the same time an SMS will be sent to the authority concerned. The proposed framework can be used by individuals who are willing to take one step forward to improve cleanliness in their valued areas as a benchmark. The paper is organized as chapter 2 discuss the Review of Literature, chapter 3 explains Implementation and Design, Results and Discussion is done in chapter 4 and Conclusion & Future enhancement in chapter 5.

II. LITERATURE REVIEW

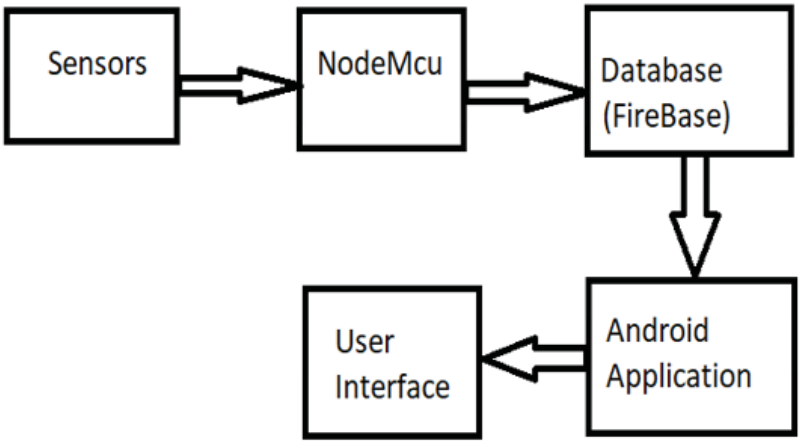
Waste management system is one of the major problem faced by the world. It should be solved efficiently to save the natural resources and environment in turn the lives of the human being. Various authors [3] has come up with different solutions but none of them is an efficient one. Waste management system can be remodified and efficiently utilized by adopting innovations happening in the latest technologies. The existing research which solves the problem to some extent has been surveyed. Minhaz Uddin Sohag and Amit Kumer Podder [2] designed an IOT based system which identifies the status of garbage bin and intimate the level to the user and also to the concerned authority with the help of Arduino UNO and Ultrasonic sensor, but this design will solve the problem for small-scale garbage bin only. Authors in [4] has designed garbage system for urban cities using Android application through which user and authorities can know the status of bins. Authors in [5] proposed a system which differentiates biodegradable and non-biodegradable waste by using ultrasonic sensor to measure the level of bin and MQ4 sensor to measure the odour level and the measured information will be sent to authorities using NodeMCU web server. Author also proposes SOS operation in app, so that if IOT system fails then an individual can send the message to the authorities using app for cleaning the bins. Shashank Mithinti, Aman Kumar, Suryansh Agarwal and Isha Malhotra [6] proposes a selection algorithm which is based on level of percentage, date and time, based on this shortest path algorithm calculates the path and selected bin to be disposed. Authors uses various technologies like Arduino [7, 8], Zigbee [10], wifi-enabled microcontrollers [13, 21], Wemos D1 mini [14], AVR microcontroller [15], NI myRIO [16], Raspberry Pi [19], VANET [22], NodeMCU [24, 25], different sensors like Ultrasonic sensor, IR sensor, Rain sensor, gas and smoke detection sensor, carbon monoxide and methane detection sensor, flame sensor, intimating through buzzer and

GSM/GPS and different languages like HTML and Embedded C. In [9], an Intelligent waste container is designed by using microcontroller, blynk IOT android, wifi module. In this ultrasonic sensor will detect the level of garbage in bins and once the bin is full then using wireless sensor nodes real time information will be transmitted to waste container, which then collects the waste from bins. In [11], author proposes neural network concept to classify the waste into digestible and indigestible and smart disposal boxes to monitor the real time data using IOT. Raspberry pi is used to classify the waste and sends an information to servo motor, so that corresponding waste will be dumped into respective disposal box. In [12], author proposes a smart dustbin which is used to separate waste from household and reuse that waste to make bio compost using machine learning technique called KNN and IOT technology. In this paper based on the three parameters called level of biodegradable, non-biodegradable and poisonous gas concentration, a message will be sent to authorities. In [17], author developed a sensor node which will monitor the garbage bin level and routes the information related to orientation sensor, distance/height sensor and real time clock module to central database using Arduino Nano microcontroller, then collected data from sensors will be displayed on webpage so that concerned authorities can make a route plan to collect the garbage. Umar Draz, Tariq Ali, Jamshaid Ali khan, Muhammad Majid and Sana Yasin [18] proposed monitoring the dumpsters and when it is full, collecting the garbage from respective dumpster. Authors used IOT technology to calculate the status of garbage based on threshold and its location with the help of GPS, so that concerned authorities can take action to clean the dumpster. To achieve this, components used are Arduino UNO, GSM module, LCD, IR Flame sensor, Ultrasonic sensor, GPS and Opt couplers. Authors [20], proposed smart bins forming a network and communicate through cloud-based techniques which is used to monitor, and analysis of data collected from smart bins to decide the shortest route for garbage trucks to collect the garbage from concerned bin. Authors [23], has given a solution for waste management system by integrating Wireless sensor network concept with IOT and also to improve garbage monitoring efficiency, machine learning technique called decision forest regression method is applied. Research work of various authors is compared with the proposed system is shown.

III. IMPLEMENTATION AND DESIGN

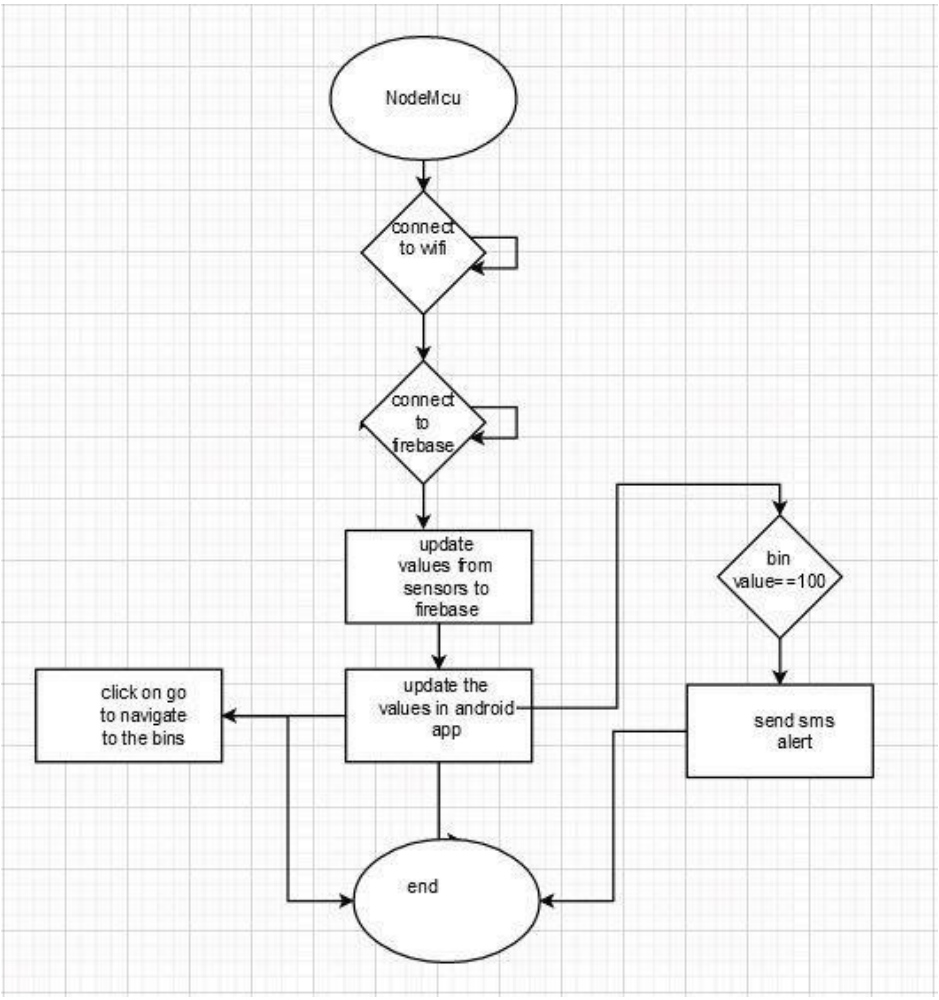
A. Block Diagram

The collection of information in the proposed system is done by ultrasonic sensors where the information is being fed to the Node-MCU and updated into the firebase database using the Wi-Fi-connectivity provided by ESP8266 SoC and at the same time the information is being updated to the user and the authorities using android application that is being made from using Kodular (a modern app making without heavy coding) which also sends the SMS alerts to the authorities without usage of separate GSM modules as shown in Fig.



B. Flow Chart

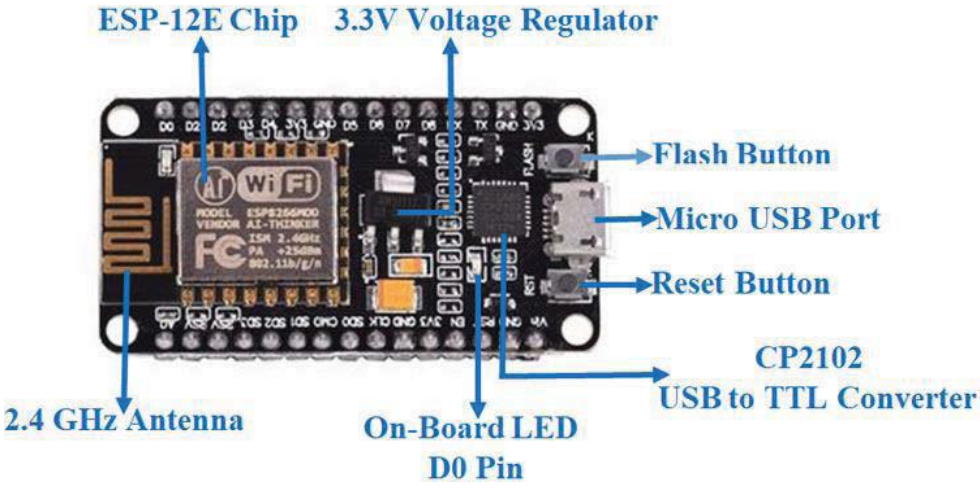
Fig. shows the proposed system flow of execution.



C. Hardware Implementation

Node-MCU

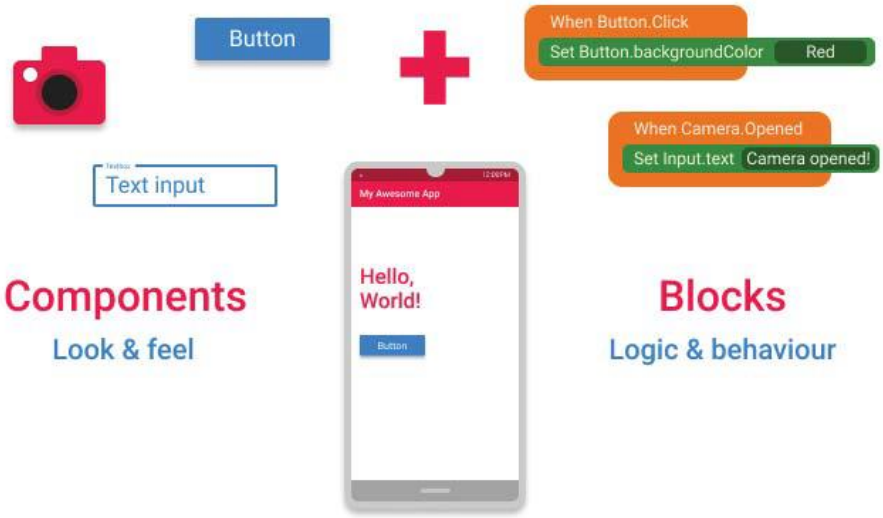
The Node-MCU (Node Micro Controller Unit) is an opensource environment for the development of software and hardware designed around a cheap System-on-a-Chip (SoC) called ESP8266. The ESP8266, developed and manufactured by Espressif systems, as shown in Fig. 4, comprises the key components of a computer: CPU, RAM, WiFi, and even a modern operating system and SDK, making it an outstanding option for all sorts of Internet of Things (IOT) applications.



D. Software Implementation

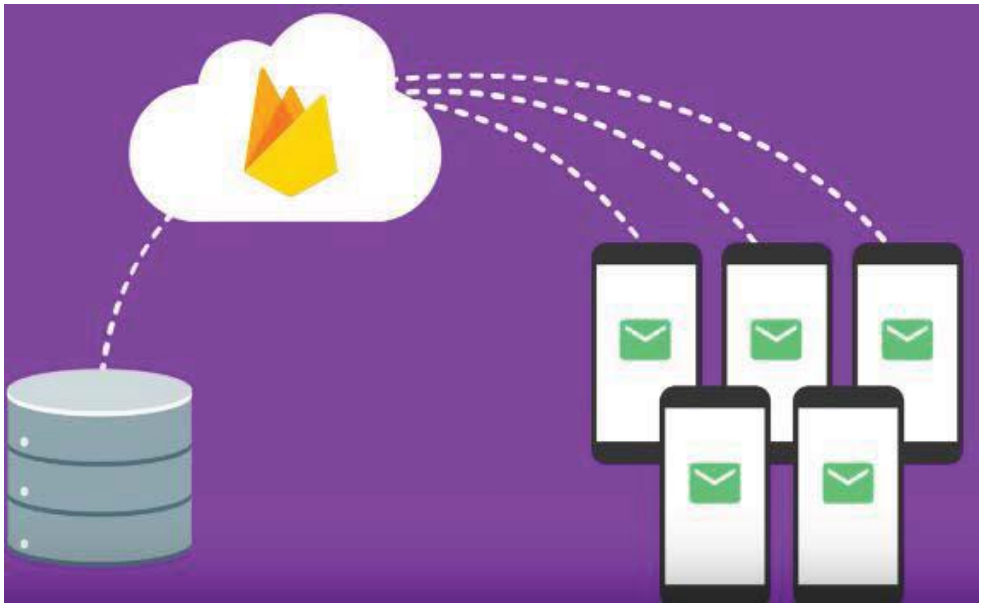
Kodular

With each individual component being used for a particular purpose, the apps in Kodular are constructed as a combination of different components. Blocks are used to configure the component's actions, as shown in Fig. 6. In Kodular, components are the basic building elements. With the assistance of one component or the other, everything in the app is finished. For various purposes, various components are used. One feature, such as a button component, can be used to design the app's user interface (UI), while others can be used to perform actions such as interfacing with a database, saving an image to the Android device's folder etc. for example the *FirebaseDB* component. Blocks are those that are used to explain how a job can be completed. Blocks are often constructed in the same way as how the components respond to different actions and events in the app. Foe example, when a button is pressed, how the app should react, what information to communicate to the database using the FirebaseDB part, etc. are all configured using the block. Thus, an app is "koded" in Kodular by using the various components and by configuring its behavior and response to user actions with the aid of blocks.



Firebase

Firebase as shown in Fig. is a google's mobile platform that provides the developers with various tools and services for development of high-quality apps. Firebase database is a cloud hosted sql database that lets the users to store and upload data dynamically.



E. Algorithm

_ Start.
_ Include required libraries and define pins.
_ Define the pinmodes of the pins and check for the wifi connectivity.
_ Check the level of garbage in the bins with Ultrasonic sensors.

_ Send the data to the firebase real time database.
_ If level of the garbage updated in bins, update the information in Database.
_ If the level of the bin is 100%, send the alert SMS to the concerned authority.
_ If the user clicks on “GO” in application, navigate to the selected bin through gmaps.
_ Stop.

F. Internet of Things (IOT)

The Internet of Things (IOT) is a broad network of related gadgets that gather and store information based on how they are used and the world in which they function. For all devices, IOT offers a common forum. IOT is an advanced technology that uses the internet to monitor or track internet-related computers, cars and other physical objects. IOT research is never ending and will be beneficial to humanity with the great possibility of linking more and more gadgets to the cyber world. The proposed system uses NodeMCU which links to the internet, where the data will be collected from the sensors and send them to the real-time Firebase database to access the information anywhere the internet is accessible. This enables the user to use the internet to track the smart garbage monitoring system.

G. Firebase Realtime Database

The Realtime Database Firebase is a cloud-hosted database. In the firebase database, information is stored in the form of a JSON (JavaScript Object Notation). For data saving and sending, JSON is a lightweight format. The real-time firebase database allows us to create rich, collaborative applications by allowing safe access to the database directly from the developed user-side code. In our project we have choose firebase because of its vast facilities to create real time applications. In our project we are sending information collected from sensors through Node-MCU to the firebase real time database.

H. App Development

Kodular is used for developing the android application for our project. Kodular uses blocks and components to create apps. Kodular allows anyone to create perfect android apps easily without writing any line of code. Kodular allows us to concentrate on ideas for creating apps. In our project the userside application is developed by Kodular where it fetches the data from the firebase real time database and displays it to the end-users it also provides the google maps extension for navigating to the garbage bins.

I. Pseudo Code

1: Program starts
2: Initialize Variables
3: Enter firebase authorization
4: Enter Wifi authorization
5: While Wifi status = = not connected
6: Continue
7: End while

8: Loop
9: If bin level = = 100
10: Send SMS to authorities
11: End if
12: Else
13: Update bin level in firebase
14: End else
15: If “GO” button clicked
16: Open gmaps to navigate
17: End if

18: Else

19: Continue

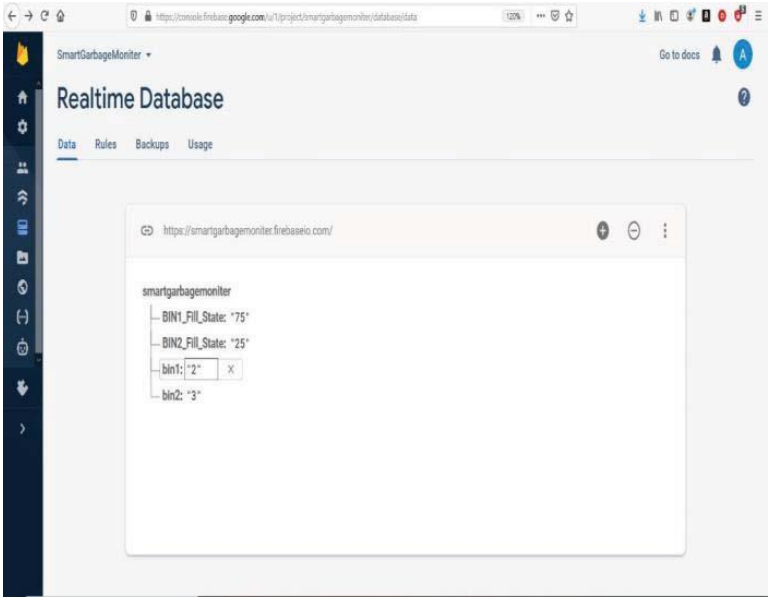
20: Go to loop

21: Program end.

IV. RESULTS AND DISCUSSION

The working model for the proposed system is designed and implemented. The performance of the system was analyzed for different conditions. All the sensors are checked in such a way that their values are above and below the threshold values. All the hardware components ought to be connected with proper connections and ensure that wifi module is connected to internet as the values will update into “firebase database in real time. The output on the monitor is observed as shown in Fig. After ensuring that the android phone is connected to the internet, the application will open the first screen as shown in Fig., then a second screen will appear as shown in Fig. which shows the values of level of a garbage of different bins. Whenever the user clicks on “GO” button on one of the bins in the application then it will navigate to the maps screen, where the location of bins will be displayed and when user clicks the direction then the app will navigate in google map screen as shown in Fig. When the level of the garbage reaches the threshold value then an SMS alert will be sent to the authorities as shown in Fig. The main advantages of the proposed system over existing system are:

_ Location of the bins whose level of garbage reaches threshold can be easily communicated to authorities.
_ For sending SMS, instead of GSM module a fire base is used which leads to low-cost maintenance.
_ Data will be periodically processed in the cloud so that data can be used for future predictions.
_ Cost and effort will be less which makes environment clean and green.



V. CONCLUSION AND FUTURE ENHANCEMENT

The main aim of the proposed scheme is to preserve the standard of cleanliness in the city and create a healthier living atmosphere. We can continuously check the amount of the garbage in the dustbins that are located in different parts of the city by using this device. If the maximum level has been reached by a specific dustbin, the the employees can be informed and they can take certain actions immediately to empty it as soon as possible. Employees can verify the status of these bins on their cell phones at any time. When used properly, this can prove to be a very useful system. The system can be used by people who are willing to take a step further to increase cleanliness in their respected regions as a benchmark. Further the proposed system can be extended by using infrared sensor and weight sensor which are used for precisely monitoring the garbage in bins. It can also be extended to classify the dry waste or a wet waste by using machine learning techniques.

References:

[1] Ganesh, R. S., Mahaboob, M., Janarthanan, A. N., Lakshman, C., Poonthamilan, S., & Kumar, K. K. (2021, December). SMART system for hazardous gases detection and alert system using internet of things. In *2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA)* (pp. 511-515). IEEE.

[2] Wang, C., Qin, J., Qu, C., Ran, X., Liu, C., & Chen, B. (2021). A smart municipal waste management system based on deep-learning and Internet of Things. *Waste Management*, *135*, 20-29.

[3] Shyam, G. K., Manvi, S. S., & Bharti, P. (2017, February). Smart waste management using Internet-of-Things (IoT). In *2017 2nd international conference on computing and communications technologies (ICCCT)* (pp. 199-203). IEEE.

[4] Mahajan, S. A., Kokane, A., Shewale, A., Shinde, M., & Ingale, S. (2017). Smart waste management system using IoT. *International Journal of Advanced Engineering Research and Science*, *4*(4), 237122.

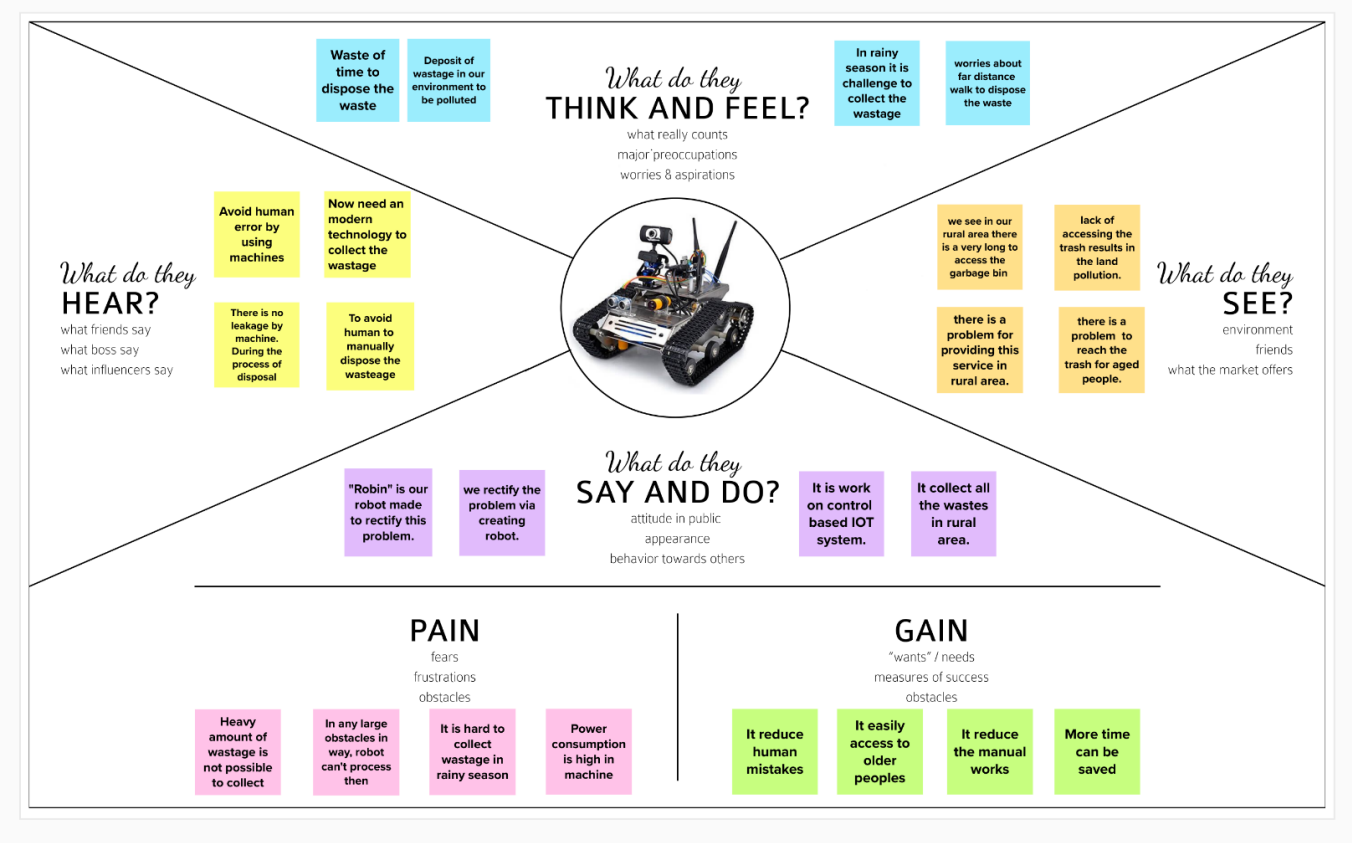
[5] Ali, T., Irfan, M., Alwadie, A. S., & Glowacz, A. (2020). IoT-based smart waste bin monitoring and municipal solid waste management system for smart cities. *Arabian Journal for Science and Engineering*, *45*(12), 10185-10198.

Problem Statement Definition

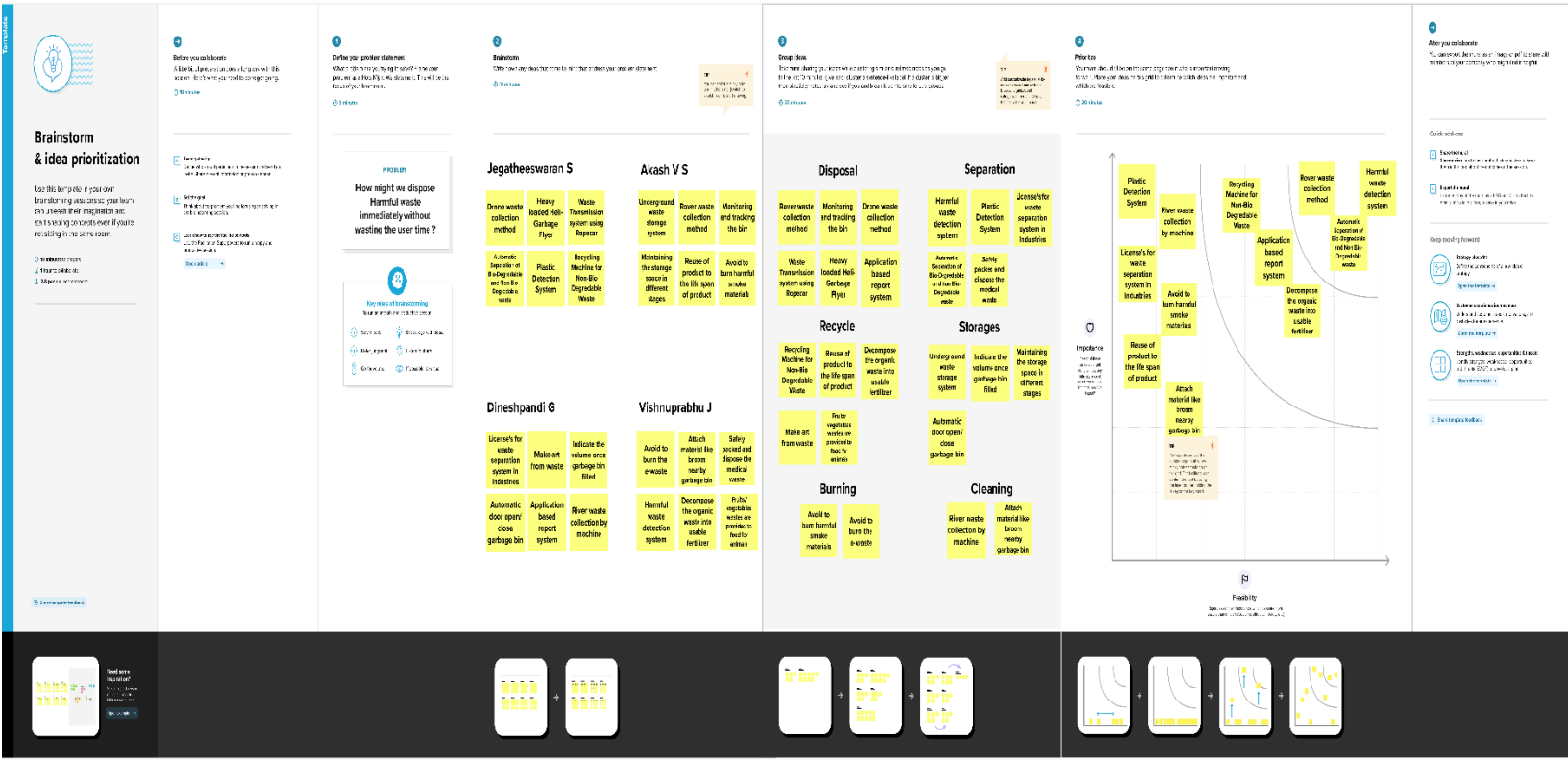
If harmful waste is detected in our home. It will able to create the disease for affecting people. Hence, we ae creating harmful waste detector to secure the people from the disease. This is our main aim of this project.

3. IDEATION & PROPOSED SOLUTION

Empathy Map Canvas



Ideation & Brainstorming

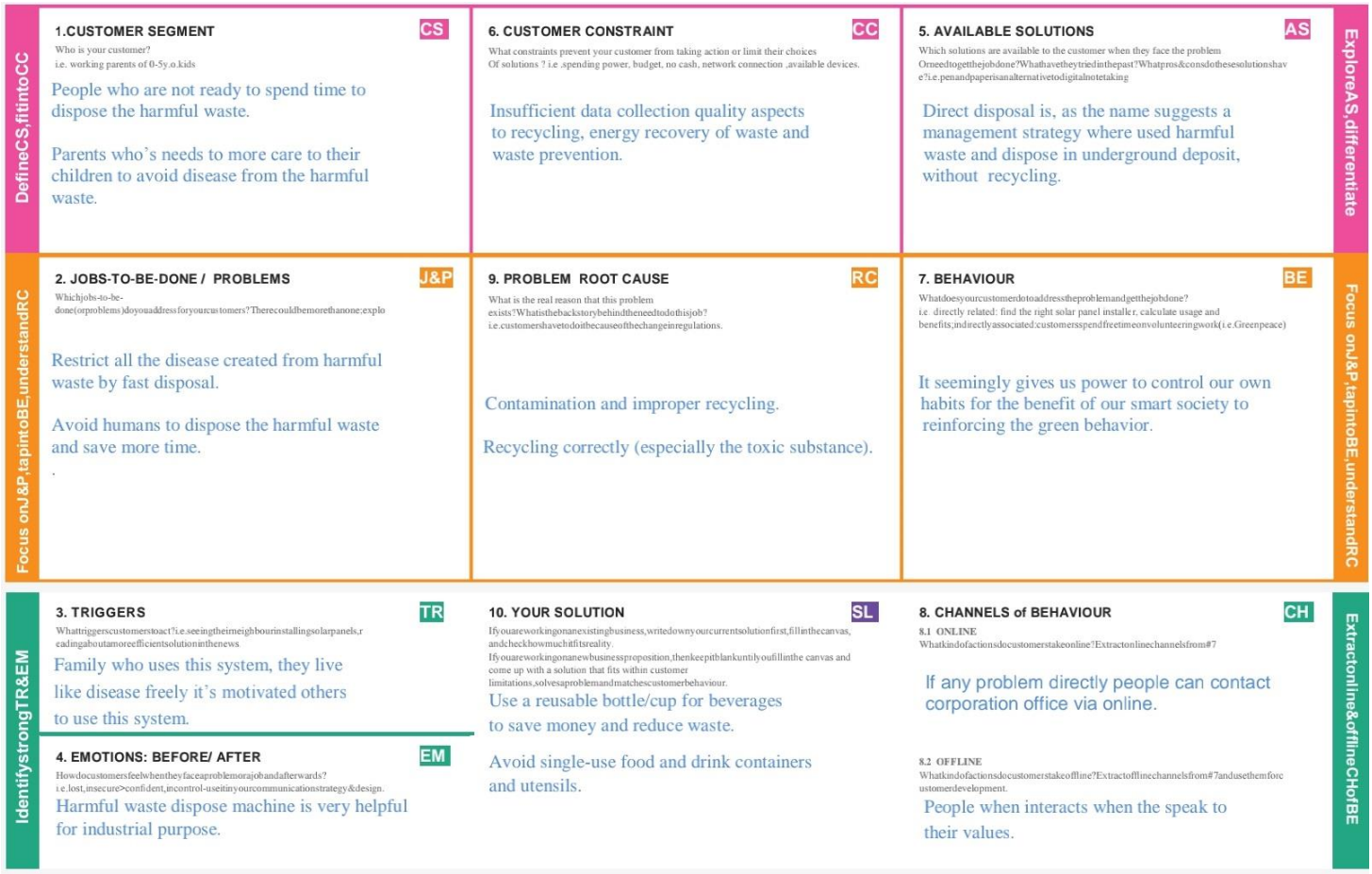


Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Harmful waste can cause disease. These wastes should be monitored regularly and disposed immediately.
2.	Idea / Solution description	By using the gas sensor for detecting harmful waste and immediately send the message.
3.	Novelty / Uniqueness	It sends message through WhatsApp and email via Internet to the appropriate disposer or corporation office.
4.	Social Impact / Customer Satisfaction	It controls the disease created from the harmful waste.
5.	Business Model (Revenue Model)	In this project is also used to detect and dispose the medical waste. By dispose of medical waste can charge for the dispose from the hospital.

6.	Scalability of the Solution	Usually, this type of services is only available for general waste bin. But we provide to every household and companies also. This led to control disease more effectively.

Problem Solution fit



4. REQUIREMENT ANALYSIS

Functional requirement

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Cloud Connectivity	Data will be stored in cloud database and used for route detection.

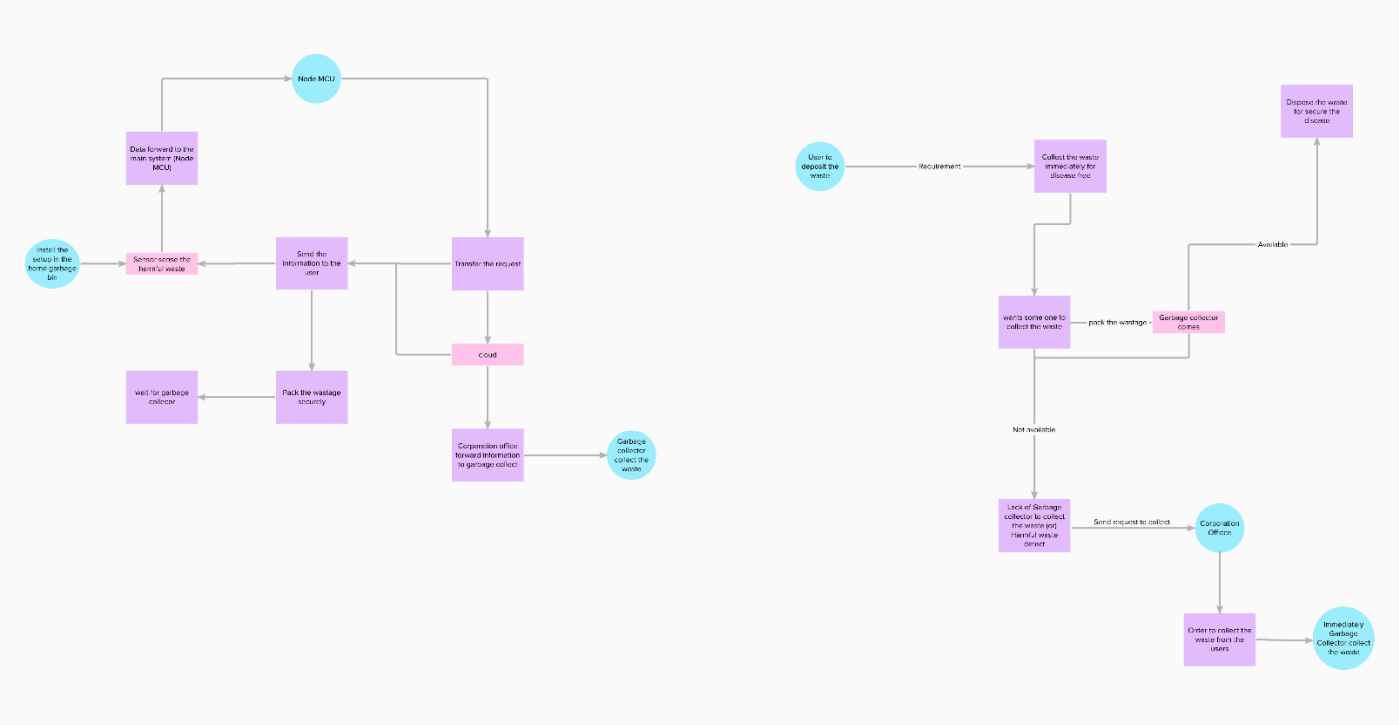
FR-2	Mobile Responsiveness	Connect the user mobile with the respective application
FR-3	Segregation Process	To segregate a garbage collection metal waste, E-waste, medical waste and harmful waste.
FR-4	Wireless Sensor	Optimize the waste management and reduce the manual work for dispose the harmful waste.
FR-5	Level Monitoring	Garbage level monitored in day by day sends data to nearby garbage collector trucks.

Non-Functional requirements

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	Reduction of man power and manage a collection routes to placed more bins effectively.
NFR-2	**Security**	It secures the user from the diseases formed by the harmful waste.
NFR-3	**Reliability**	Environmental protection from pollution and contamination of effective waste disposal.
NFR-4	**Performance**	It detects the harmful waste even it is present in small level also.
NFR-5	**Availability**	Garbage bin available in nearby building live able cities and residential and industrial sectors.
NFR-6	**Scalability**	It improves disease free environment.

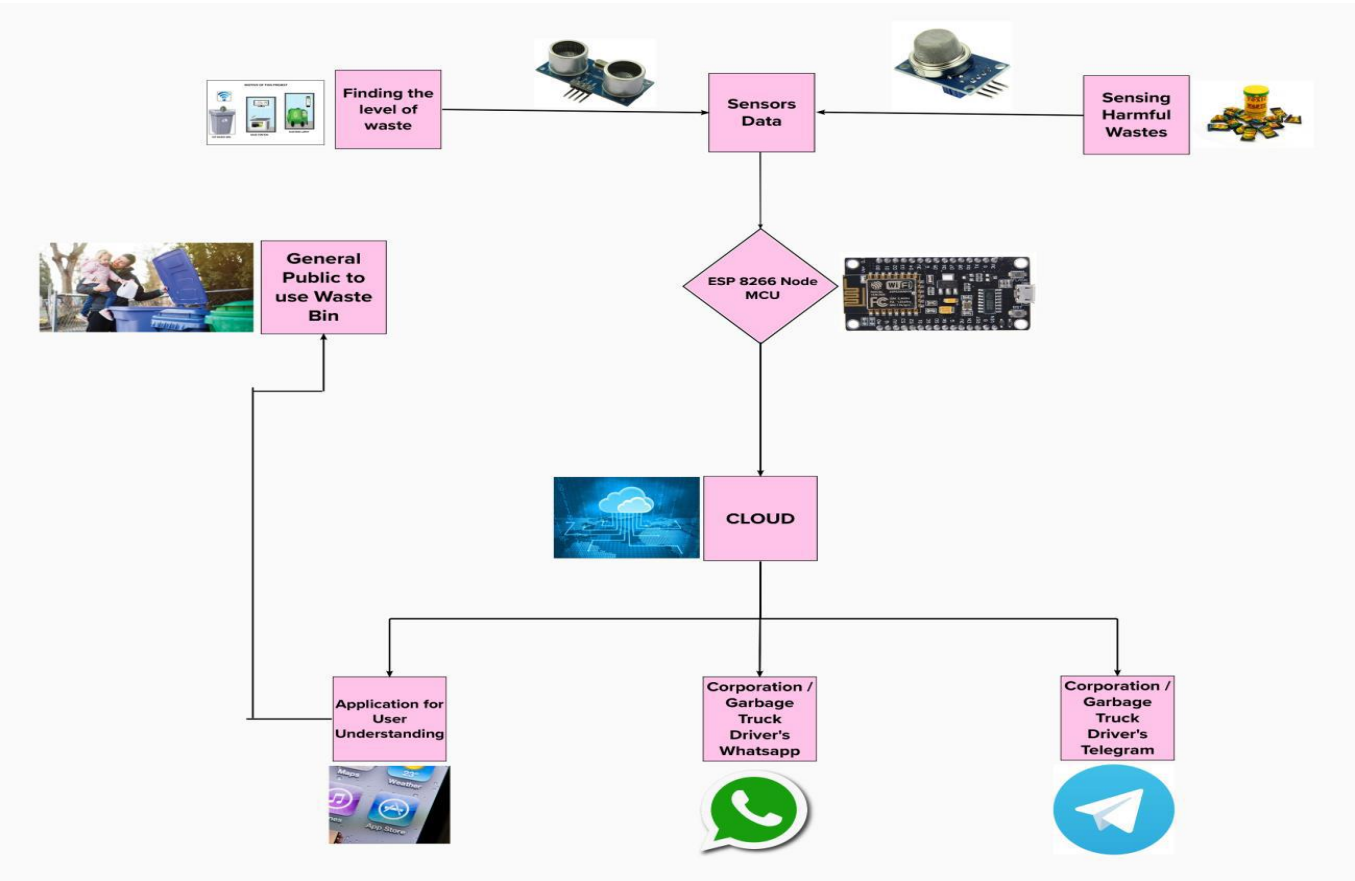
5. PROJECT DESIGN

Data Flow Diagrams



Solution & Technical Architecture

Solution Architecture



Technical Architecture

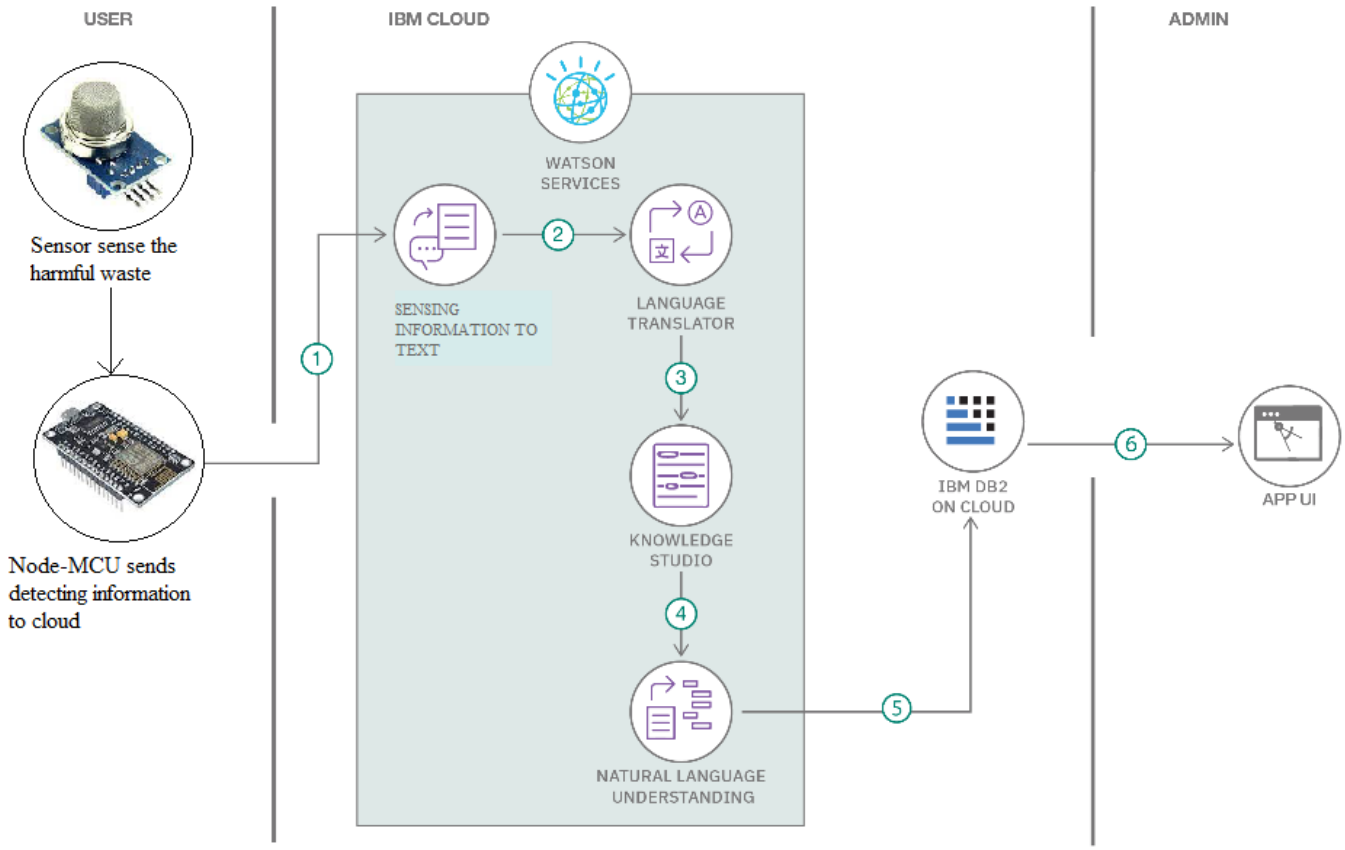


Table-1: Components & Technologies:

S.NO.	Component	Description	Technology
1	User Interface	Waste disposer and the people are our user interact the mobile application	Python
2	Application Logic-1	Sensor detects the harmful waste and it send data to Node-MCU	Python and Node-MCU
3	Application Logic-2	The Node-MCU sends data to the cloud	IBM Watson STT service
4	Application Logic-3	Message send to the user application from the cloud	IBM Watson Assistant
5	Database	Node-MCU is our database	MySQL, NoSQL, etc.
6	Cloud Database	Database service on cloud is IBM Cloud service	IBM DB2

7	File Storage	Node-MCU stores all the collection of waste data and detect the particular harmful waste only so storage is needed	IBM Cloud Storage
8	External API-1	Purpose of external API used to send information to both the user's	IBM Weather API
9	External API-2	And it also sends information to the appropriate office	IBM Weather API
10	Machine Learning Model	The Machine Learning Model is the intermediator to the cloud and the application.	Embedded System Model
11	Infrastructure (Server / Cloud)	Application deployment and it process by the Node-MCU	Cloud Foundry

Table-2: Application Characteristics:

S.NO.	Characteristics	Description	Technology
1	Open-Source Frameworks	Anyone can use this system by their choices	Smart bin Technology
2	Security Implementations	It secures the home from the disease create by the harmful waste.	Gas sensor detection system
3	Scalable Architecture	It sends normal message and it even send the message in WhatsApp and Gmail also.	Cloud Service
4	Availability	There is a normal waste collection system is only available	Robotic System
5	Performance	This is system detects the harmful waste and dispose immediately	Internet and Cloud

User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
User to deposit the waste	Registration	USN-1	As a user, I can register for install the setup of detection waste in our home.	I can register & accessthe setup through application	High	Sprint - 1

	Login	USN-2	As a user, I got login credential for the acknowledgement of this system.	I can login to conform the registration	Low	Sprint - 1
	Program for harmfulsetup	USN-3	As a user, I want the set up in our home.	I can arrange the setup in home	High	Sprint - 2
	Sending wastage information through Web UI	USN-4	As a collector, I know the information of harmful waste in user's home.	I will user for pack thewaste for collector collect	High	Sprint - 3
User contact corporation	Customer Review App	USN-5	As a user, I want to review about this system.	I can access the chat in this app	High	Sprint - 4

6. PROJECT PLANNING & SCHEDULING

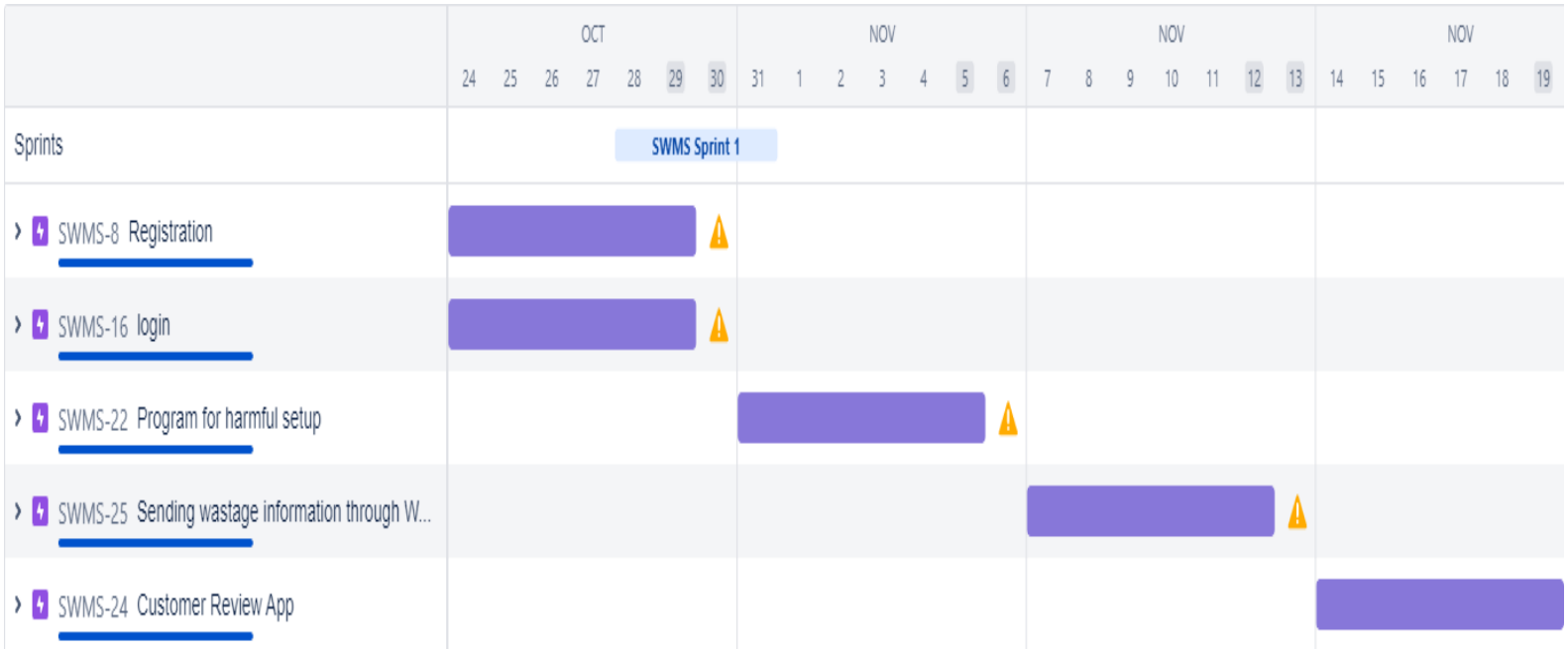
Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for install thesetup of detection waste in our home.	2	High	Vishnuprabhu J Jegatheeswaran S
Sprint-1	Login	USN-2	As a user, I got login credential for the acknowledgement of this system.	1	Low	Jegatheeswaran S Akash V S Dineshpandi G
Sprint-2	Program for harmful setup	USN-3	As a user, I want the set up in our home.	2	High	Vishnuprabhu J Akash V S
Sprint-3	Sending wastage information through Web UI	USN-4	As a collector, I know the information of harmful waste inuser's home.	2	High	Vishnuprabhu J Akash V S
Sprint-4	Customer Review App	USN-5	As a user, I want to review about thissystem.	2	High	Vishnuprabhu J Jegatheeswaran S

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	20	12 Nov 2022
Sprint-4	20	6 Days	14 Nov 2022	19 Nov 2022	20	19 Nov 2022

Reports from JIRA



7. CODING & SOLUTIONING

```
import time
import sys
import ibmiotf.application
import ibmiotf.device

#IBM Watson Device Credentials
```

```
organization = "sfouh6"
deviceType = "Python"
deviceId = "9238"
authMethod = "token"
authToken = "vishnuprabhu923819106057"


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
deviceCli.connect()

while True:
   print("\nDon't use any capital letters to given an input \nSenosr sensing gas is")  #Unavailable of sensors in the wokwi and tinkercad, we give inputs manually
   detect = input()
   Sensing = ()   #Detecting Harmful Wastage
   Location_info = ()  #Sending Location & Contact information

   if detect == "ammonia":     #Harmful material sensing by MQ-137 gas sensor
      Sensing = "Harmful Waste is detected"
      Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

   elif detect == "hydrogen sulfide":   #Harmful material sensing by MQ-136 gas sensor
      Sensing = "Harmful Waste is detected"
      Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
```

```
    elif detect == "methane":     #Harmful material sensing by TGS-2611 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

    else:
        Sensing = "Harmful Waste is not detected"
        Location_info = "Everything is normal in the particular house"

    data = { 'Sensing' : Sensing, 'Location_info' : Location_info}
        #print data

    def myOnPublishCallback():
        print ("Published Sensing data - %s " % Sensing, "\nLocation_info - %s" %Location_info, "to IBM
Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)

    if not success:
            print("Not connected to IoTF")
    time.sleep(1)


# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

sprint 2.py - C:\Users\ELCOT\Desktop\Smart Waste Management System for Metropolitan cities\Z - Programmaing\Sensing Program\sprint 2.py (3.7.0)

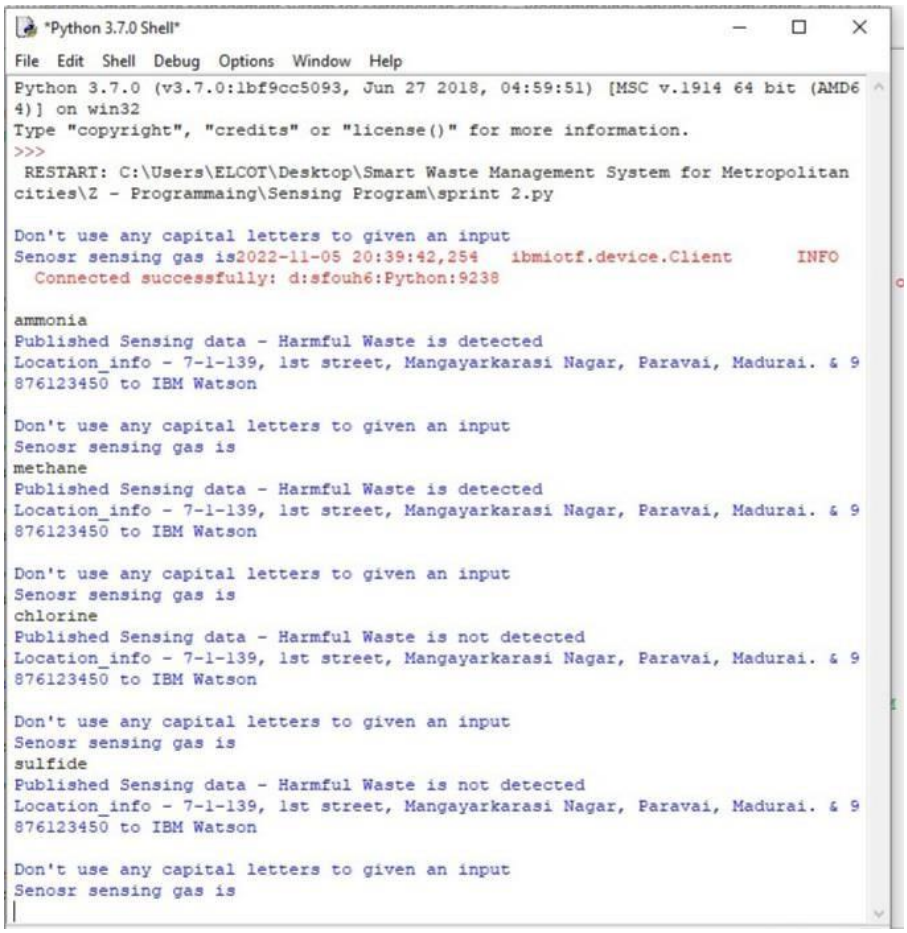
File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
import ibmiotf.device

#Provide your IBM Watson Device Credentials
organization = "sfouh6"
deviceType = "Python"
deviceId = "9238"
authMethod = "token"
authToken = "vishnuprabhu923819106057"


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
deviceCli.connect()

while True:
    print("\nDon't use any capital letters to given an input \nSenosr sensing gas is")    #Unavailable of sensors in the wokwi and tinkercad, we give inputs manually
    detect = input()
    Sensing = ()
    Location_info = ()
    if detect == "ammonia":     #Harmful material sensing by MQ-137 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "hydrogen sulfide":   #Harmful material sensing by MQ-136 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "methane":     #Harmful material sensing by TGS-2611 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    else:
        Sensing = "Harmful Waste is not detected"
```

sprint 2.py - C:\Users\ELCOT\Desktop\Smart Waste Management System for Metropolitan cities\Z - Programmaing\Sensing Program\sprint 2.py (3.7.0)

File Edit Format Run Options Window Help

```
        #.....................................

except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()

# Connect and send a datapoint
deviceCli.connect()

while True:
    print("\nDon't use any capital letters to given an input \nSenosr sensing gas is")    #Unavailable of sensors in the wokwi and tinkercad, we give inputs manually
    detect = input()
    Sensing = ()
    Location_info = ()
    if detect == "ammonia":     #Harmful material sensing by MQ-137 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "hydrogen sulfide":   #Harmful material sensing by MQ-136 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "methane":     #Harmful material sensing by TGS-2611 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    else:
        Sensing = "Harmful Waste is not detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

    data = { 'Sensing' : Sensing, 'Location_info' : Location_info}
        #print data
    def myOnPublishCallback():
        print ("Published Sensing data - %s " % Sensing, "\nLocation_info - %s" %Location_info, "to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
            print("Not connected to IoTF")
    time.sleep(1)


# Disconnect the device and application from the cloud
deviceCli.disconnect()
```



8. TESTING

Test Cases

Test case ID	Feature Type	Component	Test Scenario	Pre-Requisite
LoginPage_TC_OO1	Registration	Home Page	Install and register when users on using account	Easily accessable
LoginPage_TC_OO2	Web UI	Home Page	Verify the UI elements in Login	Verified only
LoginPage_TC_OO3	Functional	Login page	Search Functionality test cases	Easily sharing
LoginPage_TC_OO4	Functional	Login page	Enter User Name, Enter Password	Secure and Safety

Steps To Execute	Test Data	Expected Result	Actual Result	Status	Commnets
Install the app	Required App	Login should displayed	Working as expected	Pass	Easy to access
Register a Account in app	Required App	Application run by UI	Working as expected	Pass	Steps are clear
Make your app mobile friendly	Username: aaaa password: 00456	User should navigate to user account homepage	Working as expected	Pass	Performed Well
Customer data	Username: Nethaji password: 88876	Application should show 'Incorrect email or password ' validation message.	Working as expected	Pass	High Security

User Acceptance Testing

1. Purpose of Document

The purpose of this document is to briefly explain the test coverage and open issues of the [Smart Waste Management] project at the time of the release to User Acceptance Testing (UAT).

2. Defect Analysis

Resolution	Severity 1	Severity 2	Severity 3	Severity 4	Subtotal
By Design	10	3	2	3	20
Duplicate	1	0	2	0	4
External	2	3	0	1	6
Fixed	8	2	3	8	20
Not Reproduced	0	0	1	0	1
Skipped	0	0	1	1	2
Won't Fix	0	5	2	1	8
Totals	21	13	11	14	61

3. Test Case Analysis

Section	Total Cases	Not Tested	Fail	Pass
Print Engine	7	0	0	7
Client Application	30	0	0	25
Security	2	0	0	2
Outsource Shipping	3	0	0	3
Exception Reporting	8	0	0	8
Final Report Output	4	0	0	4
Version Control	2	0	0	2

9. RESULTS

Performance Metrics

			NFT - Risk Assessment						
S. No.	Project Name	Scope\ feature	Functional Changes	Hardware Changes	Software Changes	Impact of Downtime	Load/ Volume Changes	Risk Score	Justification
1.	Smart Waste Management for Metropolitan Cities	New	High	No Change	Moderate	High Impact	>70 to 100%	ORANGE	Protect from disease for peoples

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- Easy to identify the harmful waste
- Protect from the diseases for disposing the waste immediately

- Garbage collector automatically comes and collect the waste. Hence, save the time of disposing the waste.

DISADVANTAGES

- It creates the laziness from the people to dispose the waste.
- Installation cost is not affordable for everyone.

11. CONCLUSION

In this harmful waste detector is very useful for people to secure from the disease. For installing the setup, we are making the disease-free environment in our society.

12. FUTURE SCOPE

In future, we are extending our project model from each and every part of our country to create a disease-free society. This is helps to make a disease-free environment for current and upcoming generations.

13. APPENDIX

Source Code

```
import time
import sys
import ibmiotf.application
import ibmiotf.device

#IBM Watson Device Credentials
organization = "sfouh6"
deviceType = "Python"
deviceId = "9238"
authMethod = "token"
authToken = "vishnuprabhu923819106057"
```

```
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
deviceCli.connect()

while True:
   print("\nDon't use any capital letters to given an input \nSenosr sensing gas is")  #Unavailable of sensors in the wokwi and tinkercad, we give inputs manually
   detect = input()
   Sensing = ()
   Location_info = ()

   if detect == "ammonia":     #Harmful material sensing by MQ-137 gas sensor
      Sensing = "Harmful Waste is detected"
      Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

   elif detect == "hydrogen sulfide":   #Harmful material sensing by MQ-136 gas sensor
      Sensing = "Harmful Waste is detected"
      Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
   elif detect == "methane":     #Harmful material sensing by TGS-2611 gas sensor
      Sensing = "Harmful Waste is detected"
      Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

   else:
      Sensing = "Harmful Waste is not detected"
      Location_info = "Everything is normal in the particular house"
```

```
    data = { 'Sensing' : Sensing, 'Location_info' : Location_info}
        #print data

    def myOnPublishCallback():
        print ("Published Sensing data - %s " % Sensing, "\nLocation_info - %s" %Location_info, "to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)

    if not success:
            print("Not connected to IoTF")
    time.sleep(1)


# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

GitHub & Project Demo Link

GitHub Link

https://github.com/IBM-EPBL/IBM-Project-20735-1659761425

****All the deliverables are submitted as per the guidelines****

Project Demo Link

https://drive.google.com/file/d/1K_AT4ZY4Zf-HKSGvCoER9a_fHeq-ELkv/view?usp=drivesdk