Nalaiya Thiran -IBM Project

Batch No: B2 – 2M4E

KONGU ENGINEERING COLLEGE

Department of Computer Science and Engineering

Smart Waste Management System For Metropolitan Cities

Team ID: PNT2022TMID04391

Team Members:

737819CSR108	MOHAMED ROSHAN M
737819CSR050	HARI PRASATH P
737819CSR062	HARSHIT R
737819CSR165	ROHITH VIGNESH E

Project Guide:

Industry mentor: Mr. Dinesh

Faculty Mentor: Dr. R R Rajalaxmi

ABSTRACT

One issue that most cities and municipalities are dealing with currently, is the degradation of environmental cleanliness with reference to waste management. This is a result of improper garbage collection management. Dumping garbage onto the streets and in public areas is a common synopsis found in all developing countries and this mainly ends up affecting the environment and creating several unhygienic conditions. To avoid improper garbage management and to create a hygienic environment, the concept of automation is used in waste management system. Any city being referred to as a "smart city" is because of its orderly and tidy surroundings. But currently, many issues including those related to smart grids, smart environments, and smart living are faced. Today, cities and metropolitan areas' top priority is proper garbage management.

Traditional waste management techniques are too simplistic to create an effective and reliable waste management. The ideology put forward includes hardware and software technologies i.e. connecting Wi-Fi system to the normal dustbin in order to provide free internet facilities to the user for a particular period of time. The technology awards the user for keeping the surrounding clean and thus work hand in hand for the proper waste management in a locality. The smart bin uses multiple technologies - firstly the technology for measuring the amount of trash dumped and secondly the movement of the waste and lastly sending necessary signals and connecting the user to the WiFi system. The proposed system will function on client server model, a cause that will assure clean environment, good health, and pollution free society.

INDEX

INDEX

CHAPTER 1: INTRODUCTION

1.1 Project Overview

Smart waste management is an innovative approach to handling and collecting waste. Based on IoT (Internet of Things) technology, smart waste management provides data on waste generation patterns.

Our Smart waste management solution uses sensors placed in garbage bins to measure fill levels and notifies city collection services when bins are ready to be emptied. There are load and ultrasonic sensors placed to continuously monitor the bins. This data is sent to the cloud (via a microcontroller that is connected to Wi-Fi) where it is stored after which it is processed further. When the levels exceed a certain limit, a notification is sent to the garbage collector via a web application.

Over time, historical data collected by sensors can be used to identify fill patterns, optimize driver routes and schedules, and reduce operational costs.

1.2 Purpose

Around 2.1 billion tonnes of municipal solid waste is generated annually around the globe. Population growth and rapid urbanization lead to a huge increase in waste generation, so the traditional methods of waste collection have become inefficient and costly. This system cannot measure the fullness levels of containers, and as a result, half-full containers can be emptied, and in contrast, pre-filled ones need to wait until the next collection period comes. Moreover, since drivers collect empty bins, predefined collection routes of the system cause waste of time, an increase in fuel consumption, and excessive use of resources.

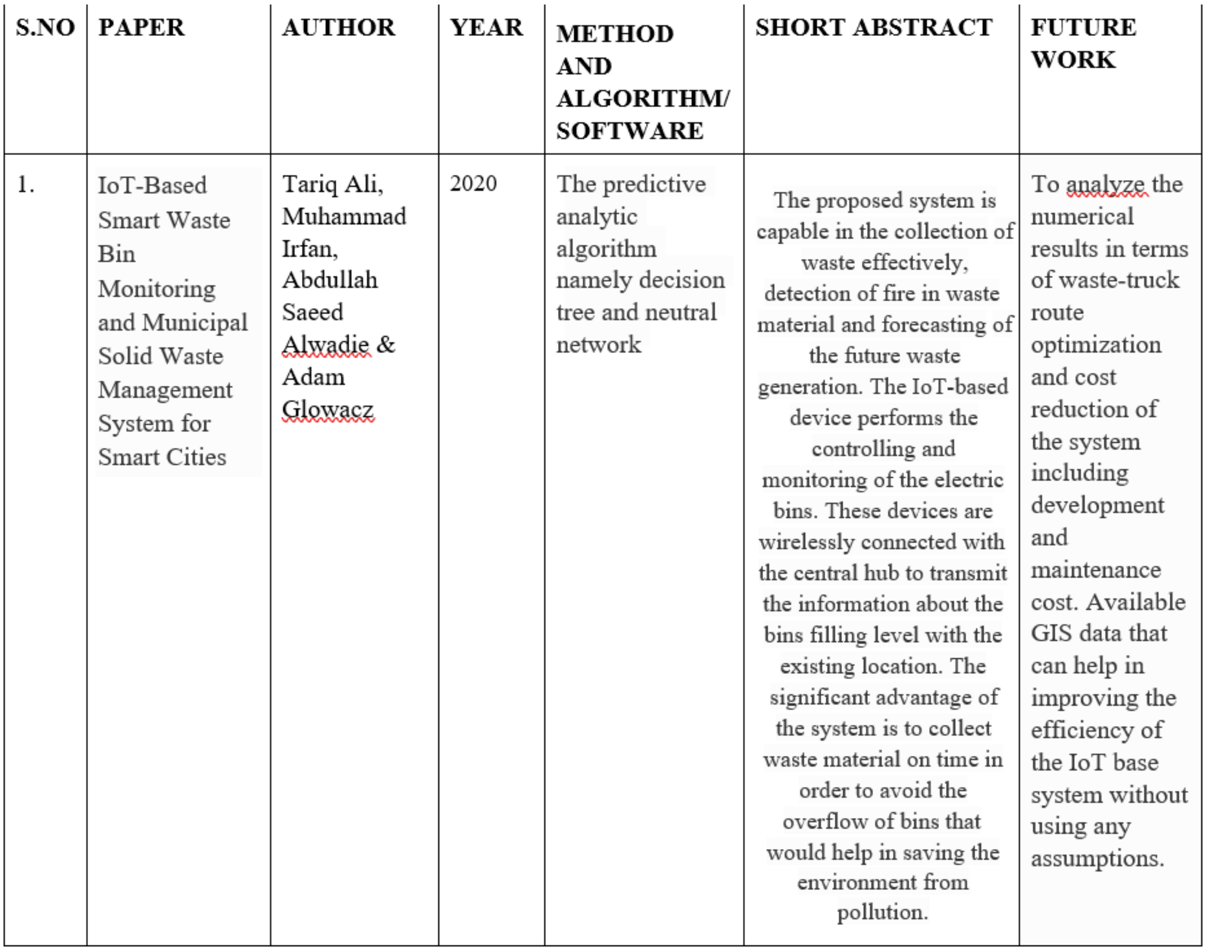
In today's ever-technological world, an innovative and data-driven approach is the only way forward, the waste sector needs a solution that empowers event-driven waste collection. The most efficient way this extraordinary amount of waste can be solved is through smart waste management without obsolete methods of waste collection. This empowers municipalities, cities, and waste collectors to optimize their waste operations, become more sustainable, and make more intelligent business decisions.

CHAPTER 2: LITERATURE SURVEY

2.1 Existing Problem

Around 80% of waste collections happen at the wrong time. Late waste collections lead to overflowing bins, unsanitary environments, citizen complaints, illegal dumping, and increased cleaning and collection costs. Early waste collections mean unnecessary carbon emissions, more traffic congestion, and higher running costs. The idea of smart garbage bins and systems have been in discussion for quite a long time. The technologies used at disposal to develop this smart system have also evolved, Internet of Things (IoT). Each idea seems to be similar but is slightly different at its core and our proposed work is no exception from the same. After the IoT field, finding its hold in our lives, this is our original plan for designing a smart garbage collection system which has provision for citizen participation and analysis of data for better decision making. At hardware level, the smart system is a garbage bin with ultrasonic sensor, a micro-controller and Wi-Fi module for transmission of data. The worldwide implementation of Internet of Things is possible with a Cloud centric vision. This work exploits the future possibilities, key technologies and application that are likely to drive IoT research. But a strong foundation to our work is provided, where the basics and applications of Arduino board is explained . It is quite interesting as it implements a GAYT (Get As You Throw) system concept as a way to encourage recycling among citizens. As we would discuss further, the citizen participation part of our system is quite influenced by their work idea of smart garbage bins and systems have been in discussion for quite a long time. The technologies used at disposal to develop this smart system have also evolved, Internet of Things (IoT). Each idea seems to be similar but is slightly different at its core and our proposed work is no exception from the same. After the IoT field, finding its hold in our lives, this is our original plan for designing a smart garbage collection system which has provision for citizen participation and analysis of data for better decision making. At hardware level, the smart system is a garbage bin with ultrasonic sensor, a micro-controller and Wi-Fi module for transmission of data. The worldwide implementation of Internet of Things is possible with a Cloud centric vision. This work exploits the future possibilities, key technologies and application that are likely to drive IoT research. But a strong foundation to our work is provided, where the basics and applications of Arduino board is explained . It is quite interesting as it implements a GAYT (Get As You Throw) system concept as a way to encourage recycling among citizens. As we would discuss further, the citizen participation part of our system is quite influenced by their work

2.2 References



S.NO	PAPER	AUTHOR	YEAR	METHOD AND ALGORITHM/ SOFTWARE	SHORT ABSTRACT	FUTURE WORK
1.	IoT-Based Smart Waste Bin Monitoring and Municipal Solid Waste Management System for Smart Cities	Tariq Ali, Muhammad Irfan, Abdullah Saeed Alwadie & Adam Glowacz	2020	The predictive analytic algorithm namely decision tree and neutral network	The proposed system is capable in the collection of waste effectively, detection of fire in waste material and forecasting of the future waste generation. The IoT-based device performs the controlling and monitoring of the electric bins. These devices are wirelessly connected with the central hub to transmit the information about the bins filling level with the existing location. The significant advantage of the system is to collect waste material on time in order to avoid the overflow of bins that would help in saving the environment from pollution.	To analyze the numerical results in terms of waste-truck route optimization and cost reduction of the system including development and maintenance cost. Available GIS data that can help in improving the efficiency of the IoT base system without using any assumptions.

S.NO	PAPER	AUTHOR	YEAR	METHOD AND ALGORITHM/ SOFTWARE	SHORT ABSTRACT	FUTURE WORK
2.	Real-time smart garbage bin mechanism for solid waste management in smart cities	Dominic Abugaa, N.SRaghavab	2021	Dijkstra's algorithm and Geographical Information System (GIS).	This paper focuses on a real-time smart garbage bin mechanism for solid waste management in smart cities.. The mechanism proposed accesses real-time information of any smart garbage bin deployed across the city and helps to resolve the problem of waste overflow from garbage bins and keep the smart cities clean.	Future work should consider using WSN in the real-time waste collection and management by integrating the GIS maps into the system for precise location identification of the nodes. In addition, the future work may consider applicability of IoT in the implementation of the proposed system.
3.	Optimal Management of Solid Waste in Smart Cities using Internet of Things	Sahar Idwan, Imran Mahmood, Junaid Ahmed Zubairi & Izzeddin Matar	2019	multiple trucks routing algorithm (MITRA), Genetic algorithms, single truck routing algorithm (SITRA).	The smart dumpsters are equipped with the sensors that measure levels of waste and a controller to send updates to the central management system using wireless network. Our target is to improve the waste collection process by reducing the congestion on the road, the service time spent and the overall trip length.	The model we initially constructed was enhanced to multiple dimensions in this article which allowed us to apply it to large metropolitan areas.
4.	A review of	A.A.I. Shah1	2020	Definition of	Through smart cities,	Additionally,

	IoT-based smart waste level monitoring system for smart cities	S.S.M. Fauzi2, R.A.J.M. Gining3,T.R.Razak4, M.N.F.Jamaluddin5, R. Maskat6		research questions, searching for relevant papers, screening papers, keywording of abstracts, and data extraction and mapping.	necessary modern facilities using ICT emerging technologies such as the internet of things (IoT) had been installed to ensure the sustainability of the city. In the perspective of waste management, several different IoT-based solutions also had been proposed as an alternative to monitor and to ensure the health of communities. This paper reviews existing IoTbased solutions in smart cites' waste level management system to bring together the state-of-the-art.	to enhance the efficiency of a smart waste level monitoring system, a monitoring system should be made available as well so that it can benefit analysts greatly as they will have the amount of data available to be analysed that will lead to better decision making in the future.
5.	Internet-ofThings-Based Smart Cities: Recent Advances and Challenges	Yasir Mehmood (ym@comnets.unibremen.de) received his Master's in electrical (telecommunications) engineering from the Military College of Signals	2017	building IoT platforms, application softwares, and service-related offerings.	The appealing IoT services and big data analytics are enabling smart city initiatives all over the world. These services are transforming cities by improving infrastructure and transportation systems, reducing traffic congestion, providing waste management, and improving the quality of human life. In this	we unearth several IoTrelated open research challenges to give future directions. It was launched by the European Commission, and aims to develop the core future technologies in

S.NO	PAPER	AUTHOR	YEAR	METHOD AND ALGORITHM/ SOFTWARE	SHORT ABSTRACT	FUTURE WORK
		(MCS), National University of Science and Technology (NUST) Islamabad, Pakistan			article, we devise a taxonomy to best bring forth a generic overview of the IoT paradigm for smart cities, integrated ICT, network types, possible opportunities and major requirements.	the IoT paradigm.

2.3 PROBLEM STATEMENT:

SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITAN CITIES

TECHNOLOGY:

IOT

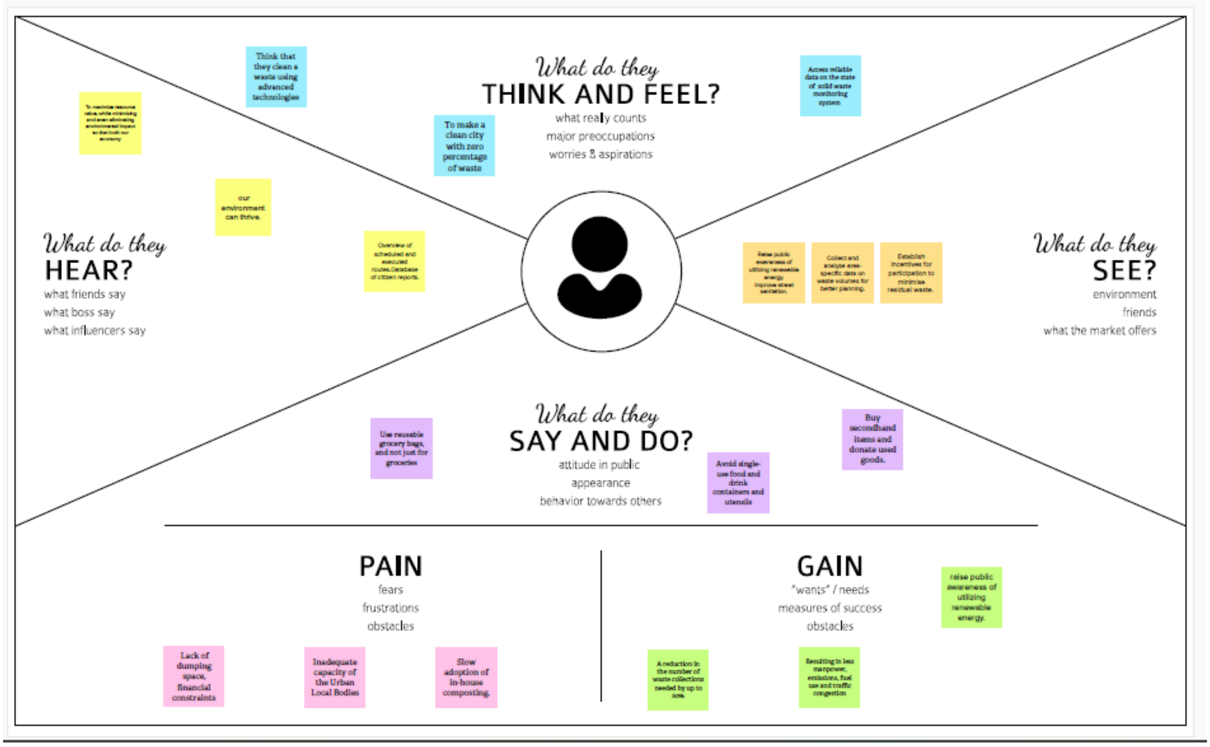
PROBLEM STATEMENT:

Urban India generates tonnes of wastes annually. Our country faces major challenge associated with waste management. Conventional garbage collection is not efficient since the authorities are not notified until the waste bin is full, and this leads to overflow of waste material. Efficient way of waste disposed garbage is essential for a sustainable and clean India.

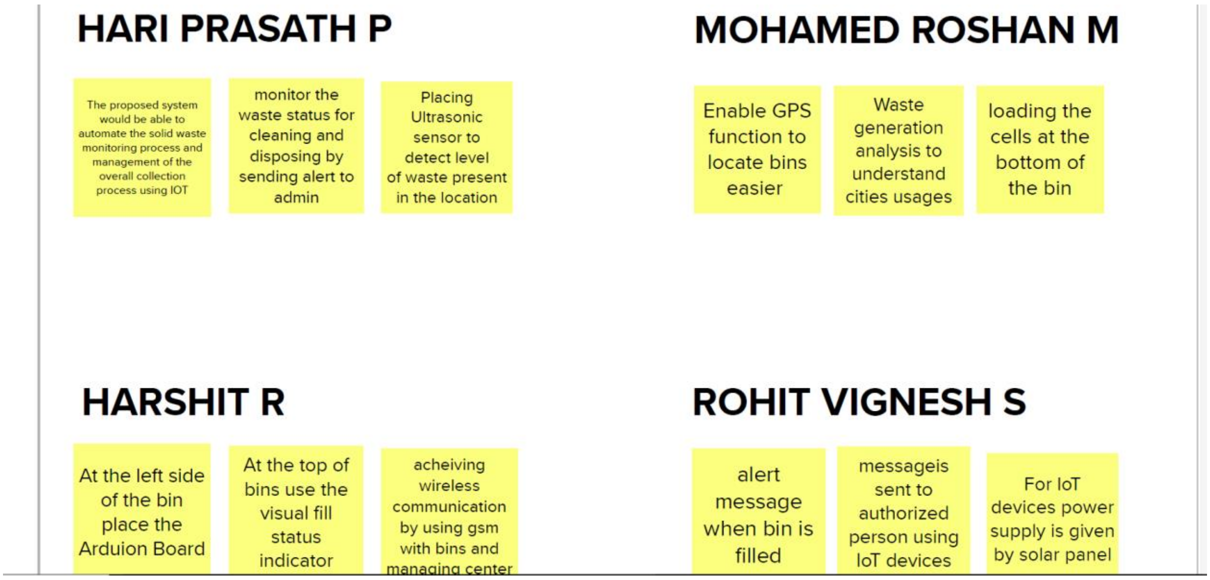
This project smart waste management using IOT based waste bin for collection and monitoring the level of waste inside bin. The system is implemented using two ultrasonic sensors which is being controlled by Node MCU. One of the ultrasonic sensor detects the level of the waste in the bin and other detects the person approaching the bin to dispose the waste. This detection helps in automatic opening and closing of the lid. Servo motor is connected to the lid which serves the action of closing and opening of the lid. In this system, level of waste in the bin will be sent to concerned authorities. The IOT data is stored and monitored using app.

CHAPTER 3: IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas

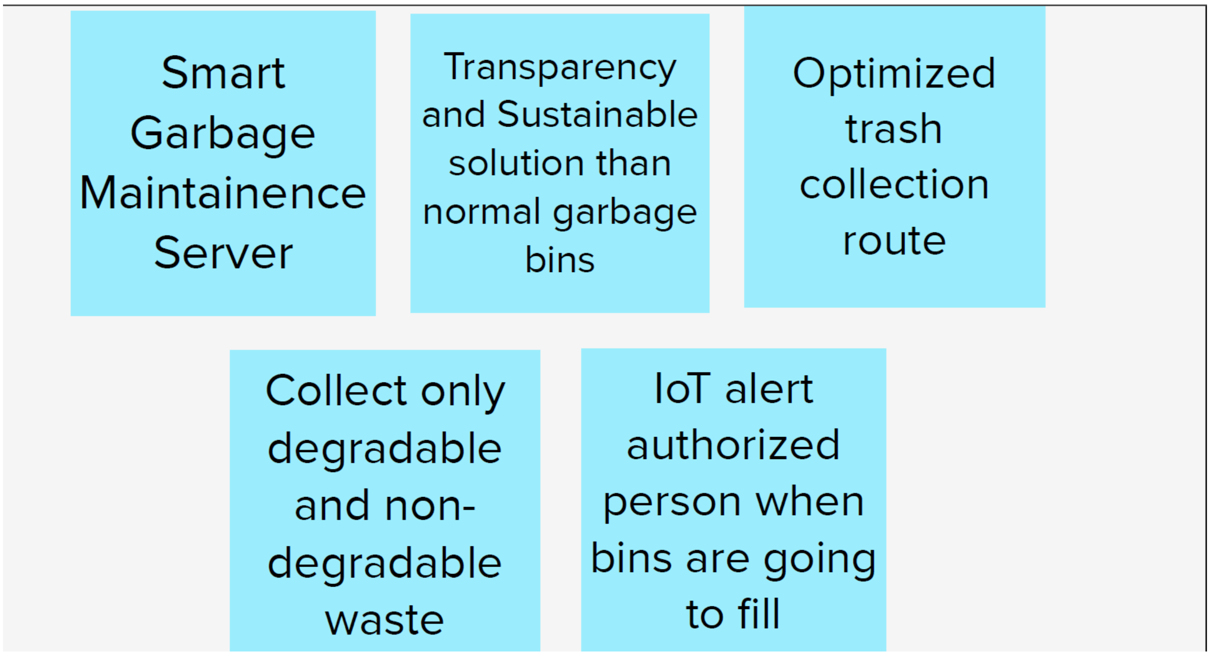

What do they THINK AND FEEL?
what really counts
major preoccupations
worries & aspirations
Think that they clean a waste using advanced technologies
To make a clean city with zero percentage of waste
Access reliable data on the state of solid waste monitoring system
What do they HEAR?
what friends say
what boss say
what influencers say
To maximize resource value, while minimizing and even eliminating environmental impact so that both our economy
our environment can thrive.
Overview of scheduled and executed routes.Database of citizen reports.
What do they SEE?
environment
friends
what the market offers
Raise public awareness of utilizing renewable energy. Improve street sanitation.
Collect and analyze area-specific data on waste volumes for better planning.
Establish incentives for participation to minimise residual waste.
What do they SAY AND DO?
attitude in public
appearance
behavior towards others
Use reusable grocery bags, and not just for groceries
Avoid single-use food and drink containers and utensils
Buy secondhand items and donate used goods.
PAIN
fears
frustrations
obstacles
Lack of dumping space, financial constraints
Inadequate capacity of the Urban Local Bodies
Slow adoption of in-house composting.
GAIN
"wants" / needs
measures of success
obstacles
raise public awareness of utilizing renewable energy.
A reduction in the number of waste collections needed by up to 80%.
Resulting in less manpower, emissions, fuel use and traffic congestion

3.2 Ideation & Brainstorming


HARI PRASATH P
The proposed system would be able to automate the solid waste monitoring process and management of the overall collection process using IOT
monitor the waste status for cleaning and disposing by sending alert to admin
Placing Ultrasonic sensor to detect level of waste present in the location
MOHAMED ROSHAN M
Enable GPS function to locate bins easier
Waste generation analysis to understand cities usages
loading the cells at the bottom of the bin
HARSHIT R
At the left side of the bin place the Arduion Board
At the top of bins use the visual fill status indicator
acheiving wireless communication by using gsm with bins and managing center
ROHIT VIGNESH S
alert message when bin is filled
messageis sent to authorized person using IoT devices
For IoT devices power supply is given by solar panel

\

3.2.2 Group ideas



Smart Garbage Maintainence Server

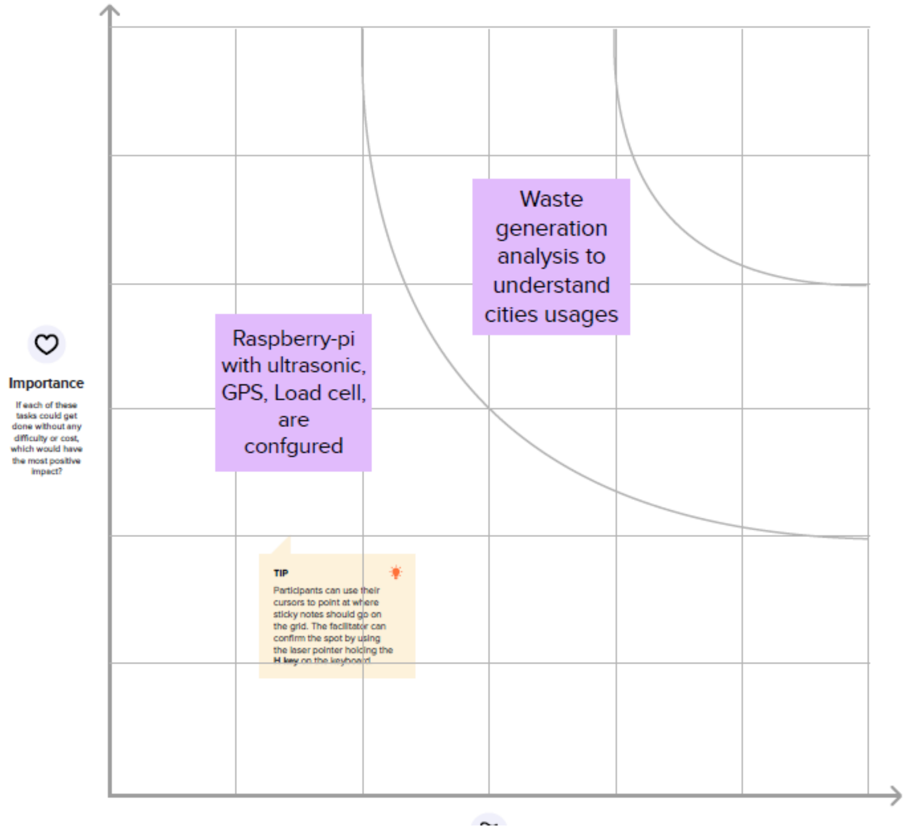
Transparency and Sustainable solution than normal garbage bins

Optimized trash collection route

Collect only degradable and non-degradable waste

IoT alert authorized person when bins are going to fill

3.2.3Prioritize



Importance
If each of these tasks could get done without any difficulty or cost, which would have the most positive impact?

Waste generation analysis to understand cities usages

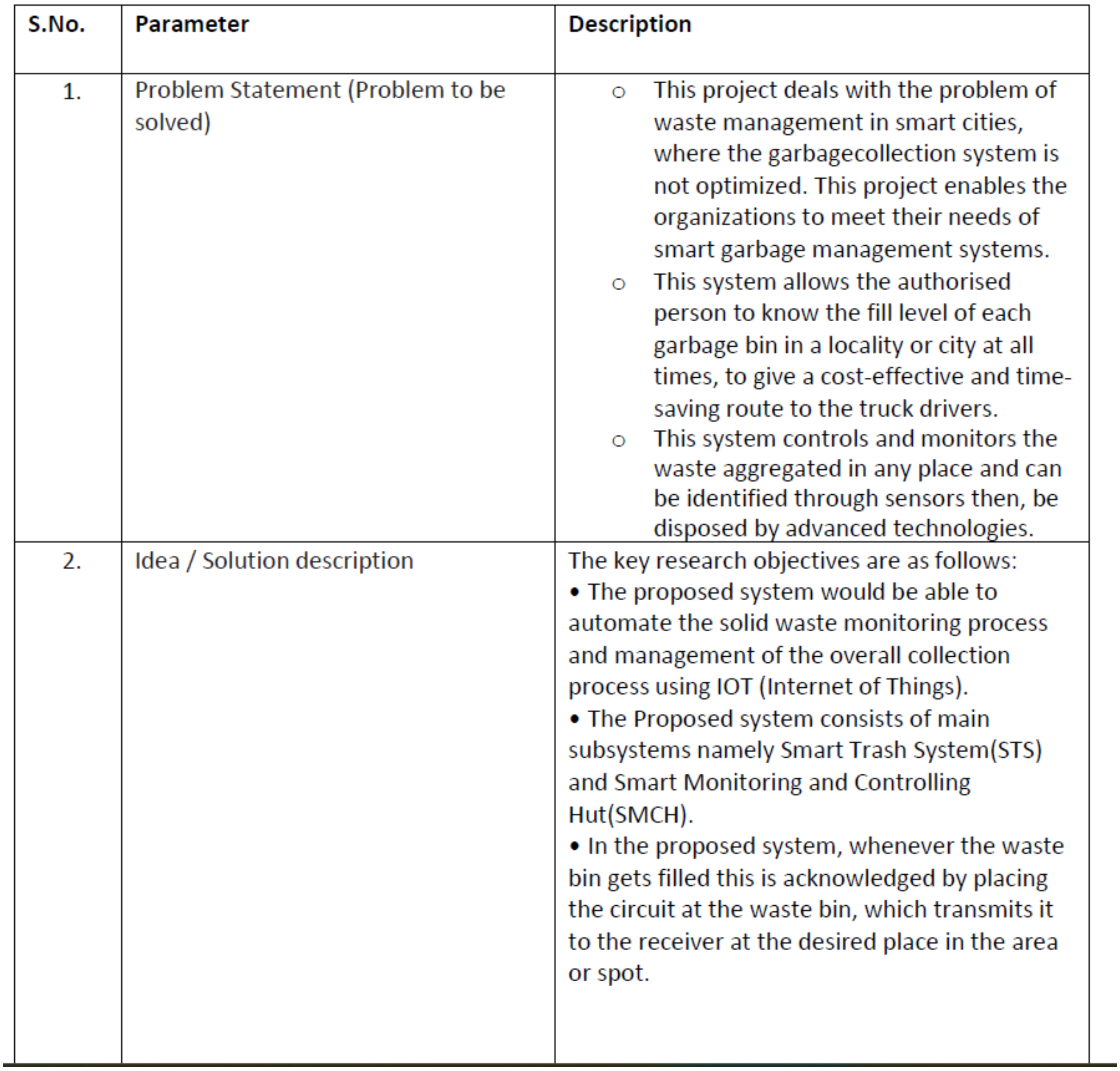
Raspberry-pi with ultrasonic, GPS, Load cell, are confgured

TIP

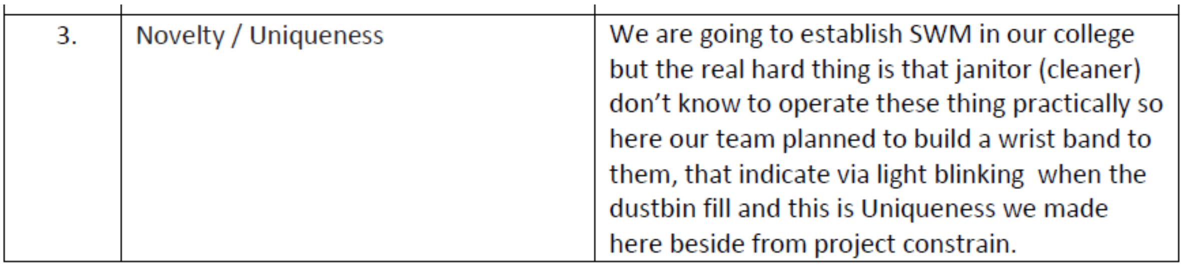
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

Proposed Solution Template:

Project team shall fill the following information in proposed solution template.



S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	○ This project deals with the problem of waste management in smart cities, where the garbagecollection system is not optimized. This project enables the organizations to meet their needs of smart garbage management systems. ○ This system allows the authorised person to know the fill level of each garbage bin in a locality or city at all times, to give a cost-effective and time-saving route to the truck drivers. ○ This system controls and monitors the waste aggregated in any place and can be identified through sensors then, be disposed by advanced technologies.
2.	Idea / Solution description	The key research objectives are as follows: • The proposed system would be able to automate the solid waste monitoring process and management of the overall collection process using IOT (Internet of Things). • The Proposed system consists of main subsystems namely Smart Trash System(STS) and Smart Monitoring and Controlling Hut(SMCH). • In the proposed system, whenever the waste bin gets filled this is acknowledged by placing the circuit at the waste bin, which transmits it to the receiver at the desired place in the area or spot.



3.	Novelty / Uniqueness	We are going to establish SWM in our college but the real hard thing is that janitor (cleaner) don’t know to operate these thing practically so here our team planned to build a wrist band to them, that indicate via light blinking when the dustbin fill and this is Uniqueness we made here beside from project constrain.

4.	Social Impact / Customer Satisfaction	From the public perception as worst impacts of present solid waste disposal practices are seen direct social impacts such as neighbourhood of landfills to communities, breeding of pests and loss in property values

5.	Business Model (Revenue Model)	Waste Management organises its operations into two reportable business segments: Solid Waste, comprising the Company’s waste collection, transfer, recycling and resource recovery, and disposal services, which are operated and managed locally by the Company’s various subsidiaries, which focus on distinct geographic areas; and Corporate and Other, comprising the Company’s other activities, including its development and operation of landfill gas-to-energy facilities in the INDIA, and its recycling brokerage services, as well as various corporate functions.
6.	Scalability of the Solution	In this regard, smart city design has been increasingly studied and discussed around the world to solve this problem. Following this approach, this paper presented an efficient IoT-based and real-time waste management model for improving the living environment in cities, focused on a citizen perspective. The proposed system uses sensor and communication technologies where waste data is collected from the smart bin, in real-time, and then transmitted to an online platform where citizens can access and check the availability of the compartments scattered around a city.

CHAPTER 4: REQUIREMENT ANALYSIS

4.1 Functional Requirements:

FR No.	Functional Requirement (Epic)	Sub Requirement (Story / Sub-Task)
FR-1	Detailed bin inventory.	All monitored bins and stands can be seen on the map, and you can visit them at any time via the Street View feature from Google. Bins or stands are visible on the map as green, orange or red circles. You can see bin details in the Dashboard – capacity, waste type, last measurement, GPS location and collection schedule or pick recognition.
FR-2	Real time bin monitoring.	The Dashboard displays real-time data on fill- levels ofbins monitored by smart sensors. In addition to the % of fill-level, based on the historical data, the tool predicts when the bin will become full, one of the functionalities that are not included even in the best waste management software.. Sensors recognize picks as well; so you can check whenthe bin was last collected. With real-time data and predictions, you can eliminate the overflowing bins and stop collecting half-empty ones.
FR-3	Expensive bins.	We help you identify bins that drive up your collectioncosts. The tool calculates a rating for each bin in termsof collection costs. The tool considers the average distance depobin-discharge in the area. The tool assigns bin a rating (1-10) and calculates distance from depo-bin discharge.

FR-4	Adjust bin distribution.	Ensure the most optimal distribution of bins.Identify areas with either dense or sparse bindistribution. Make sure all trash types are represented within astand. Based on the historical data, you can adjust bin capacity or location where necessary.

4.2 Non-functional Requirements:

Following are the non-functional requirements of the proposed solution

FR No.	Non-Functional Requirement	Description
NFR-1	**Usability**	IoT device verifies that usability is a special and important perspective to analyze user requirements, which can further improve the design quality. In the design process with user experience as the core, theanalysis of users' product usability can indeed help designers better understand users' potential needs in waste management, behavior and experience.
NFR-2	**Security**	Use a reusable bottles Use reusable grocery bags Purchase wisely and recycle Avoid single use food and drink containers.

NFR-3	**Reliability**	creating better working conditions for waste collectors and drivers. Instead of driving the same collection routesand servicing empty bins, waste collectors will spendtheir time more efficiently, taking care of bins that need servicing.
NFR-4	**Performance**	The Smart Sensors use ultrasound technology to measure the fill levels (along with other data) in binsseveral times a day. Using a variety of IoT networks ((NB- IoT, GPRS), the sensors send the data to Sensoneo's Smart Waste Management Software System, a powerful cloud-based platform, for data- driven daily operations, available also as a waste management app. Customers are hence provided data-driven decision making, and optimization of waste collection routes, frequencies, and vehicle loads resulting in route reduction by at least 30%.
NFR-5	**Availability**	By developing & deploying resilient hardware and beautiful software we empower cities, businesses, and countries to manage waste smarter.
NFR-6	**Scalability**	Using smart waste bins reduce the number of bins inside town , cities coz we able to monitor the garbage 24/7 more cost effect and scalability when we moves to smarter.

CHAPTER 5: PROJECT DESIGN

5.1 Data Flow Diagram:

· Collecting the datasets consisting of used cars features and their prices.

· Data Pre-processing – Handling the missing data and removing the noisy data.

· Feature selection – Removing the unnecessary attributes and using only important features.

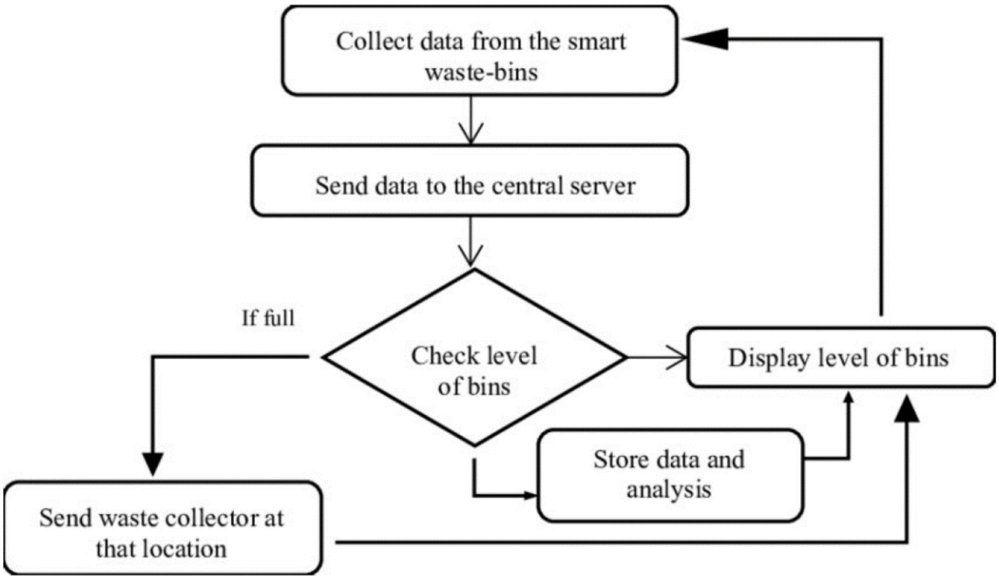
· Provide specifications according to which the solution is defined, managed, and delivered.

· Create a model using regression algorithms.

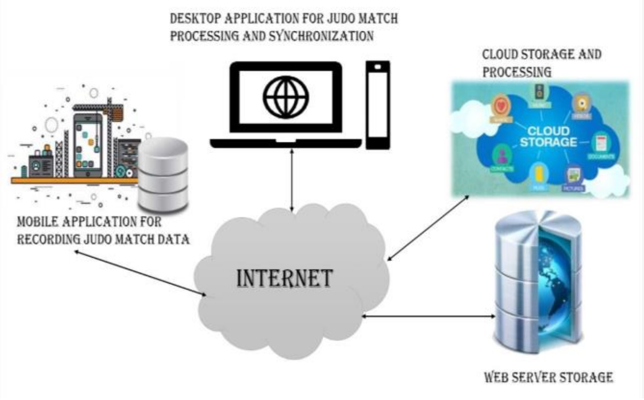
· Train the model using training and testing data.

· Evaluate and find the best model.

· Deploy the model as an application with good UI.


Collect data from the smart waste-bins
Send data to the central server
If full
Check level of bins
Display level of bins
Store data and analysis
Send waste collector at that location

5.2 Solution and Technical Architecture


DESKTOP APPLICATION FOR JUDO MATCH PROCESSING AND SYNCHRONIZATION
CLOUD STORAGE AND PROCESSING
CLOUD STORAGE
MOBILE APPLICATION FOR RECORDING JUDO MATCH DATA
INTERNET
WEB SERVER STORAGE

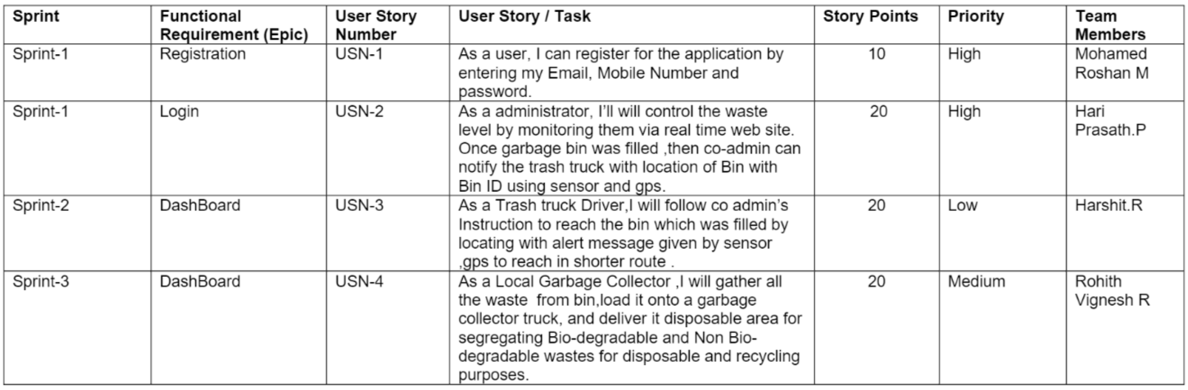
5.3 User Stories

User Type	Functional Requirement (Epic)	User Story Number	User Story/ Task	Acceptance Criteria	Priority	Release
Garbage collector	Visit website	USN – 1	As a user, I can visit website and register with email or mobile number and set password	I can visit the Website and allow the login with my credentials and allow confirmation message from website	high	Sprint - 1
		USN - 2	As a user, I can monitor the bin level of different Bins for clearing the environment	I can get notification of overflowing bins and alert the user and admin for environment free from waste	high	Sprint – 1

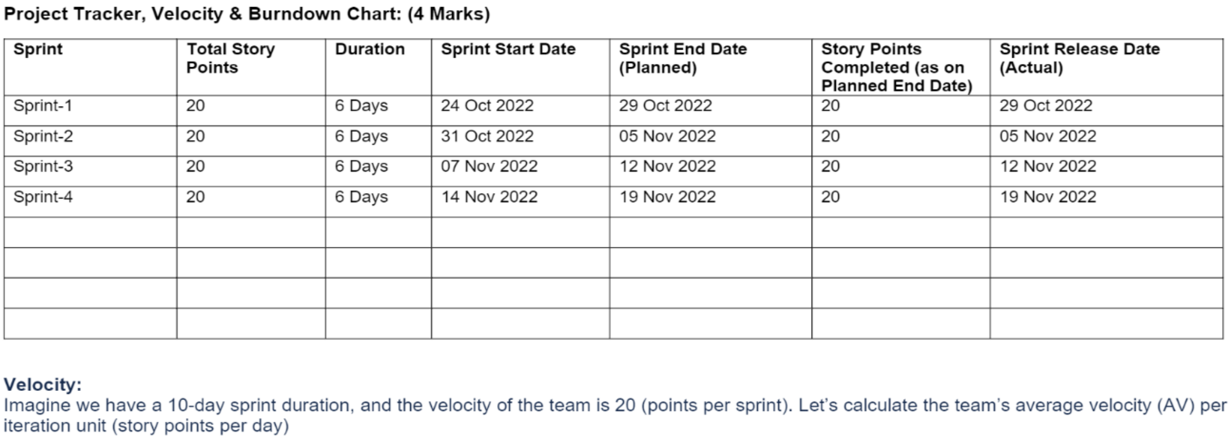
		USN – 3	AS a user, I can monitor the bin's location	I can get location of bins and notification with overflowing level of dumbs.	high	Sprint – 2
		USN – 4	As a user, I can monitor air purity rate	I can get level of air purity in bins and lots of impurities in bins are received by admin.	low	Sprint - 1
Municipal officer	Visit website	USN – 5	As a user, I can monitor bin level and location of different bins	I can intimidate garbage collectors about bin levels	high	Sprint - 1

CHAPTER 6: PROJECT PLANNING & SCHEDULING

6.1 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 the application by entering my Email, Mobile Number and password.	10	High	Mohamed Roshan M
Sprint-1	Login	USN-2	As a administrator, I'll will control the waste level by monitoring them via real time web site. Once garbage bin was filled ,then co-admin can notify the trash truck with location of Bin with Bin ID using sensor and gps.	20	High	Hari Prasath.P
Sprint-2	DashBoard	USN-3	As a Trash truck Driver,I will follow co admin's Instruction to reach the bin which was filled by locating with alert message given by sensor ,gps to reach in shorter route .	20	Low	Harshit.R
Sprint-3	DashBoard	USN-4	As a Local Garbage Collector ,I will gather all the waste from bin,load it onto a garbage collector truck, and deliver it disposable area for segregating Bio-degradable and Non Bio-degradable wastes for disposable and recycling purposes.	20	Medium	Rohith Vignesh R



Project Tracker, Velocity & Burndown Chart: (4 Marks)

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

Velocity:
Imagine we have a 10-day sprint duration, and the velocity of the team is 20 (points per sprint). Let's calculate the team's average velocity (AV) per iteration unit (story points per day)

$$AV = \frac{sprint\ duration}{velocity} = \frac{20}{10} = 2$$

CHAPTER 7: CODING AND SOLUTIONING

7.1 Feature 1:

The main and first feature of the smart waste management is to get the live location of anyone who access the website for putting out a request for garbage collection in their locality. The live location is obtained as a result of the below code.

Web Application to get the Live location:

Index.html

```
<!DOCTYPE html>
<html>

<head>
  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
<meta charset="utf-8">
<meta name="viewport" content="width=device-width">
<title>Waste Management</title>
<link rel="icon" type="image/x-icon" href="/IMAGES/DUMPSTER.png">
<link href="style.css" rel="stylesheet" type="text/css" />
<script src="https://www.gstatic.com/firebasejs/8.10.1/firebase-app.js"></script>
<script src="https://www.gstatic.com/firebasejs/8.10.1/firebase-db.js"></script>
<script type="module">
    // Import the functions you need from the SDKs you need
    import { initializeApp } from "https://www.gstatic.com/firebasejs/9.14.0/firebase-
app.js";
    import { getAnalytics } from "https://www.gstatic.com/firebasejs/9.14.0/firebase-
analytics.js";
    // TODO: Add SDKs for Firebase products that you want to use
    // https://firebase.google.com/docs/web/setup#available-libraries

    // Your web app's Firebase configuration
    // For Firebase JS SDK v7.20.0 and later, measurementId is optional
    const firebaseConfig = {
      apiKey: "AIzaSyCLmn-TzMoUVe9sBa6h56Bd4WnFJtjm0aE",
      authDomain: "ibm-smart-waste.firebaseapp.com",
      databaseURL: "https://ibm-smart-waste-default-rtdb.firebaseio.com",
      projectId: "ibm-smart-waste",
      storageBucket: "ibm-smart-waste.appspot.com",
      messagingSenderId: "426276430128",
      appId: "1:426276430128:web:4f8671bf97c4c9450728f5",
      measurementId: "G-16DJ7XEDK5"
    };

    // Initialize Firebase
    const firebase = initializeApp(firebaseConfig);
    const analytics = getAnalytics(firebase);
  </script>
<script defer src="database.js"></script>
</head>

<body style="background-color:#1F1B24;">
<script src="map.js"></script>


    <div id="map_container">
              <h1 id="live_location_heading" >LOCATION</h1>
              <div id="map"></div>
              <div id="alert_msg">ALERT MESSAGE!</div>
   </div>
</div>
<center><a href="https://www.google.com/maps/place/Perundurai+New+Bus+Stand/@11.273363,77.5778373,17z/data=!3m1!4b1!4m5!3m4!1s0x3ba96d434da63087:0x91b2c203d46019b3!8m2!3d11.273363!4d77.5800313"
  type="button" class="btn btn-dark">Get Dumpster</a></center>

<script
    src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBBLyWj-3FWtCbCXGW3ysEiI2fDfrv2v0Q&callback=myMap"></script></div>
</body>

</html>
```

Style.css:

```
html, body
{
    height: 100%;
    margin: 0px;
    padding:0px;
}
#container
```

```css
{
display: flex;
flex-direction: row;
height: 100%;
width: 100%;
position: relative;
}
#logo_container
{
height: 100%;
width: 12%;
background-color: #C5C6D0;
display: flex;
flex-direction: column;
vertical-align: text-bottom;
}
.logo
{
width:70%;
margin: 5% 15%;

/*  border-radius: 50%; */
}
#logo_3
{
vertical-align: text-bottom;

}
#data_container
{
height: 100%;
width: 20%;
margin-left: 1%;
margin-right: 1%;
display: flex;
flex-direction: column;
}
#data_status
{
height:60%;
width:8%;
margin:7%;
background-color: #691F6E;
display: flex;
flex-direction: column;
border-radius:20px;
}
#load_status
```

```css
{
background-image: url("/Images/KG.png");
background-repeat: no-repeat;
 background-size:     170px;
background-position: left center;
}
#cap_status
{
background-image: url("/Images/dust.png");
background-repeat: no-repeat;
 background-size: 150px;
background-position: left center;
}
.status
{
width: 80%;
height: 40%;
margin:5% 10%;
background-color:#185adc;
border-radius:20px;
display: flex;
     justify-content: center;
     align-items: center;
color: white;
font-size: 60px;


}
.datas
{
width:86%;
margin:2.5% 7%;
height:10%;
   background: url(water.png);
    background-repeat: repeat-x;
    animation: datas 10s linear infinite;

    box-shadow: 0 0 0 6px #98d7eb, 0  20px 35px rgba(0,0,0,1);

}
#map_container
{
height: 100%;
width: 100%;
display: flex;
flex-direction: column;
}
#live_location_heading
```

```
{
margin-top:10%;
text-align: center;
  color:  GREY;
}
#map
{
height: 70%;
width: 90%;
margin-left: 4%;
margin-right:4%;
border: 10px solid white;
border-radius: 25px;
}
#alert_msg
{
width:92%;
height:20%;
margin:4%;
background-color:grey;
border-radius: 20px;
display: flex;
     justify-content: center;
     align-items: center;
color: #41af7f;
font-size: 25px;
font-weight: bold;
}
.lat
{
margin: 0px;
font-size:0px;
}




@keyframes datas{
    0%
    {
       background-position: -500px 100px;
    }
    40%
    {
       background-position: 1000px -10px;
    }

    80% {
        background-position: 2000px 40px;
    }
    100% {
        background-position: 2700px 95px;
    }

}
```

Map.js:

```
const database = firebase.database();

function myMap()
{
var ref1 = firebase.database().ref();

     ref1.on("value", function(snapshot)
      {
        snapshot.forEach(function (childSnapshot) {
            var value = childSnapshot.val();
                        const latitude = value.latitude;
                        const longitude = value.longitude;

                        var latlong = { lat: latitude, lng: longitude}
                        var mapProp =
                        {
                                center: new google.maps.LatLng(latlong),
                                zoom: 10,
                        };
                        var map = new google.maps.Map(document.getElementById("map"),
mapProp);

                        var marker = new google.maps.Marker({ position: latlong });
                        marker.setMap(map);
        });
      }, function (error) {
        console.log("Error: " + error.code);
      });

}
```

Database.Js:

```
import "firebase/database";
const cap_status = document.getElementById('cap_status');
const alert_msg = document.getElementById('alert_msg');

console.log(firebase);
var ref = firebase.database().ref();
```

```
ref.on("value", function(snapshot)
{
snapshot.forEach(function (childSnapshot) {
var value = childSnapshot.val();

const alert_msg_val = value.alert;
const cap_status_val = value.distance_status;


alert_msg.innerHTML= `${alert_msg_val}`;
});
}, function (error) {
console.log("Error: " + error.code);
});
```

7.2 Feature 2:

In this part, the filled level of the bin is measured with the help of IBM IOT Watson platform devices, IBM Cloud interface and Node-RED is used for creating the dashboard nodes that helps us create a UI to display the distance, that is, the fill level of the bin. It also intimates the location of the bin with the fill level and alerts the collection authority if the fill level goes beyond a threshold value.

Node Red Connection With IBM IoT Platform:

```
#include <WiFi.h> //library for wifi

#include <PubSubClient.h> //library for MQTT
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 20, 4);
// credentials of IBM Accounts -
#define ORG "3defta" //IBM organisation id
#define DEVICE_TYPE "hariprasath" // Device type mentioned in ibm watson iot platform
#define DEVICE_ID "12345" // Device ID mentioned in ibm watson iot platform
#define TOKEN "CpL-H1C-Pt4i9iM-F5" // Token
// customise above values -

char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // server
name char publishTopic[] = "iot-2/evt/data/fmt/json";
char topic[] = "iot-2/cmd/led/fmt/String"; // cmd Represent type and command is test format
of strings char authMethod[] = "use-token-auth"; // authentication method  char token[] =
TOKEN;
```

```
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //Client id
//
WiFiClient wifiClient; // creating instance for wificlient
PubSubClient client(server, 1883, wifiClient);
#define ECHO_PIN 12
#define TRIG_PIN 13
float dist;

void setup()
{
Serial.begin(115200);
pinMode(LED_BUILTIN, OUTPUT);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
//pir
pinMode(4, INPUT);
//ledpins
pinMode(23,OUTPUT);
pinMode(2,OUTPUT);
pinMode(4,OUTPUT);
pinMode(15,OUTPUT); lcd.init();

lcd.backl
ight();
lcd.setC
ursor(1,0
);
lcd.print
("");
wifiCon
nect();
mqttCon
nect(); }
float
readcmC
M()
{
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN,HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);
int duration
=pulseIn(ECHO_PIN, HIGH);
return duration * 0.034 / 2;
} void
loop()
{
lcd.cle
ar();
publish
Data();
```

```
delay(
500);
if
(!client
.loop()
) {
mqttConnect(); //function call to connect to IBM
}
}
/* -retrieving to cloud */ void wifiConnect()
{
Serial.print("Connecting to ");
Serial.print("Wifi");
 WiFi.begin("Wokwi-GUEST", "", 6);
while (WiFi.status() != WL_CONNECTED)
{
del
ay(
500
);
Serial.print(".");
}
Serial.print("WiFi connected, IP address: ");
Serial.println(WiFi.localIP());
} void
mqttConn
ect()
{
if (!client.connected())
{
Serial.print("Reconnecting MQTT client to ");
Serial.println(server);
while(!client.connect(clientId, authMethod, token))
{
Serial.print("."); delay(500);
}
initManagedDevice(); Serial.println();
}
}
void initManagedDevice()
{

if (client.subscribe(topic))
{
Serial.println("IBM subscribe to cmd OK");
}
e
l
s
e
{
```

```
Serial.println("subscribe to cmd FAILED");
}
}
void publishData()
{
float cm = readcmCM();
if(digitalRead(34)) //PIR motion detection
{
Serial.println("Motion Detected"); Serial.println("Lid Opened"); digitalWrite(15, HIGH);
}
e
l
s
e
{
digitalWrite(15, LOW);
}
if(digitalRead(34)== true)
{
if(cm <= 100) //Bin level detection
{
digitalWrite(2, HIGH);
Serial.println("High Alert!!!,Trash bin is about to be full");
Serial.println("Lid Closed");
lcd.print("Full! Don't use");
delay(2000);
lcd.clear();
digitalWrite(4,
LOW);
digitalWrite(23,
LOW);
}
else if(cm > 150 && cm < 250)
{
digitalWrite(4, HIGH);
Serial.println("Warning!!,Trash is about to cross 50% of
bin level");  digitalWrite(2,LOW); digitalWrite(23,
LOW);
}
else if(cm > 250 && cm <=400)
{
digitalWrite(23, HIGH);
Serial.println("Bin is
available");
digitalWrite(2,LOW);
digitalWrite(4, LOW);
}
delay(10000);
Serial.println("Lid Closed");
}
if(cm <= 100)
```

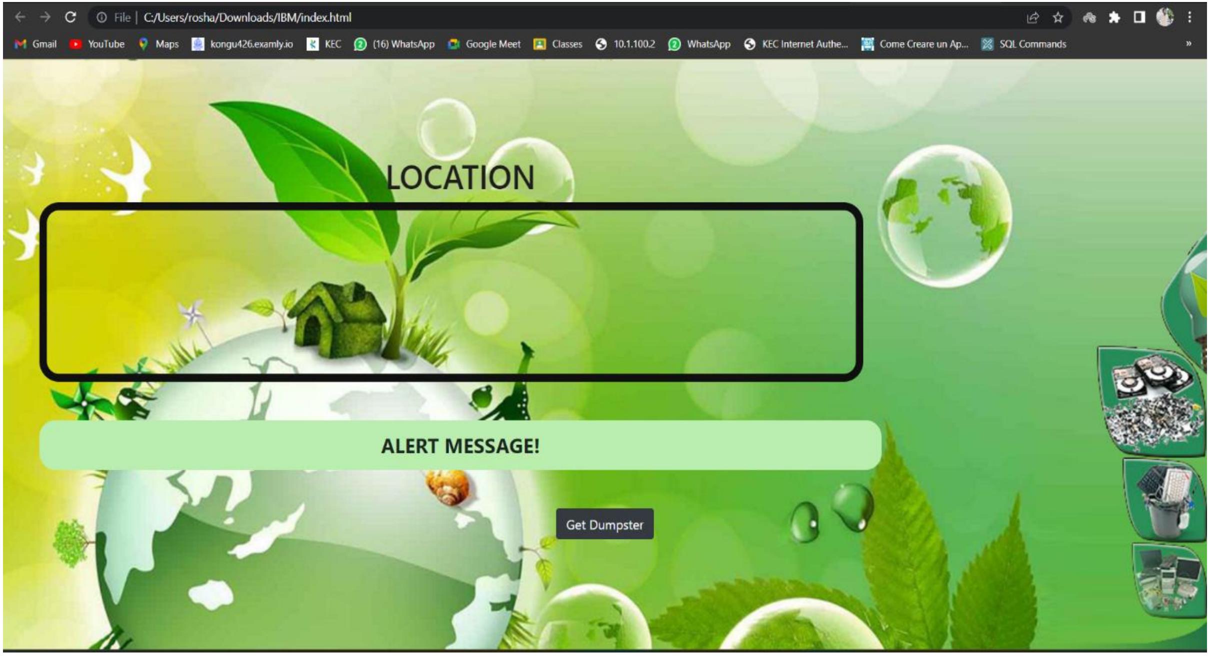
```
{
digitalWrite(21,HIGH);
String payload = "{\"High Alert!!\":\"";
payload +=
cm; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) // if data is uploaded to cloud
successfully,prints publish ok or prints publish failed
{
Serial.println("Publish OK");
}
}
if(cm <= 250)
{
digitalWrite(22,HIGH);
String payload =
"{\"Warning!!\":\"";
payload += dist; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending distance: ");
Serial.println(cm);
if(client.publish(publishTopic,(char*) payload.c_str()))
{
Serial.println("Publish OK");
}
e
l
s
e
{
Serial.println("Publish FAILED");
}
}
float inches = (cm / 2.54); //print on LCD
lcd.setCursor(0,0); lcd.print("Inches");
lcd.setCursor(4,0);  lcd.setCursor(12,0);
lcd.print("cm");  lcd.setCursor(1,1);
lcd.print(inches, 1); lcd.setCursor(11,1);
lcd.print(cm, 1);  lcd.setCursor(14,1);
delay(1000);  lcd.clear();
}
}
if(cm <= 100)

{
digitalWrite(21,HIGH);
```

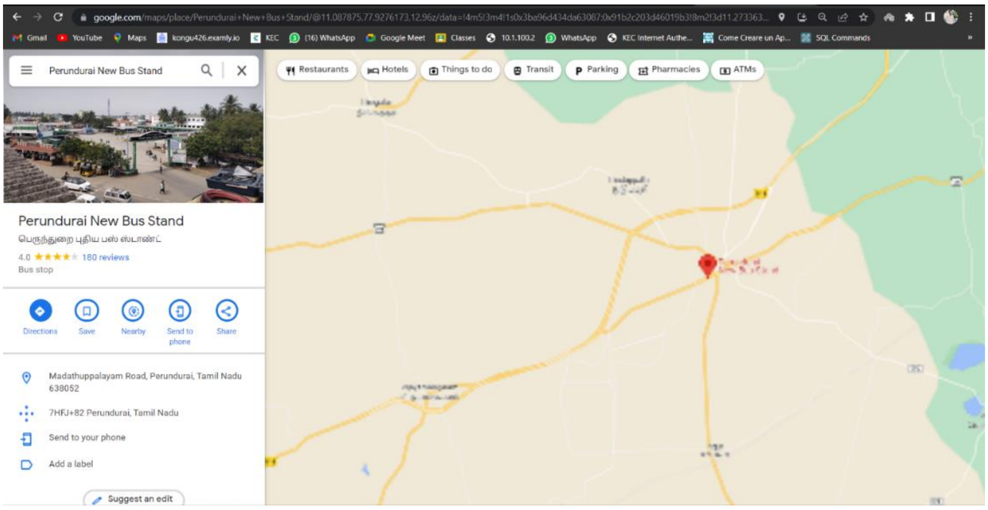
```
String payload = "{\"High Alert!!\":\"";
payload +=
cm; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) // if data is uploaded to cloud
successfully,prints publish ok or prints publish failed
{
Serial.println("Publish OK");
}
}
if(cm <= 250)
{
digitalWrite(22,HIGH);
String payload =
"{\"Warning!!\":\"";
payload += dist; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending distance: ");
Serial.println(cm);
if(client.publish(publishTopic,(char*) payload.c_str()))
{
Serial.println("Publish OK");
}
e
l
s
e
{
Serial.println("Publish FAILED");
}
}
float inches = (cm / 2.54); //print on LCD
lcd.setCursor(0,0); lcd.print("Inches");
lcd.setCursor(4,0);  lcd.setCursor(12,0);
lcd.print("cm");  lcd.setCursor(1,1);
lcd.print(inches, 1); lcd.setCursor(11,1);
lcd.print(cm, 1);  lcd.setCursor(14,1);
delay(1000);  lcd.clear();
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) // if data is uploaded to cloud
successfully,prints publish ok or prints publish failed
{
Serial.println("Publish OK");
}
}
if(cm <= 250)
```

```
{
digitalWrite(22,HIGH);
String payload =
"{\"Warning!!\":\"";
payload += dist; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending distance: ");
Serial.println(cm);
if(client.publish(publishTopic,(char*) payload.c_str()))
{
Serial.println("Publish OK");
}
e
l
s
e
{
Serial.println("Publish FAILED");
}
}
float inches = (cm / 2.54); //print on LCD
lcd.setCursor(0,0); lcd.print("Inches");
lcd.setCursor(4,0);  lcd.setCursor(12,0);
lcd.print("cm");  lcd.setCursor(1,1);
lcd.print(inches, 1); lcd.setCursor(11,1);
lcd.print(cm, 1);  lcd.setCursor(14,1);
delay(1000);  lcd.clear();
}
```

OUTPUT:

Web Application:





Perundurai New Bus Stand

பெருந்துறை புதிய பஸ் ஸ்டாண்ட்

4.0 ★★★★☆ 180 reviews

Bus stop

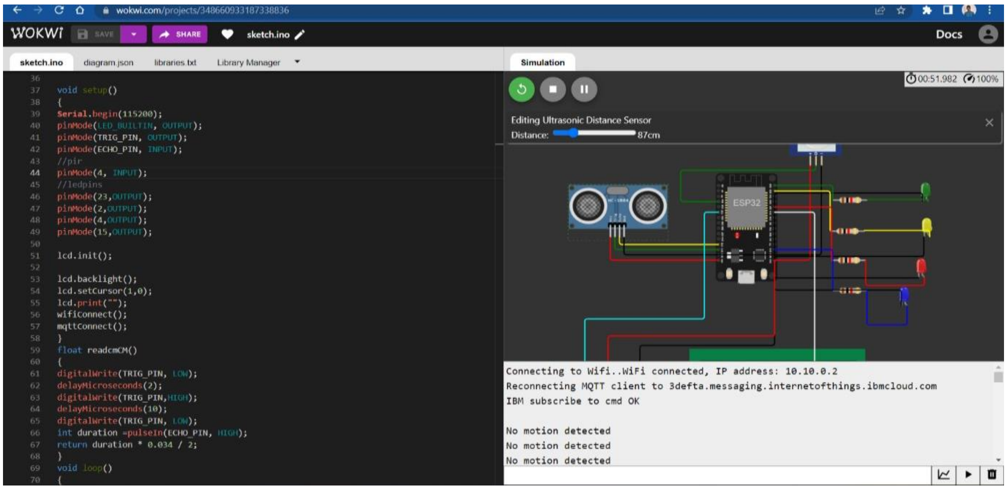
Madathuppalayam Road, Perundurai, Tamil Nadu 638052

7HFJ+82 Perundurai, Tamil Nadu

Send to your phone

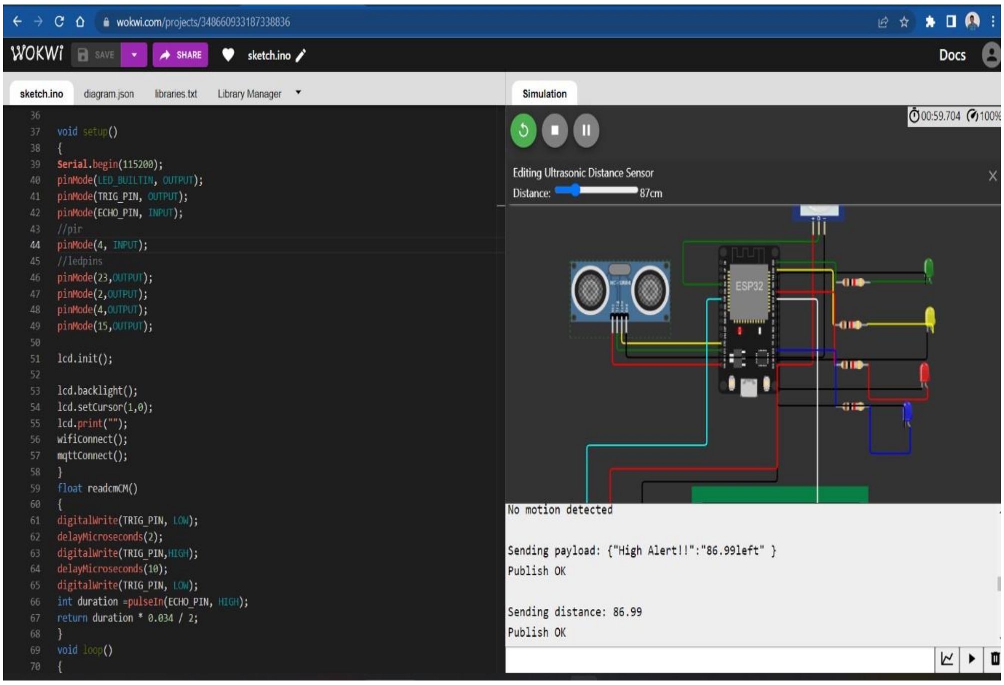
Add a label

Node Red Connection With IBM IoT Platform:



```
void setup()
{
Serial.begin(115200);
pinMode(LED_BUILTIN, OUTPUT);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
//pir
pinMode(4, INPUT);
//ledpins
pinMode(23,OUTPUT);
pinMode(2,OUTPUT);
pinMode(4,OUTPUT);
pinMode(15,OUTPUT);

lcd.init();

lcd.backlight();
lcd.setCursor(1,0);
lcd.print("");
wifiConnect();
mqttConnect();
}
float readcmCM()
{
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN,HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);
int duration =pulseIn(ECHO_PIN, HIGH);
return duration * 0.034 / 2;
}
void loop()
{
```

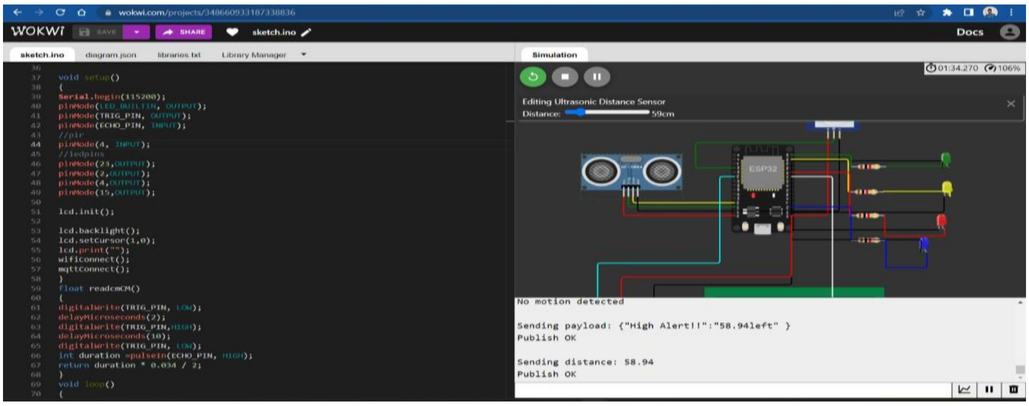
```
Connecting to Wifi..WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to 3defta.messaging.internetofthings.ibmcloud.com
IBM subscribe to cmd OK

No motion detected
No motion detected
No motion detected
```



```
void setup()
{
Serial.begin(115200);
pinMode(LED_BUILTIN, OUTPUT);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
//pir
pinMode(4, INPUT);
//ledpins
pinMode(23,OUTPUT);
pinMode(2,OUTPUT);
pinMode(4,OUTPUT);
pinMode(15,OUTPUT);

lcd.init();

lcd.backlight();
lcd.setCursor(1,0);
lcd.print("");
wifiConnect();
mqttConnect();
}
float readcmCM()
{
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN,HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);
int duration =pulseIn(ECHO_PIN, HIGH);
return duration * 0.034 / 2;
}
void loop()
{
```

Editing Ultrasonic Distance Sensor
Distance: 87cm

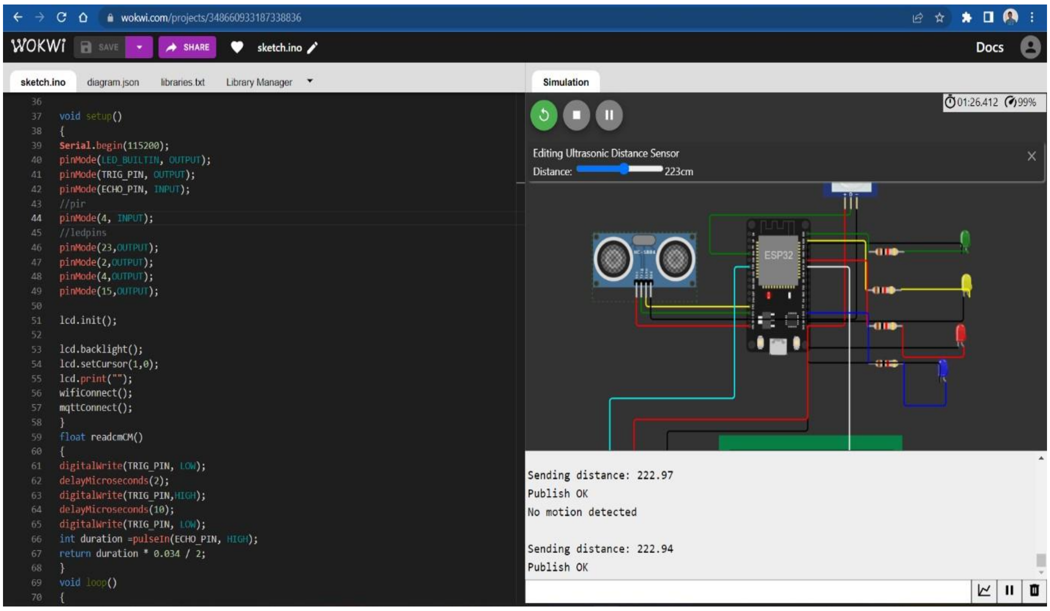
```
No motion detected

Sending payload: {"High Alert!!":"86.99left" }
Publish OK

Sending distance: 86.99
Publish OK
```



```
void setup()
{
Serial.begin(115200);
pinMode(LED_BUILTIN, OUTPUT);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
//pir
pinMode(4, INPUT);
//ledpins
pinMode(23,OUTPUT);
pinMode(2,OUTPUT);
pinMode(4,OUTPUT);
pinMode(15,OUTPUT);

lcd.init();

lcd.backlight();
lcd.setCursor(1,0);
lcd.print("");
wifiConnect();
mqttConnect();
}
float readcmCM()
{
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN,HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);
int duration =pulseIn(ECHO_PIN, HIGH);
return duration * 0.034 / 2;
}
void loop()
{
```

Editing Ultrasonic Distance Sensor
Distance: 59cm

```
No motion detected

Sending payload: {"High Alert!!":"58.94left" }
Publish OK

Sending distance: 58.94
Publish OK
```



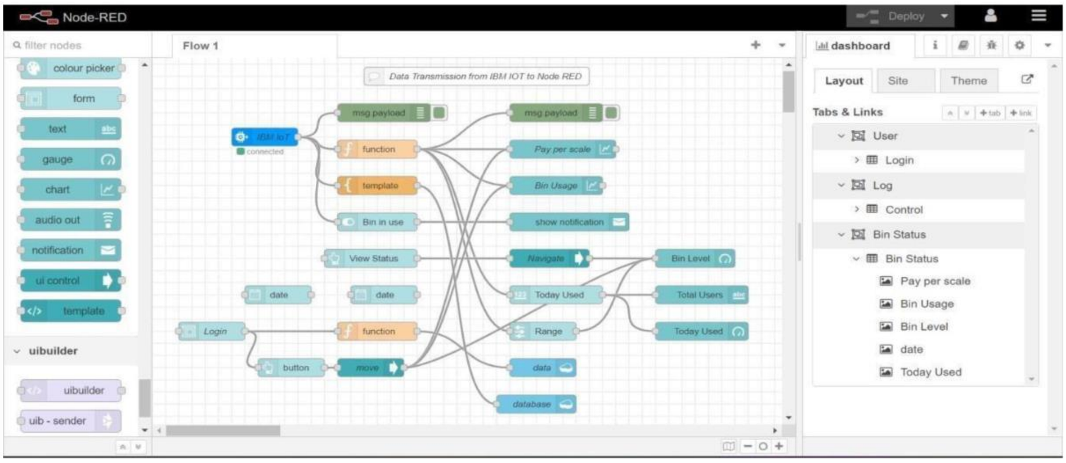
```
void setup()
{
Serial.begin(115200);
pinMode(LED_BUILTIN, OUTPUT);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
//pir
pinMode(4, INPUT);
//ledpins
pinMode(23,OUTPUT);
pinMode(2,OUTPUT);
pinMode(4,OUTPUT);
pinMode(15,OUTPUT);

lcd.init();

lcd.backlight();
lcd.setCursor(1,0);
lcd.print("");
wifiConnect();
mqttConnect();
}
float readcmCM()
{
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN,HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);
int duration =pulseIn(ECHO_PIN, HIGH);
return duration * 0.034 / 2;
}
void loop()
{
```

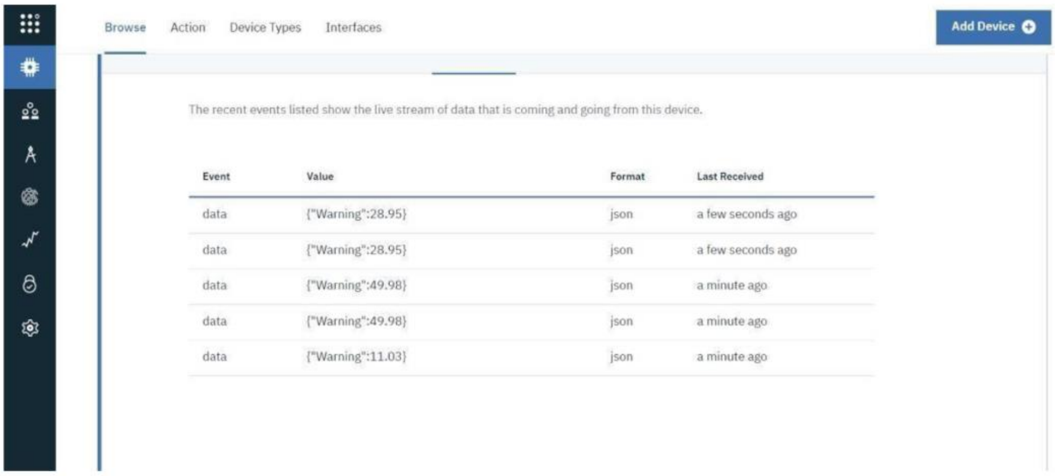
Editing Ultrasonic Distance Sensor
Distance: 223cm

```
Sending distance: 222.97
Publish OK
No motion detected

Sending distance: 222.94
Publish OK
```

Sprint 3:

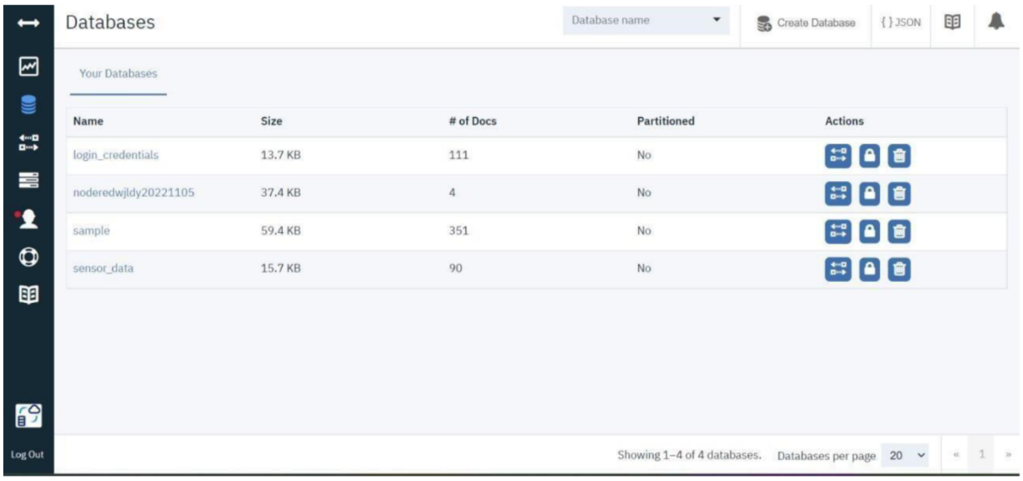
Node Red With Data Transfer from Working Setup to IBM IoT Platform





The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
data	{"Warning":28.95}	json	a few seconds ago
data	{"Warning":28.95}	json	a few seconds ago
data	{"Warning":49.98}	json	a minute ago
data	{"Warning":49.98}	json	a minute ago
data	{"Warning":11.03}	json	a minute ago

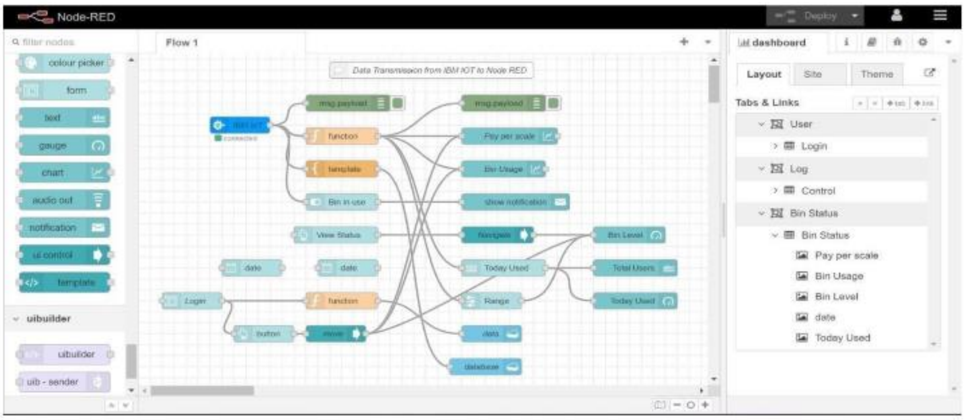


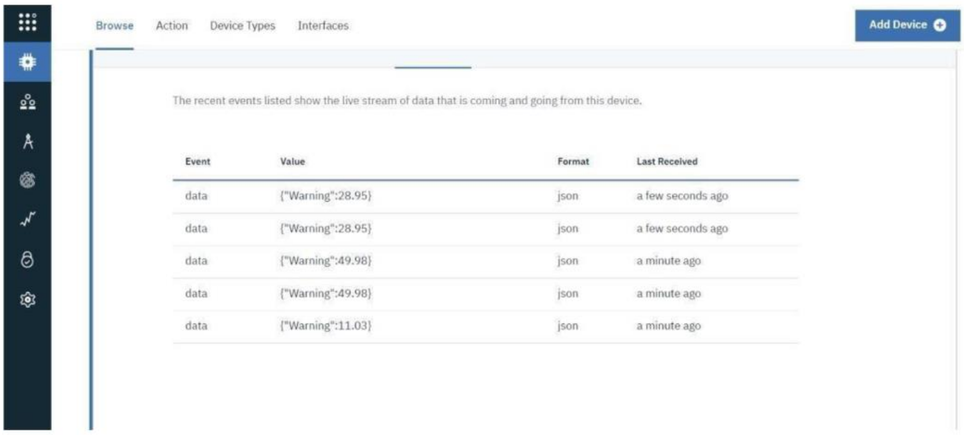
Name	Size	# of Docs	Partitioned	Actions
login_credentials	13.7 KB	111	No	
noderedwjldy20221105	37.4 KB	4	No	
sample	59.4 KB	351	No	
sensor_data	15.7 KB	90	No	

Showing 1–4 of 4 databases. Databases per page 20

Sprint 4:

Node red data transfer to IBM IoT with bin level





The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
data	{"Warning":28.95}	json	a few seconds ago
data	{"Warning":28.95}	json	a few seconds ago
data	{"Warning":49.98}	json	a minute ago
data	{"Warning":49.98}	json	a minute ago
data	{"Warning":11.03}	json	a minute ago

CHAPTER 8: TESTING

8.1 Test cases:

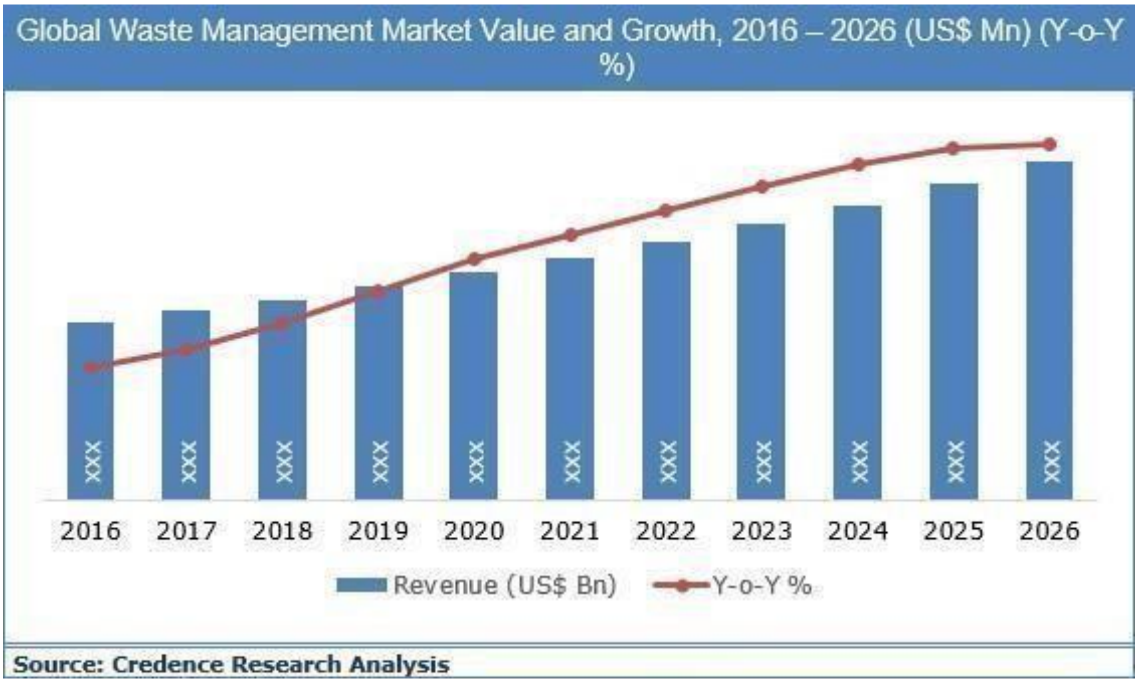
Test case no.	Sensor/Stage	Input	Expected output	Obtained output	Status
1.	Ultrasonic	Garbage level in bin i)Null ii)Full iii)Range in %	Correct level or distance	As expected	Pass
2.	ESP – 32	Microcontroller to process the input data	To collect the data from sensor	As expected	Pass
3.	Load cell	To measure mechanical force	Calculate the force due to the bin weight	As expected	Pass
4.	Gauge	To display the tares	Display the level for collection	As expected	Pass
5.	HX710	Weight of the bin (in kg)	Measure the weight	As expected	Pass

8.2 User Acceptance testing

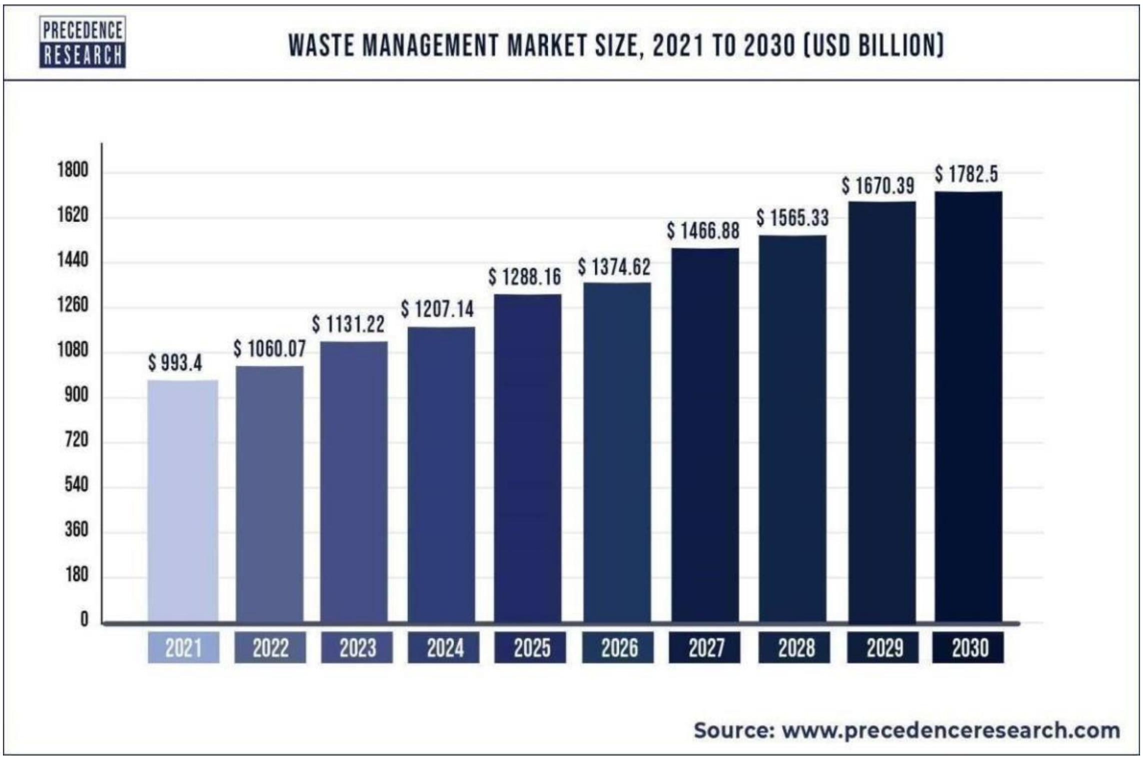
Acceptance testing - is the final phase of product testing prior to public launch. A level of the software testing process where a system is tested for acceptability. The purpose of this test is to evaluate the system's compliance with the business requirements and assess whether it is acceptable for delivery.

CHAPTER 9: RESULTS

9.1 Performance Metrics:



Source: Credence Research Analysis



Source: www.precedenceresearch.com

CHAPTER 10: ADVANTAGES AND DISADVANTAGES

10.1 Advantages:

- Intelligent compaction of waste by monitoring fill level in real-time using sensors.
- Reduces manpower requirement to handle the garbage collection
- Emphasizes of healthy environment and keep the cities cleaner and more beautiful.
- It reduces infrastructure, operating and maintenance costs by upto 30%.
- Increases recycling rate of waste.

10.2 Disadvantages:

- Initial large-scale implementation takes cost.
- System requires more number waste bins for separate waste collection.
- Wireless technologies used should have proper connections as they have shorter range and lower data speed
- Training programs should be provided to people involving in the ecosystem of smart waste management.
- Sensors may encounter damage so it should be kept under protective ambience to prevent the damage.

CHAPTER 11: CONCLUSION

Improper disposal and improper maintenance of domestic waste create issues in public health and environment pollution thus this paper attempts to provide practical solution towards managing the waste collaborating it with the use of IOT. by using the smart waste management system, we can manage waste properly we are also able to sort the Bio-degradable and non-Biodegradable waste properly which reduces the pollution in the environment. Various waste management initiatives taken for human well-being and to improve the TWM practices were broadly discussed in this chapter. The parameters that influence the technology and economic aspects of waste management were also discussed clearly. Different types of barriers in TWM, such as economic hitches, political issues, legislative disputes, informative and managerial as well as solutions and success factors for implementing an effective management of toxic organic waste within a globular context, were also discussed giving some real examples. The effect of urbanization on the environmental degradation and economic growth was also discussed. The proposed system will help to overcome all the serious issues related to waste and keep the environment clean.

CHAPTER 12: FUTURE WORK

Based on the real-time and historical data collected and stored in the cloud waste collection schedules and routes can be optimized. Predictive analytics could be used to make decisions ahead of time and offers insight into waste bin locations. Graph theory optimization algorithms can be used to manage waste collection strategies dynamically and efficiently. Every day, the workers can receive the newly calculated routes in their navigation devices. The system can be designed to learn from experience and to make decisions not only on the daily waste level status but also on future state forecast, traffic congestion, balanced cost-efficiency functions, and other affecting factors that a priori humans cannot foresee.

Garbage collectors could access the application on their mobile phone/tablets using the internet. Real-time GPS assistance can be used to direct them to the pre-decided route. As they go collecting the garbage from the containers, the management is also aware of the progress as the vehicle, as well as the garbage containers, are traced in real-time. The management staff gets their own personalized administration panel over a computer/tablet which gives them a bird eye view over the entire operations.

An alternative solution using image processing and camera as a passive sensor could be used. But, the cost of those image processing cameras is higher as compared to the ultrasonic sensors, which leads to high solution implementation cost.

CHAPTER 13: APPENDIX

13.1 Source Code:

Web Application to get the Live location:

Index.html

```
<!DOCTYPE html>
<html>

<head>
  <link rel="stylesheet"
href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
integrity="sha384-
ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQUOhcWr7x9JvoRxT2MZw1T"
crossorigin="anonymous">
<meta charset="utf-8">
<meta name="viewport" content="width=device-width">
<title>Waste Management</title>
<link rel="icon" type="image/x-icon" href="/IMAGES/DUMPSTER.png">
<link href="style.css" rel="stylesheet" type="text/css" />
<script src="https://www.gstatic.com/firebasejs/8.10.1/firebase-app.js"></script>
<script src="https://www.gstatic.com/firebasejs/8.10.1/firebase-db.js"></script>
<script type="module">
    // Import the functions you need from the SDKs you need
    import { initializeApp } from "https://www.gstatic.com/firebasejs/9.14.0/firebase-
app.js";
    import { getAnalytics } from "https://www.gstatic.com/firebasejs/9.14.0/firebase-
analytics.js";
    // TODO: Add SDKs for Firebase products that you want to use
    // https://firebase.google.com/docs/web/setup#available-libraries

    // Your web app's Firebase configuration
    // For Firebase JS SDK v7.20.0 and later, measurementId is optional
    const firebaseConfig = {
      apiKey: "AIzaSyCLmn-TzMoUVe9sBa6h56Bd4WnFJtjm0aE",
      authDomain: "ibm-smart-waste.firebaseapp.com",
      databaseURL: "https://ibm-smart-waste-default-rtdb.firebaseio.com",
      projectId: "ibm-smart-waste",
      storageBucket: "ibm-smart-waste.appspot.com",
      messagingSenderId: "426276430128",
      appId: "1:426276430128:web:4f8671bf97c4c9450728f5",
      measurementId: "G-16DJ7XEDK5"
    };

    // Initialize Firebase
    const firebase = initializeApp(firebaseConfig);
    const analytics = getAnalytics(firebase);
  </script>
<script defer src="db.js"></script>
</head>

<body style="background-color:#1F1B24;">
<script src="map.js"></script>



    <div id="map_container">
            <h1 id="live_location_heading" >LOCATION</h1>
            <div id="map"></div>
            <div id="alert_msg">ALERT MESSAGE!</div>
    </div>
</div>
<center><a
href="https://www.google.com/maps/place/Perundurai+New+Bus+Stand/@11.273363,7
7.5778373,17z/data=!3m1!4b1!4m5!3m4!1s0x3ba96d434da63087:0x91b2c203d46019b
3!8m2!3d11.273363!4d77.5800313"
  type="button" class="btn btn-dark">Get Dumpster</a></center>

<script
    src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBBLyWj-
3FWtCbCXGW3ysEiI2fDfrv2v0Q&callback=myMap"></script></div>
</body>

</html>
```

Style.css:

```
html, body
{
    height: 100%;
    margin: 0px;
    padding:0px;
}
#container
{
display: flex;
flex-direction: row;
height: 100%;
width: 100%;
position: relative;
}
#logo_container
{
height: 100%;
width: 12%;
background-color: #C5C6D0;
display: flex;
flex-direction: column;
vertical-align: text-bottom;
}
.logo
{
width:70%;
margin: 5% 15%;

/*  border-radius: 50%; */
}
#logo_3
{
vertical-align: text-bottom;

}
#data_container
{
height: 100%;
width: 20%;
margin-left: 1%;
```

```
margin-right: 1%;
display: flex;
flex-direction: column;
}
#data_status
{
height:60%;
width:8%;
margin:7%;
background-color: #691F6E;
display: flex;
flex-direction: column;
border-radius:20px;
}
#load_status
{
background-image: url("/Images/KG.png");
background-repeat: no-repeat;
 background-size:     170px;
background-position: left center;
}
#cap_status
{
background-image: url("/Images/dust.png");
background-repeat: no-repeat;
 background-size: 150px;
background-position: left center;
}
.status
{
width: 80%;
height: 40%;
margin:5% 10%;
background-color:#185adc;
border-radius:20px;
display: flex;
    justify-content: center;
    align-items: center;
color: white;
font-size: 60px;
```

```
}
.datas
{
width:86%;
margin:2.5% 7%;
height:10%;
   background: url(water.png);
    background-repeat: repeat-x;
    animation: datas 10s linear infinite;

    box-shadow: 0 0 0 6px #98d7eb, 0  20px 35px rgba(0,0,0,1);

}
#map_container
{
height: 100%;
width: 100%;
display: flex;
flex-direction: column;
}
#live_location_heading
{
margin-top:10%;
text-align: center;
  color:  GREY;
}
#map
{
height: 70%;
width: 90%;
margin-left: 4%;
margin-right:4%;
border: 10px solid white;
border-radius: 25px;
}
#alert_msg
{
width:92%;
height:20%;
margin:4%;
background-color:grey;
border-radius: 20px;
```

```
display: flex;
    justify-content: center;
    align-items: center;
color: #41af7f;
font-size: 25px;
font-weight: bold;
}
.lat
{
margin: 0px;
font-size:0px;
}




@keyframes datas{
   0%
   {
      background-position: -500px 100px;
   }
   40%
   {
      background-position: 1000px -10px;
   }

   80% {
      background-position: 2000px 40px;
   }
   100% {
      background-position: 2700px 95px;
   }

}
```

Map.js:

```
const database = firebase.database();

function myMap()
{
var ref1 = firebase.database().ref();

    ref1.on("value", function(snapshot)
     {
       snapshot.forEach(function (childSnapshot) {
           var value = childSnapshot.val();
                          const latitude = value.latitude;
                          const longitude = value.longitude;

                          var latlong = { lat: latitude, lng: longitude}
                          var mapProp =
                          {
                                   center: new google.maps.LatLng(latlong),
                                   zoom: 10,
                          };
                          var map = new google.maps.Map(document.getElementById("map"),
mapProp);

                          var marker = new google.maps.Marker({ position: latlong });
                          marker.setMap(map);
        });
     }, function (error) {
       console.log("Error: " + error.code);
     });

}
```

Node Red Connection With IBM IoT Platform:

```
#include <WiFi.h> //library for wifi
#include <PubSubClient.h> //library for MQTT
#include <LiquidCrystal_I2C.h>

LiquidCrystal_I2C lcd(0x27, 20, 4);
// credentials of IBM Accounts -
#define ORG "3defta" //IBM organisation id
#define DEVICE_TYPE "hariprasath" // Device type mentioned in ibm watson iot platform
#define DEVICE_ID "12345" // Device ID mentioned in ibm watson iot platform
```

```
#define TOKEN "CpL-H1C-Pt4i9iM-F5" // Token
// customise above values -

char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; // server
name char publishTopic[] = "iot-2/evt/data/fmt/json";
char topic[] = "iot-2/cmd/led/fmt/String"; // cmd Represent type and command is test format
of strings char authMethod[] = "use-token-auth"; // authentication method  char token[] =
TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //Client id
//
WiFiClient wifiClient; // creating instance for wificlient
PubSubClient client(server, 1883, wifiClient);
#define ECHO_PIN 12
#define TRIG_PIN 13
float dist;

void setup()
{
Serial.begin(115200);
pinMode(LED_BUILTIN, OUTPUT);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
//pir
pinMode(4, INPUT);
//ledpins
pinMode(23,OUTPUT);
pinMode(2,OUTPUT);
pinMode(4,OUTPUT);
pinMode(15,OUTPUT); lcd.init();

lcd.backl
ight();
lcd.setC
ursor(1,0
);
lcd.print
("");
wifiCon
nect();
mqttCon
nect(); }
float
readcmC
M()
{
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN,HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);
int duration
```

```
=pulseIn(ECHO_PIN, HIGH);
return duration * 0.034 / 2;
} void
loop()
{
lcd.cle
ar();
publish
Data();
delay(
500);
if
(!client
.loop()
) {
mqttConnect(); //function call to connect to IBM
}
}
/* -retrieving to cloud */ void wifiConnect()
{
Serial.print("Connecting to ");
Serial.print("Wifi");
 WiFi.begin("Wokwi-GUEST", "", 6);
while (WiFi.status() != WL_CONNECTED)
{
del
ay(
500
);
Serial.print(".");
}
Serial.print("WiFi connected, IP address: ");
Serial.println(WiFi.localIP());
} void
mqttConn
ect()
{
if (!client.connected())
{
Serial.print("Reconnecting MQTT client to ");
Serial.println(server);
while(!client.connect(clientId, authMethod, token))
{
Serial.print("."); delay(500);
}
initManagedDevice(); Serial.println();
}
}
void initManagedDevice()
{
```

```
if (client.subscribe(topic))
{
Serial.println("IBM subscribe to cmd OK");
}
e
l
s
e
{
Serial.println("subscribe to cmd FAILED");
}
}
void publishData()
{
float cm = readcmCM();
if(digitalRead(34)) //PIR motion detection
{
Serial.println("Motion Detected"); Serial.println("Lid Opened"); digitalWrite(15, HIGH);
}
e
l
s
e
{
digitalWrite(15, LOW);
}
if(digitalRead(34)== true)
{
if(cm <= 100) //Bin level detection
{
digitalWrite(2, HIGH);
Serial.println("High Alert!!!,Trash bin is about to be full");
Serial.println("Lid Closed");
lcd.print("Full! Don't use");
delay(2000);
lcd.clear();
digitalWrite(4,
LOW);
digitalWrite(23,
LOW);
}
else if(cm > 150 && cm < 250)
{
digitalWrite(4, HIGH);
Serial.println("Warning!!,Trash is about to cross 50% of
bin level");  digitalWrite(2,LOW); digitalWrite(23,
LOW);
}
else if(cm > 250 && cm <=400)
{
```

```
digitalWrite(23, HIGH);
Serial.println("Bin is
available");
digitalWrite(2,LOW);
digitalWrite(4, LOW);
}
delay(10000);
Serial.println("Lid Closed");
}
if(cm <= 100)

{
digitalWrite(21,HIGH);
String payload = "{\"High Alert!!\":\"";
payload +=
cm; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) // if data is uploaded to cloud
successfully,prints publish ok or prints publish failed
{
Serial.println("Publish OK");
}
}
if(cm <= 250)
{
digitalWrite(22,HIGH);
String payload =
"{\"Warning!!\":\"";
payload += dist; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending distance: ");
Serial.println(cm);
if(client.publish(publishTopic,(char*) payload.c_str()))
{
Serial.println("Publish OK");
}
e
l
s
e
{
Serial.println("Publish FAILED");
}
}
float inches = (cm / 2.54); //print on LCD
lcd.setCursor(0,0); lcd.print("Inches");
lcd.setCursor(4,0);  lcd.setCursor(12,0);
lcd.print("cm");  lcd.setCursor(1,1);
lcd.print(inches, 1); lcd.setCursor(11,1);
lcd.print(cm, 1);  lcd.setCursor(14,1);
delay(1000);  lcd.clear();
}
}
if(cm <= 100)

{
digitalWrite(21,HIGH);
String payload = "{\"High Alert!!\":\"";
payload +=
cm; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) // if data is uploaded to cloud
successfully,prints publish ok or prints publish failed
{
Serial.println("Publish OK");
}
}
if(cm <= 250)
{
digitalWrite(22,HIGH);
String payload =
"{\"Warning!!\":\"";
payload += dist; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending distance: ");
Serial.println(cm);
if(client.publish(publishTopic,(char*) payload.c_str()))
{
Serial.println("Publish OK");
}
e
l
s
e
{
Serial.println("Publish FAILED");
}
}
float inches = (cm / 2.54); //print on LCD
lcd.setCursor(0,0); lcd.print("Inches");
lcd.setCursor(4,0);  lcd.setCursor(12,0);
lcd.print("cm");  lcd.setCursor(1,1);
lcd.print(inches, 1); lcd.setCursor(11,1);
lcd.print(cm, 1);  lcd.setCursor(14,1);
delay(1000);  lcd.clear();
Serial.print("Sending payload: ");
Serial.println(payload);
if (client.publish(publishTopic, (char*) payload.c_str())) // if data is uploaded to cloud
successfully,prints publish ok or prints publish failed
{
Serial.println("Publish OK");
}
}
if(cm <= 250)
{
digitalWrite(22,HIGH);
String payload =
"{\"Warning!!\":\"";
payload += dist; payload
+= "left\" }";
Serial.print("\n");
Serial.print("Sending distance: ");
Serial.println(cm);
if(client.publish(publishTopic,(char*) payload.c_str()))
{
Serial.println("Publish OK");
}
e
l
s
e
{
Serial.println("Publish FAILED");
}
}
float inches = (cm / 2.54); //print on LCD
lcd.setCursor(0,0); lcd.print("Inches");
lcd.setCursor(4,0);  lcd.setCursor(12,0);
lcd.print("cm");  lcd.setCursor(1,1);
lcd.print(inches, 1); lcd.setCursor(11,1);
lcd.print(cm, 1);  lcd.setCursor(14,1);
delay(1000);  lcd.clear();
}
```

13.2 Github Link:

https://github.com/IBM-EPBL/IBM-Project-34208-1660232880

13.3 Project Demo Link: