Report on Smart Waste Management System for Metropolitan Cities

1.INTRODUCTION:

1.1 PROJECT OVERVIEW:-

*The "***Smart Waste Management System for Metropolitan Cities***"* is a GPS based one. The suggested device and implementation will track waste storage and monitoring.With the growth of the world's population and the gradual relocation of a large number of people to cities, the concept of Metropolitan cities is becoming ever more relevant. A city is a concept that entails integrating a range of information and communication technologies, such as the Internet of Things (IoT), to manage public space and city services in a sustainable manner. The bins which are going to be used in our system are smart. They are created by using sensors which are helpful to share the information about the waste bin.and The sensors which we used in our bins will monitor the waste each time after the waste disposal if the bin is going to be filled means it will send an alert message to the administrator based on the bin details and the nearby bin data administrator will arrange the vehicles to collect the waste.In android application we can monitor the waste based on the location it will be available in longitude and latitude way based on that admin can understand the location and arrange the vehicle for collection of waste.maily we will be converting the bins into smart bins by using IOT and weight measuring devices and whenever the bins was filled by 80% the alert message will be sent to node-red and from node-red .next the data will be fetched by api will be displayed in ui and also using credentials in mit app data will be viewed and if load exceed more than 80% data will be displayed for next 30 sec and will be shifted to next bin. Once the admin server received the message it forwarded the message to the worker in charge, if the worker was available, he would notify his/her presence by accepting the work and would reach the required destination. If the worker was not available, the work would be transferred to another worker in this way it will be processed further.

1.2 PURPOSE:-

Day by day the population is rapidly growing and the economy is broadening in our country and also there is a very vast growth of waste being generated. There is no actual right way of its solution or proper chain system to track and monitor the waste and disposal system. And cities are getting smart nowadays, but waste is not. Regardless of all the cities, the dustbins and waste are not getting tracked, sometimes the garbage in the bins gets to above the point, where it blemishes outside the garbage pail and opens out in whole areas and causes so many health issues to the citizens. In this work, the prototype schema which we are trying to address the waste management issues with several solutions like by using the smart bins which will indicate the level of the garbage inside the bins and will alert the admin to pick the garbage from the particular region. Next, as it is a smart waste management system, we are giving some approach to society. People can also trail the waste in its particular society or close by it. And regardless of the garbage collector not attending to the particular society or particular area, the society member can record the issue through the user app, and that can be reached directly to the admin. The motive of making this prototype is to put one step into the solution of waste management. Cleanliness and Hygiene. 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. In order to deal with these problems this system is an ideology put forward.

2.LITERATURE SURVEY:

This is not an original idea, IOT based dustbin was implemented and effectuated much before. Waste management is one of the complicated challenges in cities, the system which is currently used in cities is not much more efficient, we continue to use an old and outmoded paradigm that no longer serves the entail of municipalities, Still find over spilled waste containers giving off irritating smells causing serious health issues and atmosphere impairment.

2.1 EXISTING SYSTEM:-

The main problems of the existing solid waste collection process and management system are as follows:

- More complications in the processing
- many controlling units linked with each other
- higher implementation cost
- Manual systems in which employees clear the dumpsters periodically
- No systematic approach towards clearing the dumpsters
- Unclear about the status of a particular location
- Employees are unaware of the need for a particular location
- Very less effective in cleaning city

2.2 REFERENCE:

1. Shyam, Gopal Kirshna, Sunilkumar S. Manvi, and Priyanka Bharti. "Smart waste management using Internet-of-Things (IoT)." IEEE Computing and Communications Technologies (ICCCT), (2017) pp. 199-203.

2. Kurre, Vishesh Kumar. "Smart Garbage Collection Bin overflows Indicator using IOT." International Research Journal of Engineering and Technology (IRJET) (2016)

2.3 Problem Statement Definition:-

This project deals with the problem of waste management in smart cities, where the garbage collection system is not optimized. This project enables the organizations to meet their needs of smart garbage management systems. This system allows the user 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

The IoT-Based waste management system helps people with proper waste management. It will notify when the waste is going to fill and the metro corporation people will come and collect the waste this helps to make the city clean. By using this system we can manage the waste properly and the citizens can feel free to dispose their waste in a proper way this will make the surroundings clean and safe

3.DEATION & PROPOSED SOLUTION:

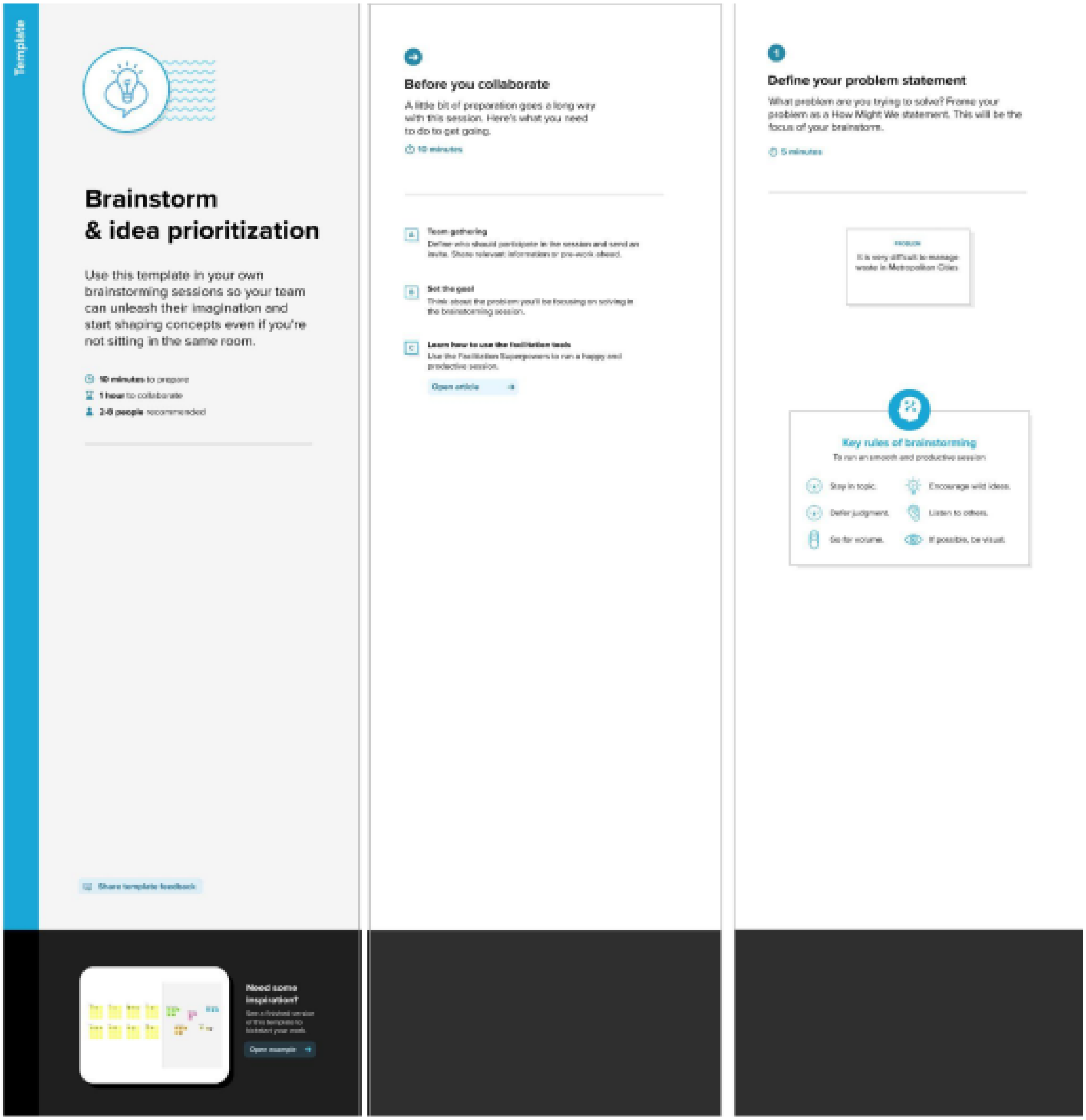
3.1 Empathy Map Canvas:-

An empathy map canvas helps brands provide a better experience for users by helping teams understand the perspectives and mindset of their customers. Using a template to create an empathy map canvas reduces the preparation time and standardizes the process so you create empathy map canvases of similar quality.

3.2 Ideation & Brainstorming:-

Brainstorming provides a free and open environment that encourages everyone within a team to participate in the creative thinking process that leads to problem solving. Prioritizing volume over value, out-of-the-box ideas are welcome and built upon, and all participants are encouraged to collaborate, helping each other develop a rich amount of creative solutions. Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.

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



Brainstorm
& idea prioritization

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

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

Share template feedback

Need some inspiration?
See a finished version of this template to kickstart your work.
Open example →

Before you collaborate

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

10 minutes

A **Team gathering**
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.

B **Set the goal**
Think about the problem you'll be focusing on solving in the brainstorming session.

C **Learn how to use the facilitation tools**
Use the Facilitation Superpowers to run a happy and productive session.

Open article →

1

Define your problem statement

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

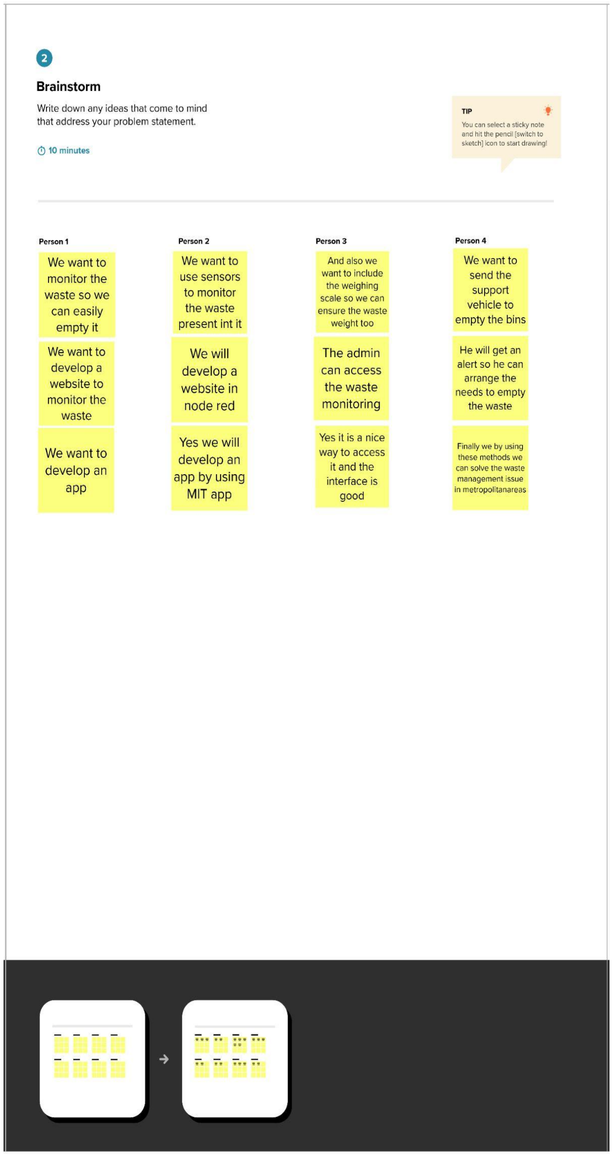
5 minutes

PROBLEM
It is very difficult to manage waste in Metropolitan Cities

Key rules of brainstorming
To run an smooth and productive session

Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.

Step-2: Brainstorm, Idea Listing and Grouping .



2

Brainstorm

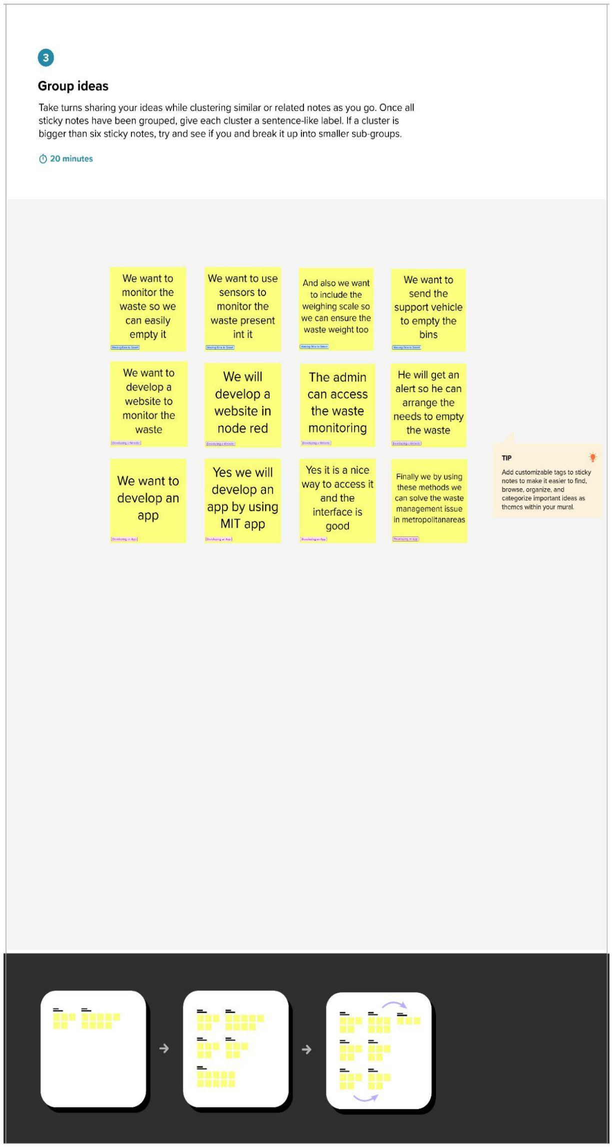
Write down any ideas that come to mind that address your problem statement.

10 minutes

TIP

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

Person 1	Person 2	Person 3	Person 4
We want to monitor the waste so we can easily empty it	We want to use sensors to monitor the waste present int it	And also we want to include the weighing scale so we can ensure the waste weight too	We want to send the support vehicle to empty the bins
We want to develop a website to monitor the waste	We will develop a website in node red	The admin can access the waste monitoring	He will get an alert so he can arrange the needs to empty the waste
We want to develop an app	Yes we will develop an app by using MIT app	Yes it is a nice way to access it and the interface is good	Finally we by using these methods we can solve the waste management issue in metropolitanareas



3

Group ideas

Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.

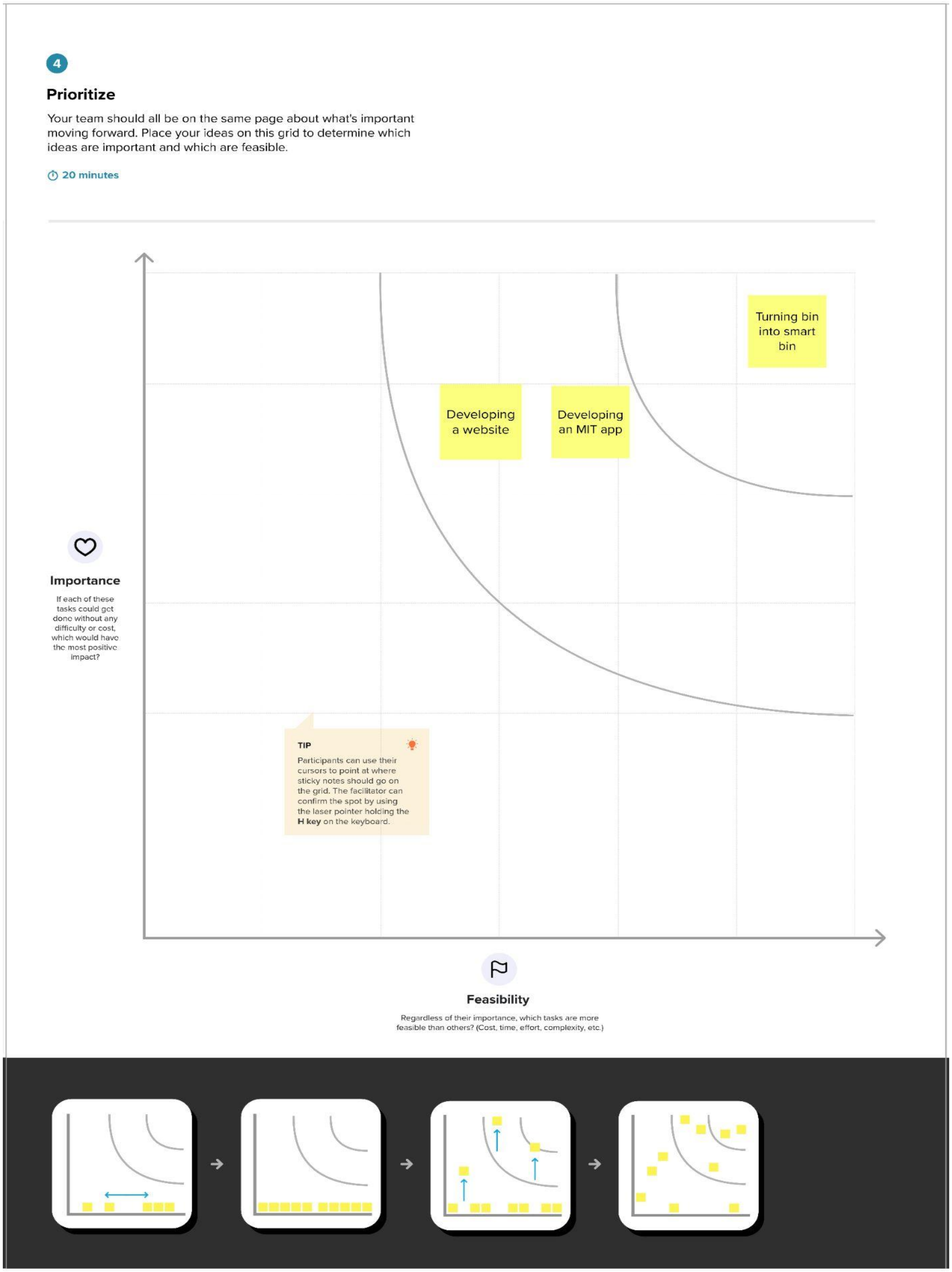
20 minutes

We want to monitor the waste so we can easily empty it	We want to use sensors to monitor the waste present int it	And also we want to include the weighing scale so we can ensure the waste weight too	We want to send the support vehicle to empty the bins
We want to develop a website to monitor the waste	We will develop a website in node red	The admin can access the waste monitoring	He will get an alert so he can arrange the needs to empty the waste
We want to develop an app	Yes we will develop an app by using MIT app	Yes it is a nice way to access it and the interface is good	Finally we by using these methods we can solve the waste management issue in metropolitanareas

TIP

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

Step-3: Idea Prioritization.



4

Prioritize

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

20 minutes

Importance

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

TIP

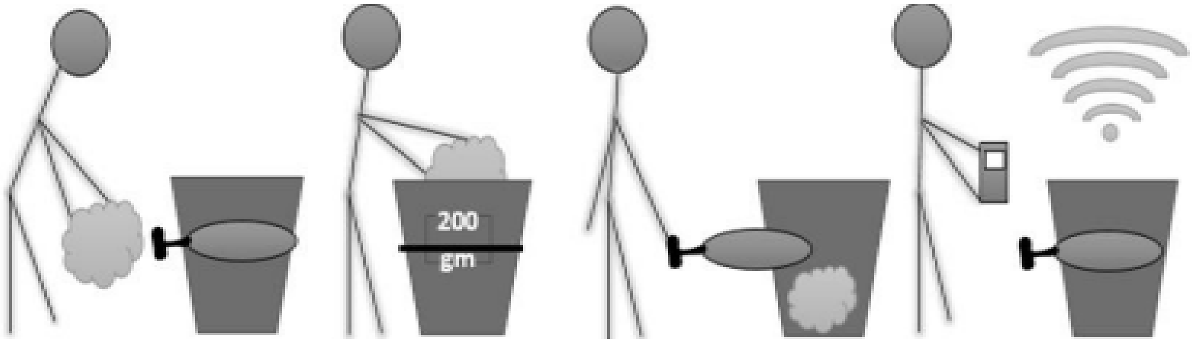
Participants can use their cursors to point at where sticky notes should go on the grid. The facilitator can confirm the spot by using the laser pointer holding the **H key** on the keyboard.

Feasibility

Regardless of their importance, which tasks are more feasible than others? (Cost, time, effort, complexity, etc.)

3.3 Proposed Solution:-

We are using IoT, Android Application, Web Application and IBM Cloud to create a solution for the problem. whenever the bins was filled by 80% the alert message will be sent to node-red and from node-red .next the data will be fetched by api will be displayed in ui and also using credentials in mit app data will be viewed and if load exceed more than 80% data will be displayed for next 30 sec and will be shifted to next bin. Once the admin server received the message it forwarded the message to the worker in charge, if the worker was available, he would notify his/her presence by accepting the work and would reach the required destination. If the worker was not available, the work would be transferred to another worker in this way it will be processed further



1.Problem Statement (Problem to be solved):-

IoT-based systems help the people to manage monitoring the solid waste present in various regions and make it to be easily manageable by using some sensors to reduce man work.

2.Idea / Solution description :-

People can monitor all the sensor parameters easily from a remote location and whenever the solid waste is full it will send an alert to the waste collection people.

3.Novelty / Uniqueness : -

By using this they can reduce lot work and they can manage waste disposal easily

4.Social Impact / Customer Satisfaction :-

n By using this advanced technology, people can manage their waste disposal in an easy way without having any communication with the disposal people.

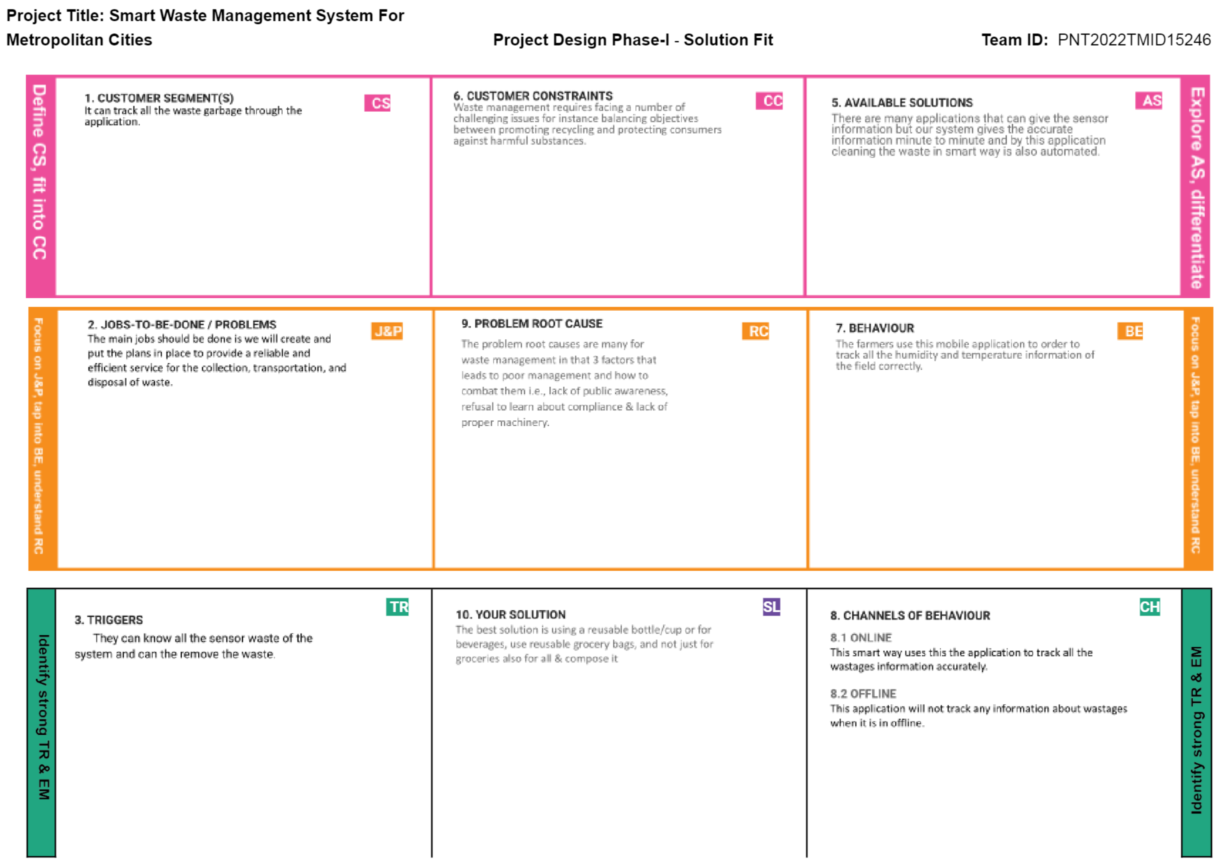
5.Business Model (Revenue Model) :-

As per business model it reduces the man work and provides better output with less machinery.

6.Scalability of the Solution :-

Scalability is another requirement that should be considered while designing a solid waste management system. Scalability in smart this system will gives a great output as compared to the manual system

3.4 Problem Solution fit:-



Project Title: Smart Waste Management System For Metropolitan Cities **Project Design Phase-I** - **Solution Fit** **Team ID:** PNT2022TMID15246

Define CS, fit into CC
Explore AS, differentiate
Focus on J&P, tap into BE, understand RC
Focus on J&P, tap into BE, understand RC
Identify strong TR & EM
Identify strong TR & EM

1. CUSTOMER SEGMENT(S) CS
It can track all the waste garbage through the application.

6. CUSTOMER CONSTRAINTS CC
Waste management requires facing a number of challenging issues for instance balancing objectives between promoting recycling and protecting consumers against harmful substances.

5. AVAILABLE SOLUTIONS AS
There are many applications that can give the sensor information but our system gives the accurate information minute to minute and by this application cleaning the waste in smart way is also automated.

2. JOBS-TO-BE-DONE / PROBLEMS J&P
The main jobs should be done is we will create and put the plans in place to provide a reliable and efficient service for the collection, transportation, and disposal of waste.

9. PROBLEM ROOT CAUSE RC
The problem root causes are many for waste management in that 3 factors that leads to poor management and how to combat them i.e., lack of public awareness, refusal to learn about compliance & lack of proper machinery.

7. BEHAVIOUR BE
The farmers use this mobile application to order to track all the humidity and temperature information of the field correctly.

3. TRIGGERS TR
They can know all the sensor waste of the system and can the remove the waste.

10. YOUR SOLUTION SL
The best solution is using a reusable bottle/cup or for beverages, use reusable grocery bags, and not just for groceries also for all & compose it

8. CHANNELS OF BEHAVIOUR CH

8.1 ONLINE
This smart way uses this the application to track all the wastages information accurately.

8.2 OFFLINE
This application will not track any information about wastages when it is in offline.

4. REQUIREMENT ANALYSIS:-

4.1 Functional requirement:-

The functional requirements of smart waste management system for metropolitan cities are ultrasonic sensors weight transducer sensor

4.2 Non-Functional requirements:-

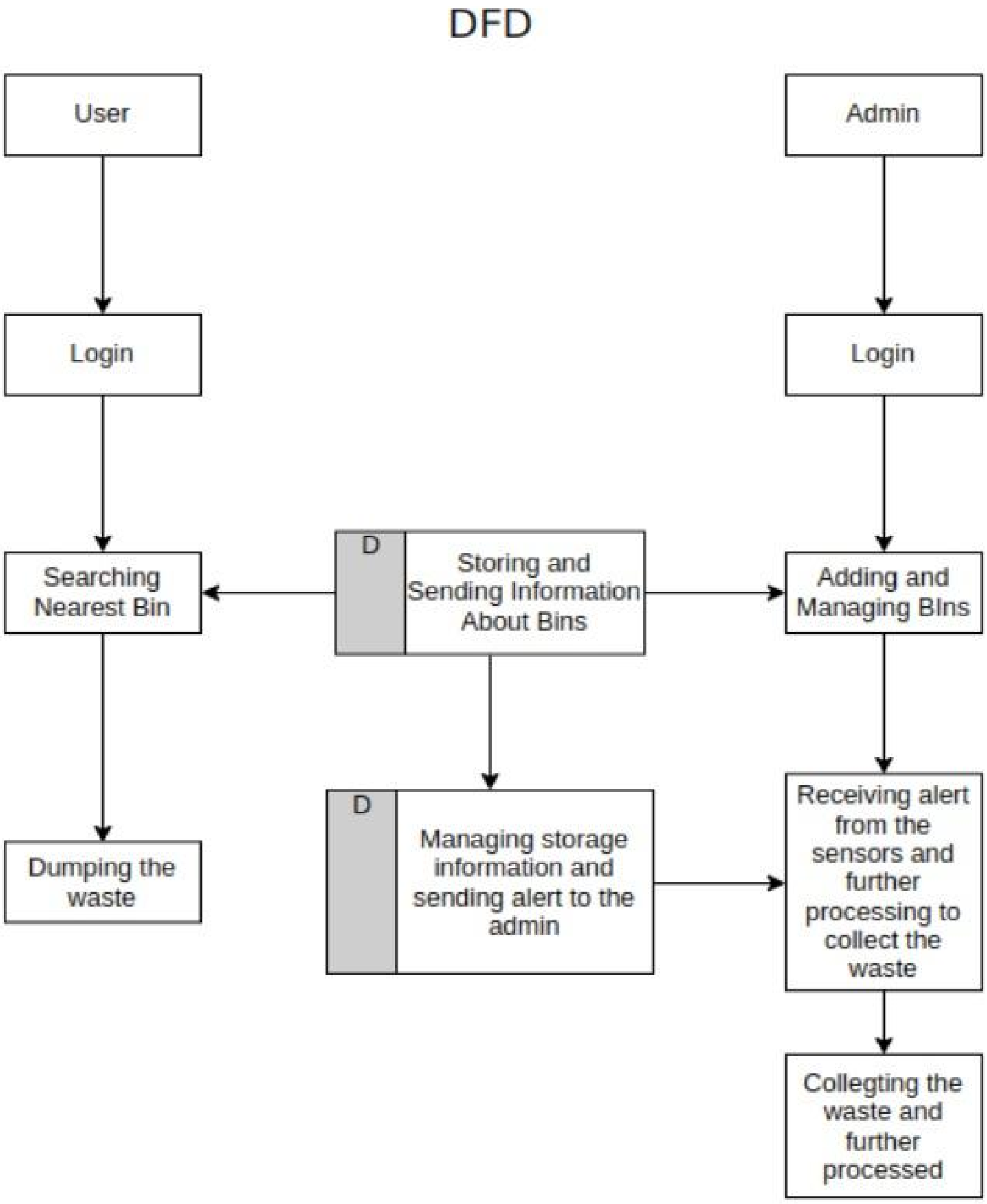
The non-functional requirements of smart waste management system for metropolitan cities are user interface for monitoring and manually intervening (if required) in the efficient and timely collection of garbage from the bins and location accessing from web.

5. PROJECT DESIGN

5.1 Data Flow Diagrams:-

A data flow diagram (DFD) is a graphical or visual representation using a standardized set of symbols and notations to describe a business's operations through data movement. They are often elements of a formal methodology such as Structured Systems Analysis and Design Method

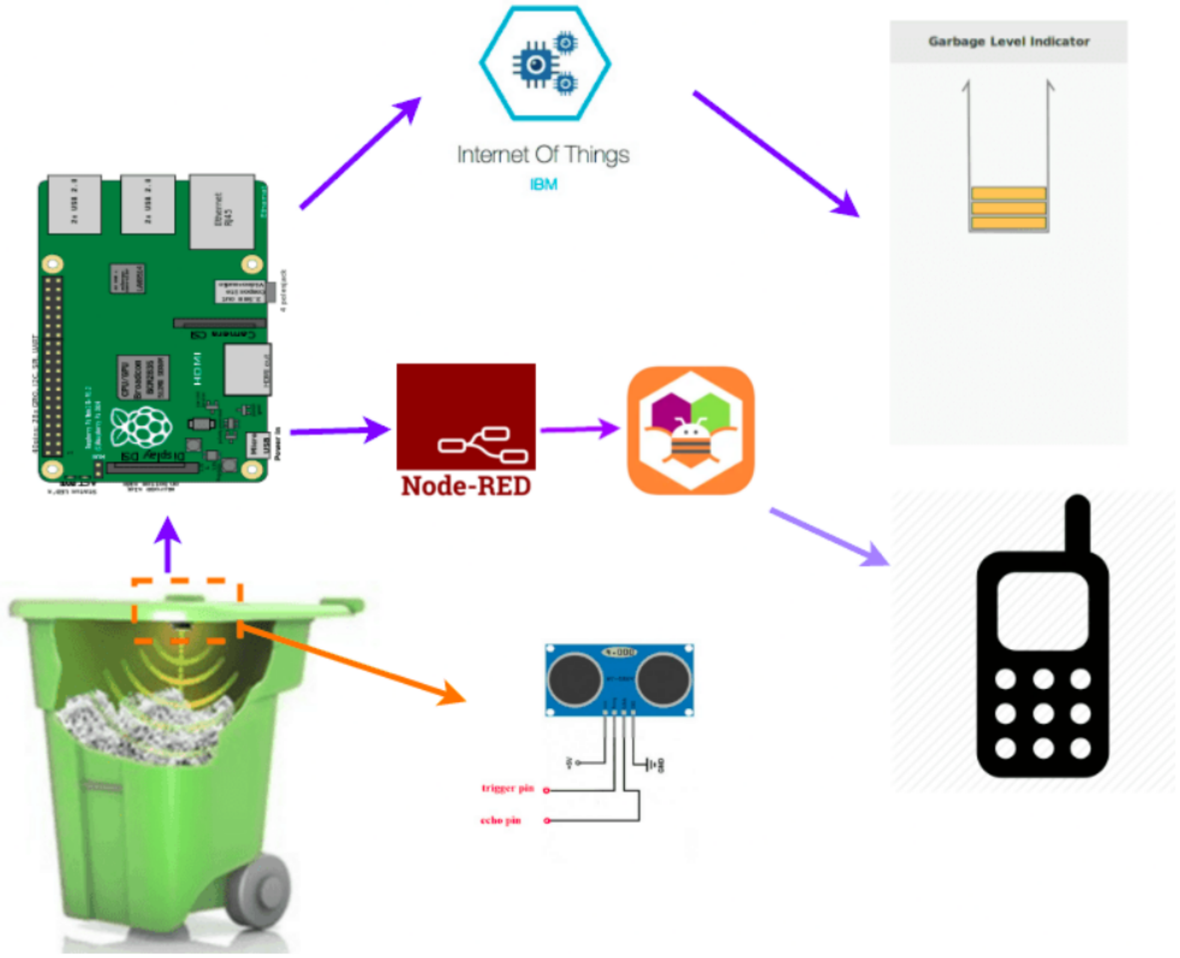
5.2 Solution & Technical Architecture:-


DFD
User
Login
Searching Nearest Bin
Dumping the waste
D
Storing and Sending Information About Bins
D
Managing storage information and sending alert to the admin
Admin
Login
Adding and Managing BIns
Receiving alert from the sensors and further processing to collect the waste
Collegting the waste and further processed

5.2 Solution & Technical Architecture:-

In this phase we will be discussing our solution architecture and technical architecture. We found the best solution is to manage the bins by using sensors and also bin weight the sensors will find the level of the bin and the weight is measured by weight load sensors and they will send the data to the watson assistant and the watson assistant will receive the data as a form of api and it will transfer the data to

the node red and by using node red we will deploy the site which will act as a website and by this website admin can get the information regarding various bin's by using this data admin can perform the arrangements for garbage collection and also we have developed an app by using MIT App the data which wa shown in MIP App.



5.3 User Stories

By using the below template we can see all the user stories for the product.

User Type	Functional Requirement (Epic)	User Story Number	User Story / Task	Acceptance criteria	Priority	Release
Customer (Mobile user)	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-1
		USN-2	As a user, I will receive confirmation email once I have registered for the application	I can receive confirmation email & click confirm	High	Sprint-1
		USN-3	As a user, I can register for the application through Facebook	I can register & access the dashboard with Facebook Login	Low	Sprint-2
		USN-4	As a user, I can register for the application through Gmail		Medium	Sprint-1
	Login	USN-5	As a user, I can log into the application by entering email & password		High	Sprint-1
	Dashboard	USN-6	As a user, I can view the data in bins	I can access the dashboard	High	Sprint-4
Customer (Web user)	Login	USN-7	As a user, I can register for the application by entering my email, password, and confirming my password.	I can access my account / dashboard	High	Sprint-3
Customer(Web user)	Dashboard	USN-8	As a user, I can view the data sent from the bins			Sprint-4

6. PROJECT PLANNING & SCHEDULING

Project Planning & Scheduling is a listing of a project's milestones, activities, and deliverables. Usually dependencies and resources are defined for each task, then start and finish dates are estimated from the resource allocation, budget, task duration, and scheduled events. A schedule is commonly used in the project planning and project portfolio management parts of project management. Elements on a schedule may be closely related to the work breakdown structure (WBS) terminal elements, the Statement of work, or a Contract Data Requirements List.

6.1 Sprint Planning & Estimation

The below template is used for sprint planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Login	USN-1	As an Administrator, I need to assist the routes for trash collection and manage the vehicles and manpower.	2	High	Srinivas
Sprint-1	Login	USN-2	As a Co-Admin, I will control the web portal and update the	1	High	Nithin
Sprint-2	Dashboard	USN-3	As an manager we will verify the waste collection process and procedure	2	Low	Pavan
Sprint-3	Dashboard	USN-4	As a lead finds the best place to dispose the waste	2	Medium	Sailesh
Sprint-4	Dashboard	USN-5	As a manager the recycling of the waste and try to make environment safe	1	High	Pavan

6.2 Sprint Delivery Schedule

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	15 Nov 2022

6.3 Reports from JIRA

Burnup report

Visualize a sprint's completed work and compare it with its total scope. Use these insights to track progress toward sprint completion.

Sprint burndown

Chart Track and manage the total work remaining within a sprint. After the sprint, summarize both team and individual performance.

Velocity report

Predict the amount of work your team can commit to in future sprints by seeing and reviewing the amount of value delivered in previous ones.

Cumulative flow diagram

Shows statuses about our project’s issues overtime. See which columns accumulate more issues, and identify bottlenecks in your workflow.

Cycle Time Report

Understand how much time it takes to ship issues through the deployment pipeline and how to deal with outlines.

Deployment Frequency Report

Understand your deployment frequency to understand risk and how often you are shipping value to your customers.

7. CODING & SOLUTIONING

In this Coding & Solution phase we will be discussing the code and features which we had done.

CODE

```
import sys

import ibmiotf.application
import ibmiotf.device
import ibmiotf.application
```

```
from geopy.geocoders import Nominatim
import time
import random




organization = "srpvqy"
deviceType = "Bin_Monitoring"
deviceId = "Bin_1"
authMethod= "token"
authToken = "(Z+UCAjRQvT)*@px)5"


def myCommandCallback(cmd):
    print("command received: %s" % cmd.data['command'])

    status = cmd.data['command']

    if status=="lighton":
        print("Led is on")
    else:
        print("Led is off")

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()
```

```
deviceCli.connect()

while True:

    level = random.randint(0, 100)

    weight = level*2.5

    ladegree = 12.0
    lodegree = 79.0

    laminute = random.randint(0, 60)
    lominute = random.randint(0, 60)

    lasecond = random.uniform(0, 3600)
    losecond = random.uniform(0, 3600)

    latitude = (round(((ladegree) + (laminute / 60) + (lasecond / 3600)), 14))
    longitude = (round(((lodegree) + (lominute / 60) + (losecond / 3600)), 14))

    if longitude > 80.33 or latitude > 13.39:
        continue
    else:

        geolocator = Nominatim(user_agent="MyApp")

        coordinates = str(latitude) + ", " + str(longitude)

        location = geolocator.reverse(coordinates)

        address = location.raw['address']
        city = address.get('city', '')
        state = address.get('state', '')
        country = address.get('country', '')
        town = address.get('town', '')
        village = address.get('village', '')
        municipality = address.get('municipality', '')
        suburb = address.get('suburb', '')
        county = address.get('county', '')

        l = [city, suburb, town, municipality, county, village]

        l = sorted(l, key=lambda x: (len(x)), reverse=True)

        fp = l[0]
        fs = state
        fc = country

    if len(fp) != 0 and len(fs) != 0 and len(fc) != 0:
        print(fp, fs, fc, sep="\n")
        data = {'level': level,'weight': weight,'coordinates': coordinates, 'city': fp, 'state': fs,
'country': fc}
        def myOnPublishCallback():
            print("published Level = %d " % level,"Weight = %d " % weight, "Coordinates
= %s " % coordinates, "City = %s " % fp, "State = %s " % fs, "Country = %s " %fc)
            # print("Published")
        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback())
        if not success:
            print("not connected to ibmiot")
            level=0
    if level>80:
        time.sleep(30)
    else:
        time.sleep(1)
    deviceCli.commandCallback = myCommandCallback
deviceCli.disconnect()
```

7.1 Feature 1

The first feature is that we can access the waste levels in the bins by using this feature we can monitor and empty the bins so citizens are able to dispose of their waste properly. And this makes the waste collection very easy in metropolitan cities and also we can save a lot of fuel, manpower and monitor the waste management.

By using this feature we can manage and access the waste disposal and recycling techniques that make the environment safe and secure.

7.2 Feature 2

The second feature is accessing the bin location. By using the bin location we can arrange all kinds of arrangements to collect the waste and make it simple for all the people to access and monitor the waste.

7.3 Feature 3

By using the coordinates we have generated the location so everyone can easily understand the bin location so it will be easy to dispose and collect the waste from bins.

8 Testing

Software Testing is a method to check whether the actual software product matches expected requirements and to ensure that software product is Defect free. It involves execution of software/system components using manual or automated tools to evaluate one or more properties of interest. The purpose of software testing is to identify errors, gaps or missing requirements in contrast to actual requirements.

8.1 Test Cases

A test case is a document, which has a set of test data, preconditions, expected results and postconditions, developed for a particular test scenario in order to verify compliance against a specific requirement. Test Case acts as the starting point for the test execution, and after applying a set of input values, the application has a definitive outcome and leaves the system at some end point or also known as execution postcondition.

8.2 User Acceptance Testing

User Acceptance Testing (UAT) is a type of testing performed by the end user or the client to verify/accept the software system before moving the software application to the production environment. UAT is done in the final phase of testing after functional, integration and system testing is done.

9. RESULTS

This is the final phase of the project. In this phase we will deploy the application and we will go through it and perform all kinds of operations.

9.1 Performance Metrics

Project metrics are used to track the progress and performance of a project. Monitoring parts of a project like productivity, scheduling, and scope make it easier for team leaders to see what's on track. As a project evolves, managers need access to changing deadlines or budgets to meet their client's expectations

10. ADVANTAGES & DISADVANTAGES

ADVANTAGES

- We can manage the waste in the bins easily
- We can avoid waste overflow problem
- We can save lot of fuel and manpower
- We can access the proper waste collection data
- We can maintain the city waste management

DISADVANTAGES

- It is difficult to access the data all the time properly there may be some malfunction in sensors.
- Some times location might not be accurate and it leads to difficult to find the bin
- Sometimes there may be difficulties to understand the user interface

11. CONCLUSION

By using the Smart Waste Management System For Metropolitan Cities we can make the waste management easier in metropolitan cities and this will be helpful to save lot of manpower and fuel, it makes easy to access the bin locations so it is easy to collect the waste in all the bins and also by using this techniques we can avoid the waste overflow.

12. FUTURE SCOPE

In the future we can make this project to link with smart robots so they will automatically go and collect the waste from specified locations and it will be most helpful for cities due to the huge amount of population and a lot of waste generated every day. It makes our life easier.

13. APPENDIX

Source Code

IBNIoT.py

```
import sys

import ibmiotf.application
import ibmiotf.device
import ibmiotf.application
from geopy.geocoders import Nominatim
import time
import random




organization = "srpvqy"
deviceType = "Bin_Monitoring"
deviceId = "Bin_1"
authMethod= "token"
authToken = "(Z+UCAjRQvT)*@px)5"


def myCommandCallback(cmd):
    print("command received: %s" % cmd.data['command'])
    status = cmd.data['command']

    if status=="lighton":
        print("Led is on")
    else:
        print("Led is off")

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()

deviceCli.connect()

while True:

    level = random.randint(0, 100)

    weight = level*2.5

    ladegree = 12.0
    lodegree = 79.0

    laminute = random.randint(0, 60)
    lominute = random.randint(0, 60)

    lasecond = random.uniform(0, 3600)
    losecond = random.uniform(0, 3600)
```

```
latitude = (round(((ladegree) + (laminute / 60) + (lasecond / 3600)), 14))
longitude = (round(((lodegree) + (lominute / 60) + (losecond / 3600)), 14))

if longitude > 80.33 or latitude > 13.39:
    continue
else:

    geolocator = Nominatim(user_agent="MyApp")

    coordinates = str(latitude) + ", " + str(longitude)

    location = geolocator.reverse(coordinates)

    address = location.raw['address']

    city = address.get('city', '')
    state = address.get('state', '')
    country = address.get('country', '')
    town = address.get('town', '')
    village = address.get('village', '')
    municipality = address.get('municipality', '')
    suburb = address.get('suburb', '')
    county = address.get('county', '')

    l = [city, suburb, town, municipality, county, village]

    l = sorted(l, key=lambda x: (len(x)), reverse=True)

    fp = l[0]
    fs = state
    fc = country
    if len(fp) != 0 and len(fs) != 0 and len(fc) != 0:
        print(fp, fs, fc, sep="\n")
        data = {'level': level,'weight': weight,'coordinates': coordinates, 'city': fp, 'state': fs,
'country': fc}
        def myOnPublishCallback():
            print("published Level = %d " % level,"Weight = %d " % weight, "Coordinates
= %s " % coordinates, "City = %s " % fp, "State = %s " % fs, "Country = %s " %fc)
            # print("Published")
        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback())
        if not success:
            print("not connected to ibmiot")
            level=0
    if level>80:
        time.sleep(30)
    else:
        time.sleep(1)
    deviceCli.commandCallback = myCommandCallback
deviceCli.disconnect()
```

GitHub Link

https://github.com/IBM-EPBL/IBM-Project-34421-1660235665

Project Demo Link

https://drive.google.com/drive/u/2/my-drive