SMART WASTE MANAGEMENT SYSTEM

USING IOT

A Project report submitted in partial fulfilment of 7th semester in degree

of

BACHELOR OF ENGINEERING

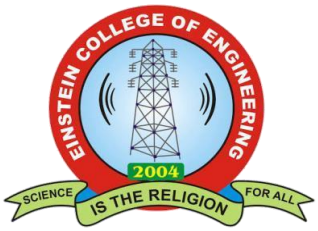
IN

COMPUTER SCIENCE AND ENGINEERING

Submitted by

Team ID: PNT2022TMID49859

M. ANJALAMANI	950619104006
K. BRINTHA	950619104014
M. GEETHA	950619104021
R. MAHIBA	950619104037



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

EINSTEIN COLLEGE OF ENGINEERING

TIRUNELVELI - 627012

BONAFIDE CERTIFICATE

Certified this Report “SMART WASTE MANAGEMENT SYSTEM” for the project, is the bonafide work of M. ANJALMANI (950619104006) K. BRINTHA (950619104014) M. GEETHA (950619104021) R. MAHIBA (950619104037) who carried out the project work under my supervision. Certified further that to the best of my knowledge the work reported here in does not from part of any other thesis or dissertation on the basis of which a degree or award was co-offered on the earlier occasion this or any other candidate.

SIGNATURE
Dr. T.SURESH THANKRISHNAN
HEAD OF THE DEPARTMENT
COMPUTER SCIENCE AND ENGINEERING
EINSTEIN COLLEGE OF ENGINEEING
TIRUNELVELI-12

SIGNATURE
Mrs. K. Sarika
ASSISTANT PROFESSOR
COMPUTER SCIENCE AND ENGINEERING
EINSTEIN COLLEGE OF ENGINEERING
TIRUNELVELI-12

ACKNOWLEDGEMENT

We have successfully completed the project with blessings showered onus by god, the almighty, A project of this nature needs co-opreation and support from many for successful completion.

We express our heartfelt thanks to **Mr.A.MATHIVANAN, BE., M.sc.,(Agri),** Managing Trustee of Einstein college of Engineering, Tirunelveli, for his mortal support and device.

Our thanks to **Prof.A.AMUTHAVANAN, BE., M.S (USA).B.L,**Chairman of our college for making necessary arrangements to do this project.

Our heartly thanks to **Prof.EZHILVANAN, MBA.,** Secretary of our college for making necessary arrangements to do this project.

We wish to express our gratitude to **Dr.VELAYUTHAM,M.E,Ph.D.,FIE.** Principal for the support he provided us to carry out this project successfully.

We are very much thankful to **Dr. T.SURESH THANGAKRISHNAN**, Head of the Department , Computer science and Engineering who is always a constant of inspiring us. We are extend our sincere thanks to all teaching and non-teaching staff members and our family members, friends for their help in completing this project.

TABLE OF CONTENTS

CHAPTER NO. TITLE

1. **INTRODUCTION**
   - 1.1 Project Overview
   - 1.2 Purpose
2. **LITERATURE SURVEY**
   - 2.1 Existing Problem
   - 2.2 References
   - 2.3 Problem Statement Definition
3. **IDEATION & PROPOSED SOLUTION**
   - 3.1 Empathy Map Canvas
   - 3.2 Ideation & Brainstorming
   - 3.3 Proposed Solution
   - 3.4 Problem Solution Fit
4. **REQUIREMENT ANALYSIS**
   - 4.1 Functional Requirements
   - 4.2 Non – Functional Requirements
5. **PROJECT DESIGN**
   - 5.1 Data Flow Diagrams
   - 5.2 Solution & Technical Architecture
   - 5.3 User Stories
6. **PROJECT PLANNING & SCHEDULING**
   - 6.1 Sprint Planning & Estimation
   - 6.2 Sprint Delivery Schedule
7. **CODING & SOLUTIONING**
   - 7.1 Feature 1
   - 7.2 Feature 2
   - 7.3 Database Schema
8. **TESTING**
   - 8.1 Test Cases
   - 8.2 User Acceptance Testing
9. **RESULTS**
   - 9.1 Performance Metrics
10. ADVANTAGES & DISADVANTAGES
11. CONCLUSION

12. FUTURE SCOPE
13. APPENDIX

Source Code

Github&Project demo Link

CHAPTER 1

INTRODUCTION

1.1 Project Overview

Waste Management is all the activities and actions required to manage waste from its inception to its final disposal. This includes collection, transportation, treatment and disposal of waste together with monitoring and regulation. Waste Collection methods vary widely among different countries and regions. Domestic waste collection services are often provided by local government authorities.

Curbside collection is the most common method of disposal in most countries, in which waste is collected at regular intervals by specialized trucks. Waste collected is then transported to an appropriate disposal area.

Nowdays, cities with developing economics experience exhausted waste collection services, inadequately managed and uncontrolled dumpsites and the problems are worsening. Waste collection method in such countries is an on-going challenge and many struggle due to weak institutions and rapid urbanization.

1.2 Purpose

- By 2030, almost two-third of the world's population will be living in cities. This fact requires the development of sustainable solutions for urban life, managing waste is a key issue for the health.
- Efficient and energy-saving waste management, reducing CO_2, air pollution and vehicle exhaust emissions-these are just a few examples for the demands of future cities. In views of that, the efficient use and responsible handling of resources become more important.
- Effectively managing waste is important in developed countries. Waste management swallow upto 50% of a city's budget, but only serve a small part of the population.
- Sometimes, upto 60% of waste is not being collected, it is often simply burned by the roadside. It can pollute drinking water, it can spread disease to people living nearby.
- Even with great route optimization, the worker must still physically go to the dustbin to check waste levels. Because of this, trucks often visit containers that do not need emptying, which wastes both time and fuel.
- Waste Management prevents harm to human health and the environment by reducing the volume and hazardous character of residential and industrial waste.
- Improving proper waste management will reduce pollution, recycle useful materials and create more green energy.

CHAPTER 2

LITERATURE SURVEY

2.1 Existing Problem

In the existing system garbage is collected by corporation by weekly once or by 2 days once. Tough the garbage shrinks and overflows the garbage bin and spread over the roads and pollutes the environment. The smell will be heavy and produces air pollution and spreads disease. The street dogs and animals eat the waste food and spreads over the area and creates dirty environment to avoid such situation we are planning to design IOT Based Garbage Management For Smart Cities.

2.2 References

1. Glossary of Environment Statistics : Series F, No. 67 / Department for Economic and Social Information and Policy Analysis, United Nations. New York: UN, 1997.

2. United Nations Environmental Programme *(2013). "Guidelines for National Waste Management Strategies Moving from Challenges to Opportunities."*

3. *"Arduino - Introduction"*. arduino.cc.

4. "Programming Arduino Getting Started with Sketches". McGraw-Hill. Nov 8, 2011. Retrieved 2013-03-28.

5. Anton A. Huurdeman, The Worldwide History of Telecommunications, John Wiley & Sons, 31 juli 2003, page 529

6. *"GSM Global system for Mobile Communications"*. 4G Americas. Retrieved 2014-03-22.

7. The national environment policy, 2006 available at http://www.tnpcb.gov.in/pdf/nep2006e.pdf

8. Florence Nightingale, Selected Writings of Florence Nightingale, ed. Lucy Ridgely Seymer (New York: The Macmillan Co., 1954), pp. 38287

9. National Waste & Recycling Association. *"History of Solid Waste Management"*. Washington, DC. Retrieved 2013-12-09.

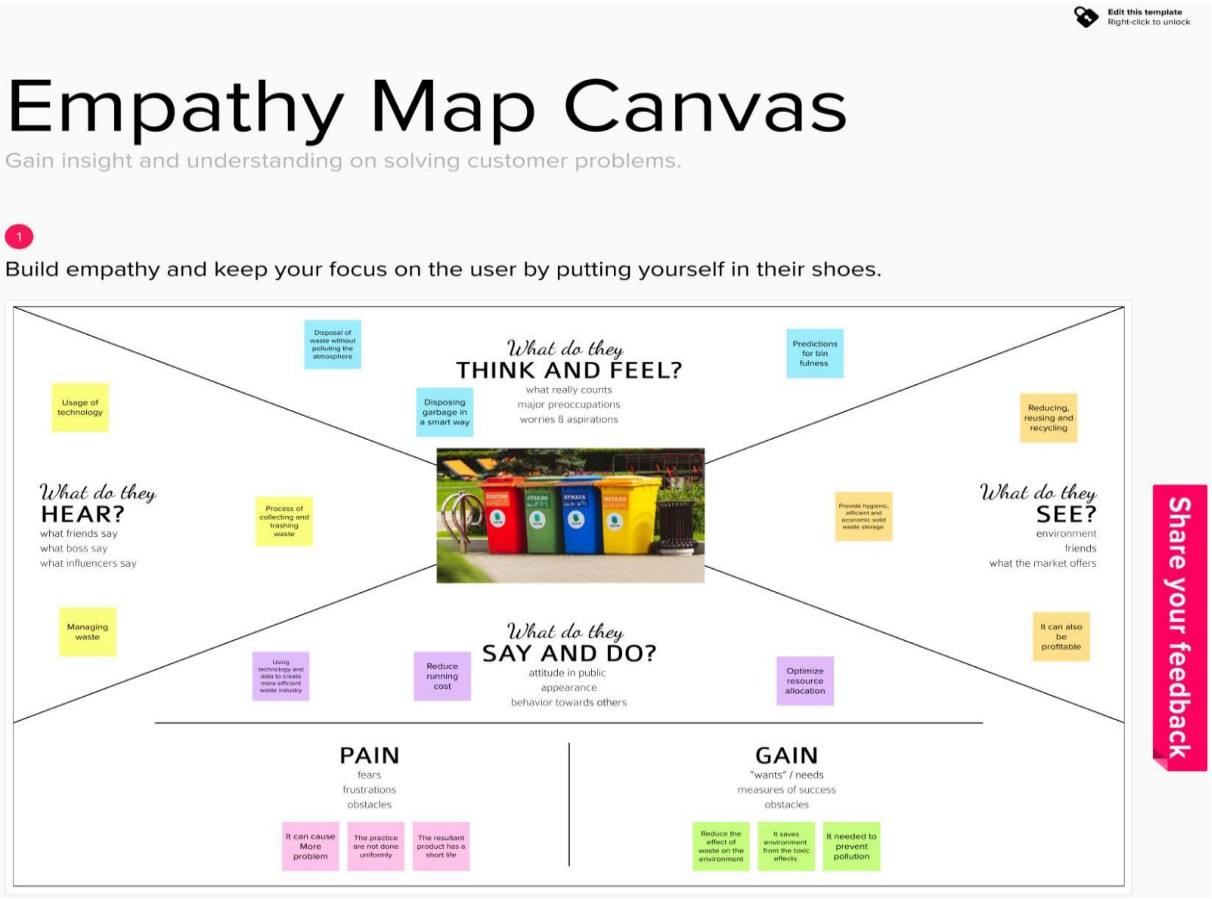
2.3 Problem Statement Definition

- The Main Problems of the existing solid waste collection process and management system are as follows.
- Lack of the information about the collecting time and area.
- Lack of the proper system for monitoring, tracking the trucks and trash bins that have been collected in real time.
- Loss of productivity due to inefficient utilization and unauthorized use of vehicles.
- There is no quick response to urgent cases like truck accident, breakdown, longtime idling.
- There is no quick way to response to client's complaints about uncollected waste.

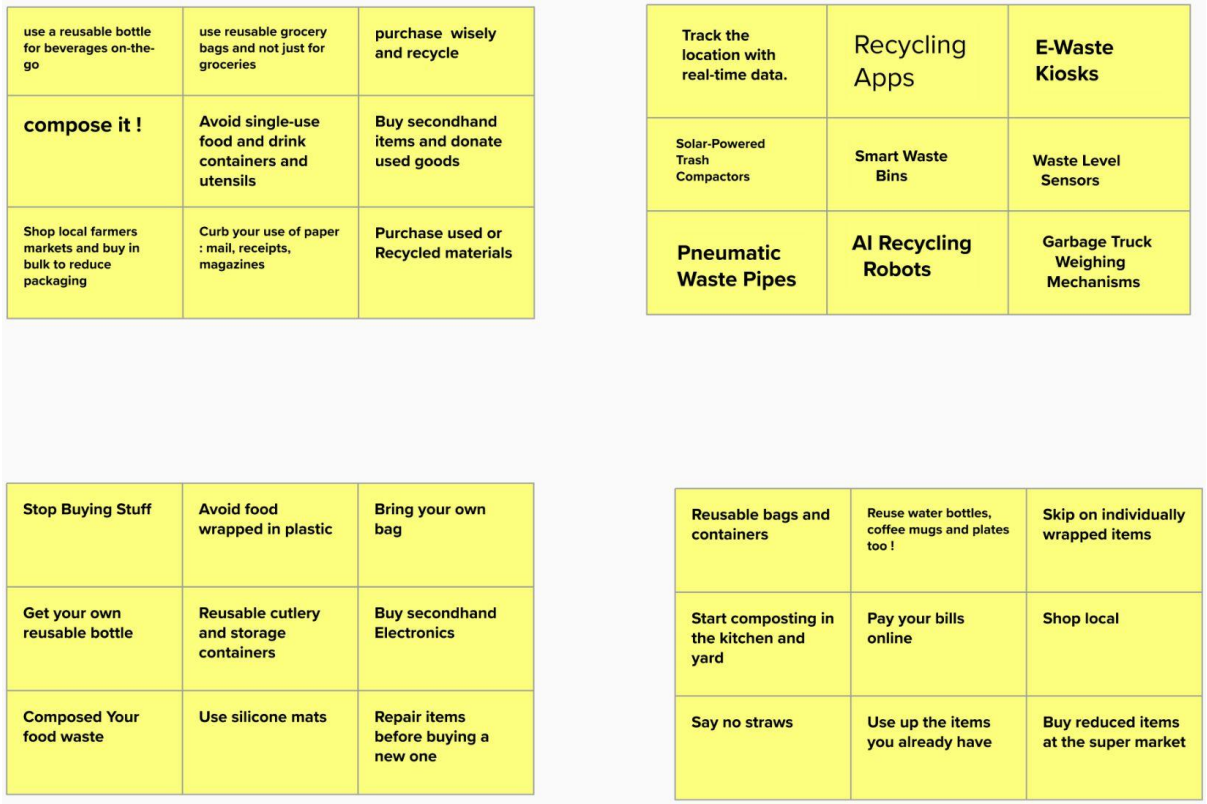
CHAPTER 3

IDEATION & PROPOSED SOLUTION

3.1 Empathy Map Canvas



3.2 Ideation & Brainstorming



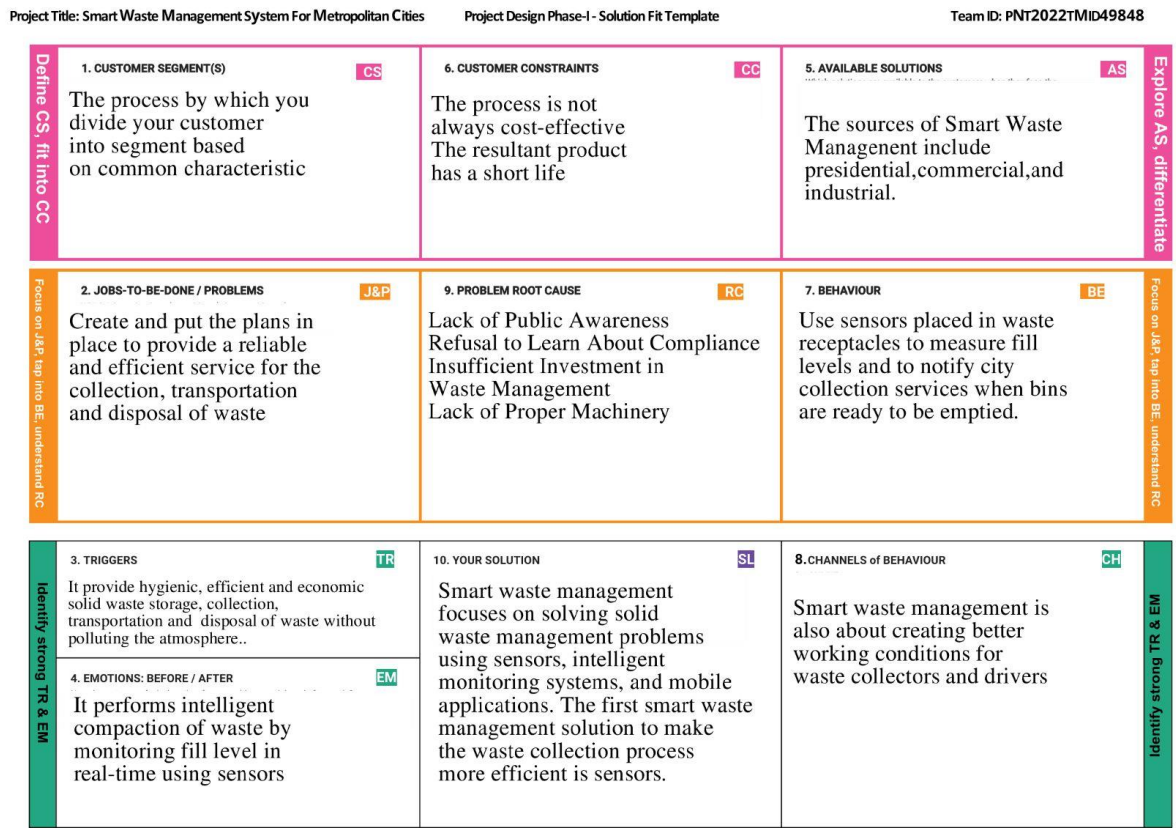
3.3 Proposed Solution

S.No.	Parameter	Description
1.	Problem Statement (Problem to be Solved)	With the existing methods of collecting and disposal it is near impossible to manage such amount of waste in the future as around 30% of waste end up on the roads and public places due to ineffective disposing and collecting methods.
2.	Idea/Solution Description	Smart waste management is about using technology and data to create a more efficient waste industry. Based on IoT(Internet of Things) technology, smart waste management aims to optimize resource allocation, reduce running costs, and

		increase the sustainability of waste services.
3.	Novelty/Uniqueness	Through its unique smart waste management technology, Sensoneo is redefining the way waste is managed. Sensoneo solutions cover from asset tracking for bins all the way to the automated on-demand collection planning.
4.	Social Impact/Customer	Using IoT and smart sensors, waste management companies can increase operational efficiency, cut costs and enhance customer satisfaction.
5.	Business Model(Revenue Model)	Waste Management generates revenue through the provision of various waste management and disposal services and recycling solutions to residential, commercial, industrial, and municipal clients. The Company derives its revenue in the form of various fees associated with its service offerings
6.	Scalability of the Solution	Scalable system for waste bins that can sense and send accurate waste level of the bins while consuming less resources and having cost-effective components. The system operates by utilizing ultrasonic sensors that senses and transmits waste fill-level estimations. The system was modelled, simulated using MATLAB and physical implemented. In the implementation RFID technology is employed having an active RFID tags that stores the information as

		well as RFID readers that reads and interpret the information.

3.4 Problem Solution Fit



CHAPTER 4

REQUIREMENT ANALYSIS

4.1 Functional Requirement

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. All the conditions may occur in system and these may include calculations, data manipulation and processing and other specific functionality. Some conditions are needed to specify the logical input otherwise the system will not function as per its implementation. In this system following are the functional requirements for the admin and the user.

- The Admin has to login by using valid user name and password.
- After Login successful he can do some operations such as add contents, view all contents, list all searching history, list ranking of images, list of all personalized search, attacker details, recover contents, list of all user and logout.
- The admin can add n-number of contents. If the admin want to add a new content, then admin will enter a URL, domain, title, description, uses related images of the particular content, then submit and that data will stored in data base.
- The admin can view list of all users. Here button, it will display all personalized search details.
- The time delay Generation chart results. This chart shows the time delay by using greedy DP and time delay using greedy IP.
- The user can attack contents and then user should enter name to attack and click on attack button.
- The Attributes are Privacy Protection, personalized web search, utility, risk, profile, profile based personalization, Admin, users.
- All register users are stored with the details such as user ID, user name, E-mail ID, mobile no, Location, date of birth, address, pin code, general key and personalized key.
- The admin can view the attacker details. If admin clicks on attacker details button, the admin will get attacker information.
- There are n numbers of users are present. User should register before doing some operations. After registration successful he has to login by using authorized user name and password.

4.2 Non-Functional Requirements

Non-functional Requirements, as the name suggests, are those requirements that are not directly concerned with the specific functions delivered by the system. They may relate to

emergent system properties such as reliability response time and store occupancy. Alternatively, they may define constraints on the system such as the capability of the input output devices and the data representations used in system interfaces. Many non-functional requirements relate to the system as whole rather than to individual system features. This means they are often critical than the individual functional requirements. The following non-functional requirements are worthy of attention. The key non-functional requirements are

1. SECURITY

   The system should allow a secured communication between server, Admin and Users.

2. ENERGY EFFICIENCY

   The Energy Consumed by the users to receive the file information from the server and admin.

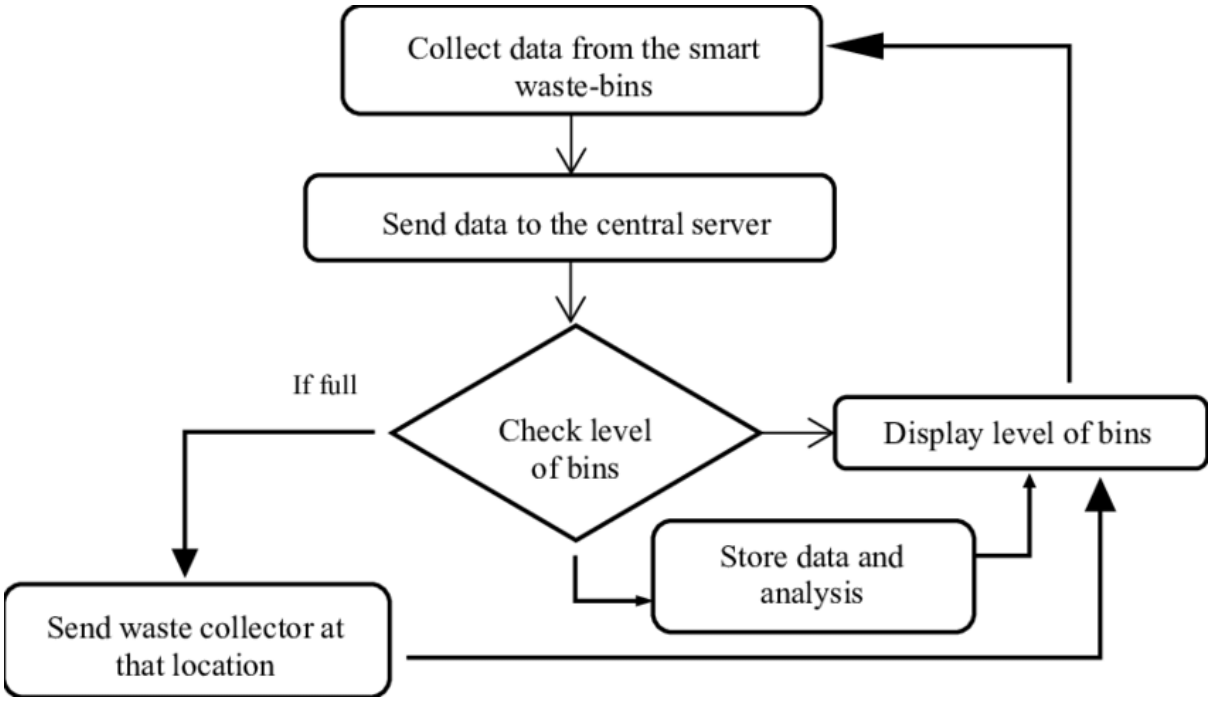
3. RELIABILITY

   The System should be reliable and must not degrade the performance of the existing system and should not lead to the hanging of the system.

CHAPTER 5

PROJECT DESIGN

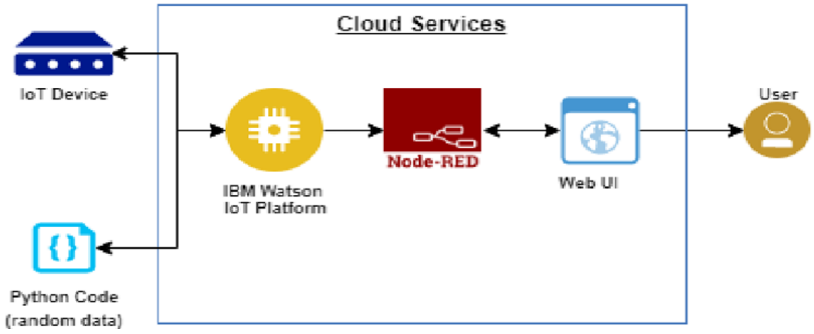
5.1 Data Flow Diagrams

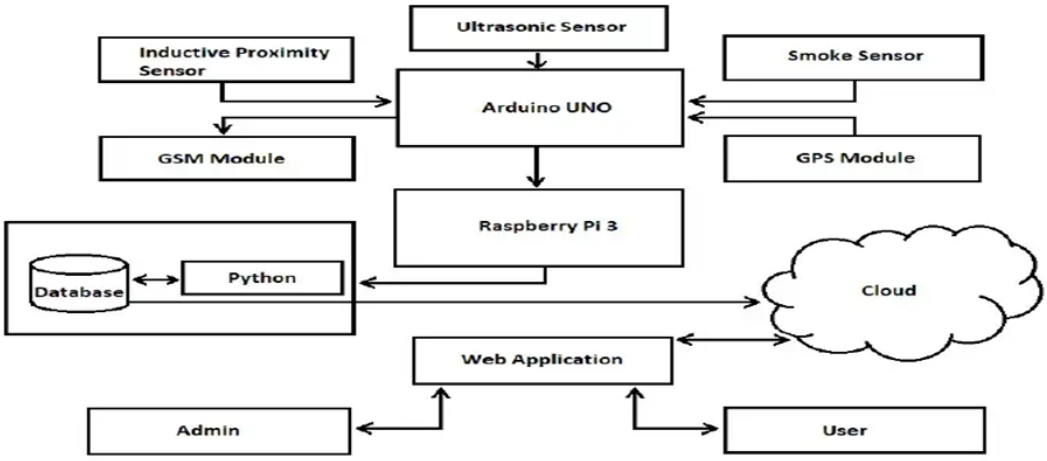


5.2 Solution & Technical Architecture

Solution Architecture

A system consist of an inductive proximity sensor on the conveyor belt and ultrasonic sensors at the top of the dustbin, a smoke sensor to detect fire sensor each smart bin is assigned with GPS to provide the location and a GSM to send the message to the workers. All the sensors and modules are connected directly to ardino and they are controlled by the Raspberry Pi board. Ardino's programs monitors sensors and issues saction based on the status of the bin.





5.3 User stories

User Type	Functional Requirement (Epic)	User Story Number	User Story/ Tas	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.	IIcan access any account dashboard	High	Sprint-1
		USN-2	Asa user Facebook Login, 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 & accesss the dashboard with	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					

CHAPTER 6

PROJECT PLANNING & SCHEDULING

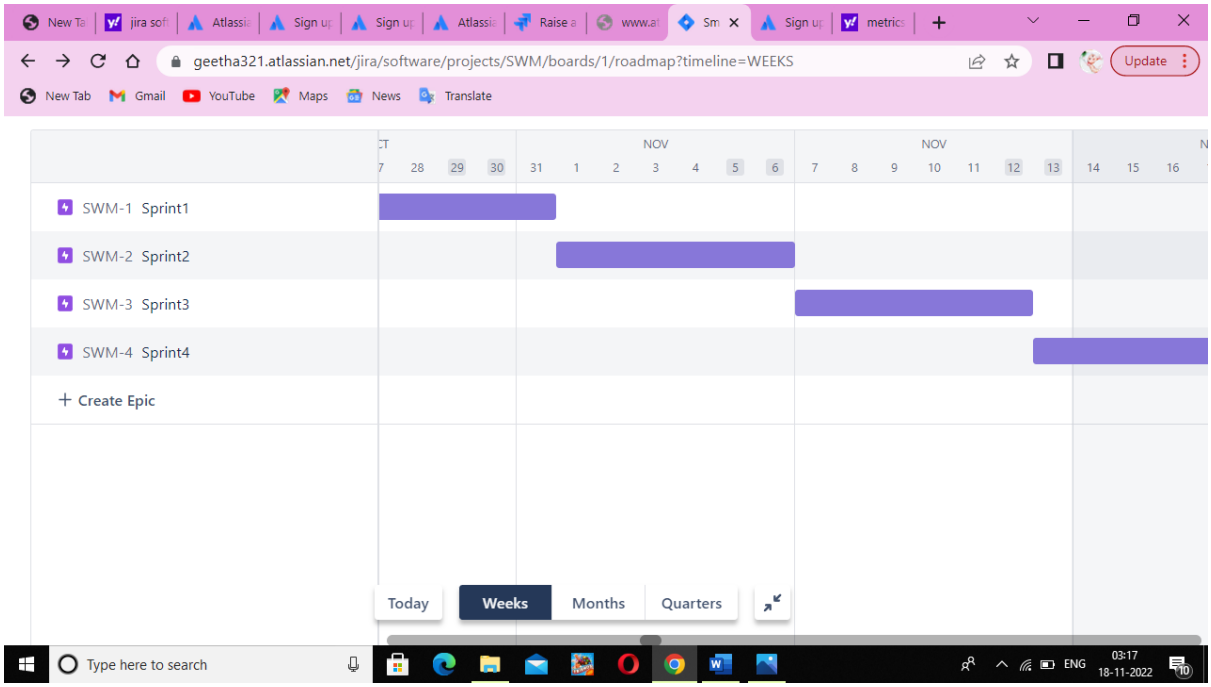
6.1 Sprint Planning & Estimation

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Point	Priority	Team Members
Sprint-1	Registration	USN-2	As a user,1 can register for the application by entering my email, password, and confirming my password	2	High	M.Geetha
Sprint-1		USN-2	As a user,1 will receive confirmation For the application through Gmail	1	High	M.Geetha
Sprint-1	Login	USN-5	As a User, 1 can log into the application by entering email & password	1	High	M.Geetha
Sprint-1		USN-4	As a user, 1 can register I can log into the application by entering email & password	2	Medium	R.Mahiba
Sprint-2		USN-3	As a user, I can register for the application through gmail.	2	Low	K.Brintha

6.2 Print Delivery Schedule

Sprint	Total Story Point	Duration	Sprint Start Date	Sprint end Date (Planned)	Story Point Completed (as on planned end date)	Sprint Release Date (Actual)
Sprint-1	20	6 Days	26 Oct 2022	31 Oct 2022	20	31 Oct 2022
Sprint-2	20	6 Days	1 Nov 2022	6 Nov 2022	20	6 Nov 2022
Sprint-3	20	6 Days	7 Nov 2022	12 Nov 2022	20	12 Nov 2022
Sprint-4	20	6 Days	13 Nov 2022	18 Nov 2022	20	18 Nov 2022

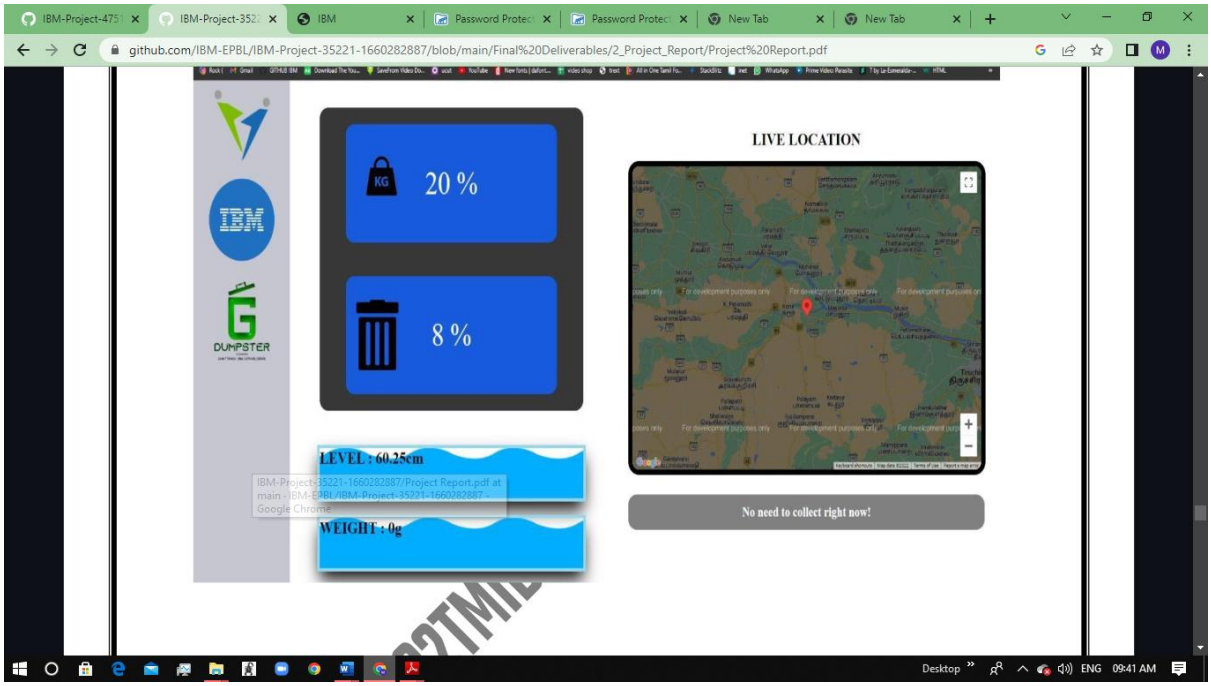
6.3 Reports from JIRA



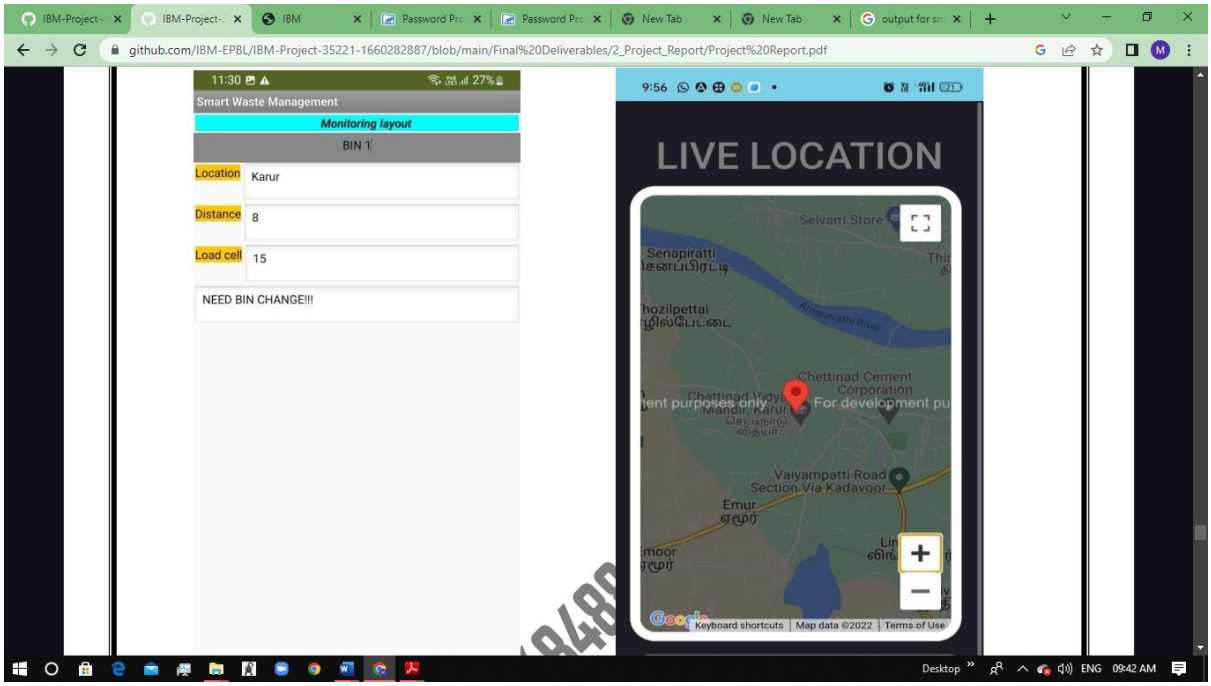
CHAPTER 7

CODING & SOLUTIONING (Explain the features added in the project along with code)

7.1 Feature 1



7.2 Feature 2



CHAPTER 8

TESTING

8.1 Test Cases

Test Case ID	Test Case Name	Inputs	Excepted Outputs	Actual Outputs
1	Garbage Full	Garbage Level=85%	Garbage is full	Garbage is full
2	Garbage Empty	Garbage Level=0%	Garbage is empty	Garbage is empty
3	Garbage Filling	Garbage Level=40%	No Output	Garbage filled 40%
4	Garbage Overflow	Garbage=110%	Error	Garbage is full

8.2 User Acceptance Testing

- **Identify the Audience**

  UAT workflows often feel like a blindfolded relay race. The handover has so many dependencies users are unaware of. Identifying the target audience is critical to determining end-user problems and needs. Choosing the right business users for UAT is vital as their feedback helps track changes and improve the software significantly.

- **Support Key Users**

  Help your key users execute their steps without dependencies, even in a multi-tester business process with embedded workflow automation capabilities within the business process. You can set a "Reset" notification for testers to update when a bug is fixed and a "close" notification for developers when a retest is successful.

- **Declare Testing Objectives**

  Declare the objectives of what you want to verify and validate, including the scope. Smart testing determines what gets tested and what doesn't. Eliminate the hassle of realigning each project and encourage critical users to focus on testing activities and increase productivity.

- **Automate Workflows**

  Automated record-and-play reduces back-and-forth between developers and testers and provides a record of steps for test reproducibility. In addition, it has the added value that the documentation is always audit-proof and ensures compliance with all interval and external quality standards.

- **System-Wide defects solutions**

  Defect inflation occurs when multiple bugs are associated with the same objects or problem, System-wide fault resolution automates fault monitoring and management. This allows you to quickly determine the impact of a single defect on your project and define the degree of impact on your testing activities.

- **Smart defect Management**

  It includes centralized monitoring and resolution of defects and the business processes they affect to identify impacted tests and block or warn testers until the primary defect is resolved.
- **Business objective confirmation**

  Sign-off of UAT before going live. The sign-off approval indicates that the change meets business requirements and is ready for deployment.

CHAPTER 9

9.1 Performance Metrics

This system monitors the garbage bins and informs about the level of garbage collected in the garbage bins via a web page. For this the system uses ultrasonic sensors placed over the bins to detect the garbage level and compare it with the garbage bins depth.

CHAPTER 10

ADVANTAGES

- It saves time and money by using smart waste collection bins and systems equipped with fill level sensors. As smart transport vehicles go only to the filled containers or bins. It reduces infrastructure, operating and maintenance costs by upto 30%.
- It decreases traffic flow and consecutively noise due to less air pollution as result of less waste collection vehicles on the roads. This has become possible due to two way communication between smart dustbins and service operators.
- It Keeps our surroundings clean and green and free from bad odour of wastes, emphasizes on healthy environment and keep cities more beautiful.
- It further reduces manpower requirements to handle the garbage collection process.
- Applying smart waste management process to the city optimizes management, resources and costs which makes it a “smart city”

DISADVANTAGES

- System requires more number of waste bins for separate waste collection as per population in the city. This results into high initial cost due to expensive smart dustbins compare to other methods.
- Sensor nodes used in the dustbins have limited memory size.
- Wireless technologies used in the system such as zigbee and wifi have shorter range and lower data speed, In RFID based systems, RFID tags are affected by surrounding metal objects.
- It reduces man power requirements which results into increase in unemployments for unskilled people.
- The training has to be provided to the people involved in the smart waste management system.

CHAPTER 11

CONCLUSION

Analysis has been carried out and results indicate that urban solid waste comprises mostly biodegradable and non-biodegradable materials. Furthermore, this is not done often by the department responsible for the evacuation of this waste. Furthermore, it was observed that due to the rapid municipal situation in India, rise in unexpected slums and residential buildings, and absence of sustainable waste management technology in India. The current waste disposal situation is likely to worsen. The bulk of the waste comes from the private industries and recycling in the metropolis is technically not officially carried out. This paper will help to solve the waste Management system in India and also other developing countries.

CHAPTER 12

FUTURE SCOPE

- Implementation is done only for a a single bin,
- Integration of many bins each with unique id can be done by implementing principles of IOT
- Database can be created for each bin by using SQL technology.
- Automated system can be developed which is able to pick up waste in and around the bin, segregate them and put them in respective bins.

CHAPTER 13

APPENDIX

Source Code

```
import requests
import json
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys
# watson device details

organization = "4yi0vc"
devicType =  "BIN1"
deviceId = "BIN1ID"
authMethod= "token"
authToken= "123456789"

#generate random values for randomo variables (temperature&humidity)




def myCommandCallback(cmd):
    global a
    print("command recieved:%s" %cmd.data['command'])
    control=cmd.data['command']
    print(control)

try:
```

```
    deviceOptions={"org": organization, "type": devicType,"id": deviceId,"auth-method":authMethod,"auth-token":authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
    print("caught exception connecting device %s" %str(e))
    sys.exit()

#connect and send a datapoint "temp" with value integer value into the cloud as a type of event for every 10 seconds
deviceCli.connect()

while True:

  distance= random.randint(10,70)
  loadcell= random.randint(5,15)
  data= {'dist':distance,'load':loadcell}


  if loadcell < 13 and loadcell > 15:
    load = "90 %"

  elif loadcell < 8 and loadcell > 12:
     load = "60 %"

  elif loadcell < 4 and loadcell > 7:
     load = "40 %"
  else:
     load = "0 %"

  if distance < 15:
     dist = 'Risk warning:' 'Dumpster poundage getting high, Time to collect :) 90 %'
```

```
    elif distance < 40 and distance >16:
        dist = 'Risk warning:' 'dumpster is above 60%'

    elif distance < 60 and distance > 41:
        dist = 'Risk warning:' '40 %'
    else:
        dist = 'Risk warning:' '17 %'



    if load == "90 %" or distance == "90 %":
        warn  = 'alert :' ' Dumpster poundage getting high, Time to collect :)'

    elif load == "60 %" or distance == "60 %":


        warn = 'alert :' 'dumpster is above 60%'
    else :
        warn = 'alert :' 'No need to collect right now '
    def myOnPublishCallback(lat=10.678991,long=78.177731):
       print("Gandigramam, Karur")
       print("published distance = %s " %distance,"loadcell:%s " %loadcell,"lon = %s " %long,"lat = %s"
%lat)
       print(load)
       print(dist)
       print(warn)

    time.sleep(10)



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

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




    if not success:
        print("not connected to ibmiot")
    time.sleep(30)






    deviceCli.commandCallback=myCommandCallback
#disconnect the device
deviceCli.disconnect()



 import requests
import json
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys
# watson device details
organization = "4yi0vc"
devicType = "BIN2"
deviceId = "BIN2ID"
authMethod= "token"
authToken= "123456789"
#generate random values for randomo variables (temperature&humidity)
def myCommandCallback(cmd):
global a
print("command recieved:%s" %cmd.data['command'])
control=cmd.data['command']
print(control)
try:
deviceOptions={"org": organization, "type": devicType,"id": deviceId,"auth-
method":authMethod,"auth-token":authToken}
```

```
deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
print("caught exception connecting device %s" %str(e))
sys.exit()
#connect and send a datapoint "temp" with value integer value into the cloud as a type of event for
every 10 seconds
deviceCli.connect()
while True:
distance= random.randint(10,70)
loadcell= random.randint(5,15)
data= {'dist':distance,'load':loadcell}
if loadcell < 13 and loadcell > 15:
load = "90 %"
elif loadcell < 8 and loadcell > 12:
load = "60 %"
elif loadcell < 4 and loadcell > 7:
load = "40 %"
else:
load = "0 %"
if distance < 15:
dist = 'Risk warning:' 'Dumpster poundage getting high, Time to collect :) 90 %'
```

```
elif distance < 40 and distance >16:
dist = 'Risk warning:' 'dumpster is above 60%'
elif distance < 60 and distance > 41:
dist = 'Risk warning:' '40 %'
else:
dist = 'Risk warning:' '17 %'
if load == "90 %" or distance == "90 %":
warn = 'alert :' ' Dumpster poundage getting high, Time to collect :)'
elif load == "60 %" or distance == "60 %":
warn = 'alert :' 'dumpster is above 60%'
else :
warn = 'alert :' 'No need to collect right now '
def myOnPublishCallback(lat=10.939091,long=78.135731):
print("Bustand, Karur")
print("published distance = %s " %distance,"loadcell:%s " %loadcell,"lon = %s " %long,"lat = %s" %lat)
print(load)
print(dist)
print(warn)
```

```
time.sleep(10)
success=deviceCli.publishEvent ("IoTSensor","json",warn,qos=0,on_publish= myOnPublishCallback)
success=deviceCli.publishEvent ("IoTSensor","json",data,qos=0,on_publish= myOnPublishCallback)
if not success:
print("not connected to ibmiot")
time.sleep(30)
deviceCli.commandCallback=myCommandCallback
#disconnect the device
deviceCli.disconnect()
```

Github Link

IBM-EPBL/IBM-Project-47515-1660800056

Project Demo Link

https://youtu.be/-8Vanw0c7gs