

**D. Y. PATIL COLLEGE OF ENGINEERING & TECHNOLOGY**

**KASABA BAWADA, KOLHAPUR**

**DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING**

(2018-2019)



**A**

**Domain Specific Mini Project Synopsis**

**On**

**“Waste Monitoring”**

**Submitted by:**

<b>Roll No.</b>	<b>Name</b>
36	Ruturaj Hemant Desai
37	Ruturaj Ravindra Sawant
38	Siddhi Ashok Jadhav
39	Rushikesh Ravindra More
40	Karan Dattatray Kenjale

**UNDER THE GUIDANCE OF**  
**Prof. Mr. K.T. Mane**

**Class: TE (CSE)**

**Div.: A**

**Batch: T3**

# INDEX

S.N.	Title	Pg.no
1.	Abstract	
2.	Introduction	
3.	Problem statement	
4.	Objectives	
5.	Proposed System Architecture	
6.	Modules	
7.	System Requirements	
8.	Conclusion	
9.	References	

## **1. Abstract**

In this project we will be using ultrasonic sensors for measuring the waste deposited in the respective garbage bin, thus analyzing and reporting the bin divers for the collection of the waste. In this project, we are providing a statistical report of garbage bins with the help of web user interface, notify the bin divers via SMS. We are also keeping a database in which we will store analyzed data sensed through ultrasonic sensors installed in respective bins. Our aim is to provide an optimized path for bin divers (Garbage Collectors), thus providing efficient diving time.

## **2. Introduction**

Internet of Things (IoT) is a concept in which surrounding objects are connected through wired and wireless networks without user intervention. In the field of IoT, the objects communicate and exchange information to provide advanced intelligent services for users. This project deals with the problem of Waste management, where the garbage collection system is not optimized. This project enables the organizations to meet their needs of the smart garbage management system. This system allows the user to know the fill level of each garbage bin, to give an efficient and time-saving route to the bin divers. Project will be build using Ultrasonic Sensors and raspberry PI breadboard. The Ultrasonic Sensor sends out a high-frequency sound pulse and then times how long it takes for the echo of the sound to reflect back. The sensor has 2 openings on its front. One opening transmits ultrasonic waves, (like a tiny speaker), the other receives them, (like a tiny microphone). The speed of sound is approximately 341 meters (1100 feet) per second in air. The ultrasonic sensor uses this information along with the time difference between sending and receiving the sound pulse to determine the distance to an object. The Built-in WIFI module in Raspberry Pi is used for transmitting data over the internet. The Sensors are physically connected to the Raspberry Pi via Breadboard using jumper cables. A breadboard is a construction base for prototyping of electronics. “Breadboard” is also a synonym for “prototype”. Because the solderless breadboard does not require soldering, it is reusable. A jump wire is an electrical wire or group of them in a cable with a connector or pin at each end (or sometimes without them – simply “tinned”), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. This project here is a model of the large-scale application which spans areas, localities, cities, etc. We are here to display the live working of the model and give an idea about the actual implications.

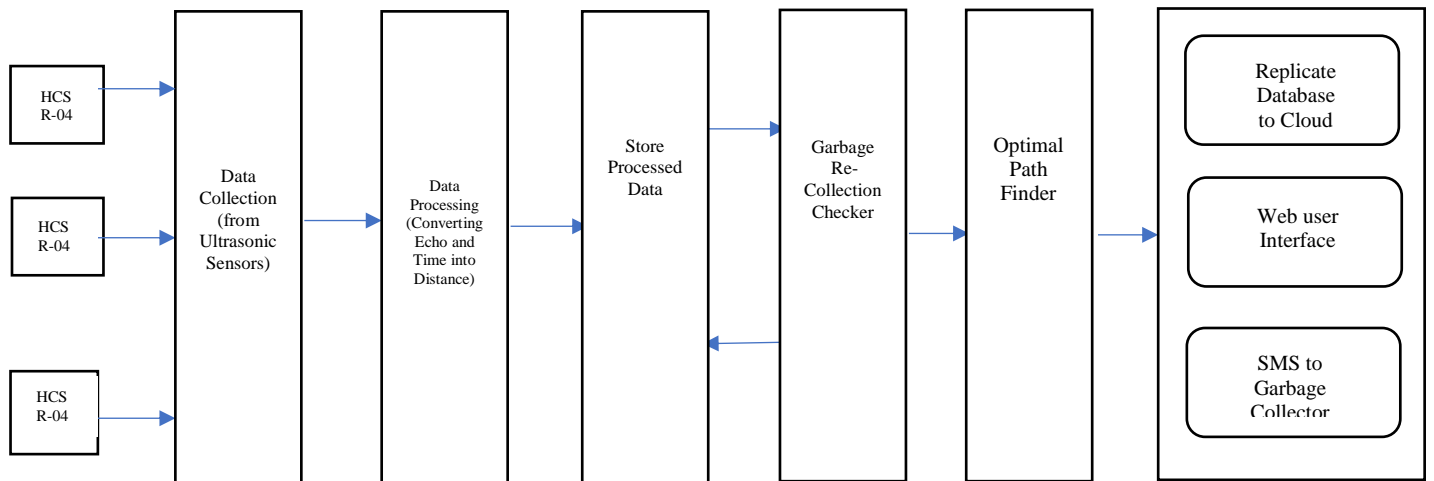
### **3. Problem Statement**

To design and implement a system for waste monitoring to provide an optimal path for bin diving.

### **4. Objectives**

- To design and build a prototype of a system that could efficiently manage bin-diving.
- To provide the ideal diving schedules.
- To implement this, ultrasonic sensors HC-SR04 used attached to each bin forming a layered structure.
- The sensed data is then forwarded over the cloud to the database.
- Then this data will be processed using an algorithm.
- The algorithm provides an optimized path for collecting waste from the bin.
- The optimal diving schedule is forwarded to the respective waste management organization and data is then displayed on the web interface.

## 5. Proposed System Architecture



**Fig (1).**

## **6. Modules**

### **6.1 Designing**

In this module, we are setting up raspberry pi and the ultrasonic sensors. First, the Raspbian OS is installed on the raspberry pi. Then we connect the ultrasonic sensors to the raspberry pi's GPIO via breadboard using jumper cables. The sensors are then attached to the bins. The sensors we are using is HC-SR04 ultrasonic range sensor.

The HC-SR04 Ultrasonic sensor we'll be using in this project for the Raspberry Pi has four pins: ground (GND), Echo Pulse Output (ECHO), Trigger Pulse Input (TRIG), and 5V Supply (Vcc). We power the module using Vcc, ground it using GND, and use our Raspberry Pi to send an input signal to TRIG, which triggers the sensor to send an ultrasonic pulse. The pulse waves bounce off any nearby objects and some are reflected back to the sensor. The sensor detects these return waves and measures the time between the trigger and returned pulse, and then sends a 5V signal on the ECHO pin. ECHO will be "low" (0V) until the sensor is triggered when it receives the echo pulse. Once a return pulse has been located ECHO is set "high" (5V) for the duration of that pulse. Pulse duration is the full time between the sensor outputting an ultrasonic pulse, and the return pulse is detected by the sensor receiver.

### **6.2 Data Collection (From Ultrasonic sensors)**

The python program is implemented for collecting the data from the sensors as they are connected to raspberry pi using GPIO pins. So, the data sent by the sensors need to be collected by that python script and sent to the next Module for processing.

### **6.3 Data Processing (Converting Echo and Time into Distance)**

The data received from these sensors is in form of pulses and time so we need to convert that to actual distance using some formulas. The data is processed using a python script. For this script, we are using some python libraries such as Rpi and Time.

### **6.4 Store Processed Data**

Then the processed data is stored in the database for further analysis. The database we are using is MySQL for local storage. For this, we are using the library MySQLdb. The stored data is again replicated to the cloud database and it is also used to plot the graph for monthly analysis of garbage collection.

## **6.5 Garbage Recollection Checker**

In this module, we are analyzing data from the database. We are calculating if the recollection of garbage is required or not using the program we are developing.

## **6.6 Optimal Path Finder**

If the recollection is required then we are calculating optimal path using the custom algorithm. The algorithm is customized using the shortest path algorithm. The path is provided based on a collection of garbage for the particular bin at the optimal time.

## **6.7 Replicate Local Database to AWS Cloud**

In this module, we are replicating the local MySQL database the database instance in the cloud. This database will help in updating web user interface and sending a notification to the bin divers. We can also use this database to show weekly, monthly statistics.

## **6.8 Updating Web UI**

In this module, we graphically represent of waste collection stats with the help of Web User Interface. This stored data is continuously fetched from the database and updated on Web UI after periodic time. The information provided by Web UI will be very helpful for respective waste management organizations.

In the development of this module, we are using some of the scripting and programming languages. For developing Front-End of Web UI we are using HTML, CSS, Javascript languages and their respective frameworks such as Bootstrap 4, Semantic UI, ReactJS. For developing Back-End we are using Python programming language and its framework Django.

## **6.9 SMS to the garbage collector**

In this module, we are using AWS's SMS API to send a message to the driver. The API we are using is Amazon Web Services, Simple Notification Service (SNS). This API will help us in sending optimal path to the driver.

## 7. System Requirements:

- **Hardware**

HC-SR04 Module

Resistors: 330Ω and 470Ω

Jumper wire

Breadboard

RAM : 1 GB

Processor : Pentium 4 onwards

- **Software**

Languages : Python, HTML, CSS, JavaScript, Bootstrap, D3.js

Database : MySQL

Operating system : Windows

Compiler : Python Interpreter

## 8. Conclusion

This Domain Specific Mini Project is developed to provide the optimal path for garbage collectors which results in the efficient waste collection. This also helps to reduce overflowing of garbage from a container which causes health issues. The provided survey will help in the planning of better waste management schemes for organizations.



## 9. References

### Website Links

<https://tutorials-raspberrypi.com/>

<https://www.geeksforgeeks.org/probabilistic-shortest-path-routing-algorithm-optical-networks/>

<https://stackoverflow.com/>

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element>

[www.geeksforgeeks.com](http://www.geeksforgeeks.com)

<https://www.python.org/>

<https://d3js.org/>

### Group members:

Roll no.	Name	Sign
36	Ruturaj Hemant Desai	
37	Ruturaj Ravindra Sawant	
38	Siddhi Ashok Jadhav	
39	Rushikesh Ravindra More	
40	Karan Dattatray Kenjale	

**Date:**

**Place:** Kolhapur

**Mr. K. T. Mane**  
**(Guide)**

**Ms. R. J. Dhanal**  
**(Coordinator)**

**Prof. Dr. G. A. Patil**  
**(H. O. D.)**