**Smart Waste Management System Using IoT**



**Project ID: Fall-2020-00**

**Session: BSCS Fall Fall 2018 to 2022**

**Project Advisor: Ma’amMubashiraAhmed**

**Submitted By**

|  |  |
| --- | --- |
| **Hassaan Bin Fida** | 70067770 |
| **Nabeel Sheraz** | 70067228 |
| **Fatima Asad** | 70068631 |
|  |  |

Department of Computer Science & IT

The University of Lahore

Lahore, Pakistan

## *Declaration*

We have read the project guidelines and we understand the meaning of academic dishonesty, in particular plagiarism and collusion. We hereby declare that the work we submitted for our final year project, entitled **Smart Waste Management System Using IOT** original work and has not been printed, published, or submitted before as final year project, research work, publication, or any other documentation.

## 

**Group Member 1 Name: Hassaan Bin Fida**

**SAP No: 70067770**

**Signature: …………………………**

**Group Member 2 Name: Nabeel Sheraz**

**SAP No: 70067228**

**Signature: …………………………**

**Group Member 3 Name: Fatima Asad**

**SAP No: 70068631**

**Signature: …………………………**

## *Statement of Submission*

This is to certify that **Hassaan Bin Fida** Roll No.**70067770, Nabeel Sheraz** Roll No. **70067228,** and **Fatima Asad** Roll No. **70068631** have successfully submitted the final project named as **Smart Waste Management System using IOT**, at Computer Science & IT Department, The University of Lahore, Lahore Pakistan, to fulfill the partial requirement of the degree of **BS in Computer Science**.

**Supervisor Name: Ma’am Mobashirah Nasir**

**Signature: …………………………**

**Date: ………………………**

## *Dedication*

This project is dedicated to my father, who taught me that the best kind of knowledge to have is that which is learned for its own sake. It is also dedicated to my mother, who taught me that even the largest task can be accomplished if it is done one step at a time.

## *Acknowledgment*

## 

We truly acknowledge the cooperation and help make by Name of **Ma’am Mobashirah Nasir**, **University of Lahore**. He has been a constant source of guidance throughout the course of this project. We would also like to thank **Ma’am Mobashirah Nasir** from the **University of Lahore** for his help and guidance throughout this project. We are also thankful to our friends and families whose silent support led us to complete our project.

Date:

Jan 1, 2022

## *Abstract*

With the rapid increase in population, the issues related to sanitation concerning garbage management are degrading immensely. It creates unhygienic conditions for the citizens in the nearby surrounding, leading to the spread of infectious diseases and illness. To avoid this problem, IoT-based “Smart Waste Management” is the best and trending solution. In the proposed system, public dustbins will be provided with an embedded device that helps in real-time monitoring of the level of garbage in garbage bins. The data regarding the garbage levels will be used to provide an optimized route for garbage collecting vans, which will reduce costs associated with fuel. The load sensors will increase the efficiency of data related to garbage levels, and moisture sensors will be used to provide data on waste segregation in a dust bin. The analysis of ceaseless data gathered will help municipality and government authorities to improve plans related to smart waste management with the help of various system generated reports

***Area of the Project***

1. IoT
2. Android application

***Technologies used***

1. Arduino /Raspberry PI (Depending on feasibility)
2. Ethernet Shield
3. AWS Cloud Server
4. Google Maps (AI Optimized Path Findings)
5. End-User Application /Navigation App
6. Admin Panel Toolkit

## List of Figures

[Figure 1 Usecase Diagram Create Account 5](#_206ipza)

[Figure 2 Architecture Diageam 7](#_4k668n3)

[Figure 3 ERD 7](#_2zbgiuw)

[Figure 4 Level 0 DFD 8](#_1egqt2p)

[Figure 5 Level 1 DFD 8](#_3ygebqi)

[Figure 6 Class Diagram 9](#_2dlolyb)

[Figure 7 Activity Diagram Create Account 10](#_sqyw64)

[Figure 8 Sequence Diagram Create Account 11](#_3cqmetx)

[Figure 9 Collaboration Diagram 12](#_1rvwp1q)

[Figure 10 State Transition Diagram 13](#_4bvk7pj)

[Figure 11 Component Diagram 14](#_2r0uhxc)

[Figure 12 Deployment Diagram 15](#_1664s55)

## List of Tables

[Table 1 Functional Requirement Create Account 3](#_3q5sasy)

[Table 2 Usecase Create Account 6](#_25b2l0r)

**Table of Content**

[*Declaration* i](#_17dp8vu)

[*Statement of Submission* ii](#_3rdcrjn)

[*Dedication* iii](#_26in1rg)

[*Acknowledgement* iv](#_lnxbz9)

[*Abstract* v](#_35nkun2)

[List of Figures vii](#_2jxsxqh)

[List of Tables viii](#_z337ya)

[Chapter 1: Introduction to the Problem 1](#_3j2qqm3)

[1.1 Introduction 1](#_4i7ojhp)

[1.2 Purpose 1](#_qsh70q)

[1.3 Objective 1](#_2p2csry)

[1.4 Existing Solution 1](#_147n2zr)

[1.5 Proposed Solution 1](#_3o7alnk)

[Chapter 2: Software Requirement Specification 2](#_19c6y18)

[2.1 Introduction 2](#_kgcv8k)

[2.1.1](#_34g0dwd) Purpose 2

[2.1.2](#_1jlao46) Scope 2

[2.1.3](#_43ky6rz) Definitions, acronyms, and abbreviations 2

[2.2 Overall description 2](#_2iq8gzs)

[2.2.1](#_xvir7l) Product perspective 2

[2.2.2](#_3hv69ve) Product functions 3

[2.2.3](#_1x0gk37) User characteristics 4

[2.2.4](#_4h042r0) Constraints 4

[2.2.5](#_2w5ecyt) Assumptions and dependencies 4

[2.2.6](#_1baon6m) Apportioning of requirements 4

[2.3 Specific requirements 4](#_3vac5uf)

[2.3.1](#_2afmg28) Functional Requirement 4

[2.3.2](#_pkwqa1) Non-functional Requirements 4

[Chapter 3: Use Case Analysis 6](#_39kk8xu)

[Chapter 4: Design 8](#_1opuj5n)

[4.1 Architecture Diagram 8](#_48pi1tg)

[4.2 ERD with data dictionary 8](#_2nusc19)

[4.3 Data Flow diagram 9](#_1302m92)

[4.3.1](#_3mzq4wv) The level 0 9

[4.3.2](#_2250f4o) The level 1 10

[4.4 Class Diagram 11](#_haapch)

[4.5 Activity Diagram 11](#_319y80a)

[4.6 Sequence Diagram 12](#_1gf8i83)

[4.7 Collaboration Diagram 13](#_40ew0vw)

[4.8 State Transition Diagram 14](#_2fk6b3p)

[4.9 Component Diagram 15](#_upglbi)

[4.10 Deployment Diagram 16](#_3ep43zb)

[Chapter 5: User Manual 17](#_1tuee74)

[References 19](#_4du1wux)

[Appendix 23](#_2szc72q)

### 

### Chapter 1: Introduction to the Problem

##### Introduction

##### Traditional waste management system operates based on a daily schedule which is highly inefficient and costly. The existing recycle bin has also proved its ineffectiveness in the public as people do not recycle their waste properly. With the development of the Internet of Things (IoT) and Artificial Intelligence (AI), the traditional waste management system can be replaced with smart sensors embedded into the system to perform real-time monitoring and allow for better waste management.

##### Internet of things (IoT) is a communication paradigm that envisions a future paradigm where everyday life objects will be equipped with a microcontroller and some form of communication protocol. One well-known creation of IoT is the smart city, which can be defined as a city with smart technology, smart people, and smart collaboration. IoT shall visibly and effortlessly integrate many diverse end systems while providing open access to select subsets of data for the development with plenty of digital services. One major topic within the smart city is smart waste management. When it comes to waste management systems, the communication distance between the waste collection center and the waste collection point is a major factor in determining the system’s effectiveness. Studies in the field of wireless communication in IoT have also been accelerating. On The Contrary, communication technology such as Bluetooth, Wi-Fi, and Zigbee offer better data transmission rates, but these are limited by their data transmission ranges.

##### Improvement in the field of IoT has made it possible to improve the existing waste management system. Sensor’s implementation in the waste bin together with IoT connectivity allows for real-time monitoring, which is absent in the existing waste management system. Data such as filling level, temperature, and any necessary data can be gathered from the sensors. These data can then be transferred to the cloud for storage and processing. The processed data can then be used to study and access the limitation of the current waste management system and therefore enhances the system’s efficiency. IoT application in the waste bin is one step towards a smart city.

##### Domestic Waste Management has become a big challenge for the Metropolitan Authority hence this idea helps to minimize the Improper disposal and maintenance of domestic waste that could generate issues in public health and environment thus this system attempts to provide the practical solution towards managing the waste with the help of IOT Smart Things Model i.e., the trash is dumped into the bin stays for a long period causing environment pollution., fuel wastage, etc.

##### 

##### Purpose

We have seen several times the dustbins get overflowed and the concerned people do not get the information within a time and due to which filthy condition formed around the surroundings, at the same time bad odor spread out due to waste, bad condition of the streets which paves the way for air pollution and to some harmful diseases around the locality which is easily spreadable. So that we develop a system that uses the information collected from sensors to manage the waste collection. The dustbin is connected to the Internet wirelessly, as they are equipped with sensors that collect data about the level of collected waste in the dustbin. By this system, the dustbins do not get overflowed and the environment does not get polluted.

The main purpose of this application is to reduce human resources and efforts along with the enhancements of a smart city vision. At regular intervals, the dustbin will be squashed. Once these smart bins are implemented on a large scale, by replacing our traditional bins present today, waste can be managed efficiently as it avoids unnecessary lumping of wastes on the roadside. The system will consist of setting up smart waste bins/ trash cans per society, which will be IoT enabled. It will measure the fullness of the dustbins and update the status of each dustbin on the municipal server. It notifies them when the dustbin is full and provides the shortest route to empty all the dustbins based on the capacity of the municipal waste loading vehicles. The major advantage of this proposed system is it will stop the dustbin overflowing around the roadside and localities as smart bins are used in real-time.

With the rise of the population, the situation of cleanliness concerning waste management has become crucial. Waste management includes planning, collection, transport, treatment, recycle and disposal of waste together with monitoring and regulation. The existing waste management system, where the garbage is collected from the streets, houses, and other sites on a routine basis, is not able to effectively manage the waste generated.

In smart cities where environmental pollution is supposed to be significantly reduced, sanitation measures are essential, and cleanliness begins with providing garbage bins for waste disposal at strategic locations. The real-time monitoring and control of garbage bins placed in strategic locations and disposal at the destination is very essential.

The high increase in industrialization and human population resulted in higher levels of garbage generated in urban areas. Therefore, the number of garbage bins needs to be increased and placed in strategic locations for real-time monitoring and collection to save the environment. Without the proper collection and disposal of the waste, the garbage bins would be filled up or overfilled and spill out onto the surrounding, which can cause health-related problems to human beings and the environment

##### Objective

Waste management is one of the major environmental problems of Pakistani cities. The problem is the overflowing wastage on the roads. This, in terms, leads to various hazards such as bad odor & ugliness to that place which may be the root cause for the spread of various diseases. Several efforts have been invested in tackling. waste is an important issue, which needs to be tackled smartly. so that we make the proper monitoring system to control the waste management issues.

The major objectives of this proposed system are given below:

* Stop the dustbin overflowing around the roadside and localities.
* This can create a neat and clean environment while eliminating the overflow of wastage.
* According to the filled level of the smart waste dustbin, the vehicles will choose the shortest path which will save their time.
* Avoiding human intervention.
* Control lots of problems which disturbs the society in diseases.
* Low implementation cost.
* Resulting in a healthy and waste-ridden environment.
* Dustbins will be provided with an embedded device that helps in real-time monitoring.
* Cleaner Atmosphere.
* Efficient and effective Functioning.
* Automated Machine Based System
* The ultimate step to Time Saving
* Reduce the employee cost
* Pollution-free and stinking-free environment.

##### Existing Solution

The existing technologies aim to develop a smart waste management system using LoRa communication protocol and TensorFlow-based deep learning model. that sends the sensor data and Tensorflow performs real-time object detection and classification. The bin consists of several compartments to segregate the waste including metal, plastic, paper, and general waste compartment which are controlled by the servo motors. Object detection and waste classification are done in the TensorFlow framework with a pre-trained object detection model. This object detection model is trained with images of waste to generate a frozen inference graph used for object detection which is done through a camera connected to the Raspberry Pi 3 Model B+ as the main processing unit. The ultrasonic sensor is embedded into each waste compartment to monitor the filling level of the waste. GPS module is integrated to monitor the location and real-time of the bin. LoRa communication protocol is used to transmit data about the location, real-time, and filling level of the bin. RFID module is embedded for waste management personnel identification.

For an IoT-based solution to be implemented, it should be energy efficient, communicate, and share information across a particular parameter. The First embedded system that has been proposed in the market consists of GSM communication technology that is used as the platform to perform data transmission to the server. Web-based Android applications are developed to interface with a web server to provide information from sensors monitoring bin status, amount of waste in the bin, and time of waste collection. The data are processed by a graph theory optimization algorithm to obtain the shortest path for reaching the bin to efficiently manage the waste collection strategies. Graph theory optimization provides a very cost-efficient procedure to reduce the operation costs of a waste management system.

Waste management is a costly operation as it takes up a lot of resources and labor. Efforts have been taken by the authorities to improve waste management systems by setting up the recyclable bin and launching the 3Rs campaign (recycle, reuse and reduce). A study on public awareness of recycling activities in Kota Bharu, Kelantan Malaysia shows that only 31.8% of the total of 384 participants were involved in recycling. This shows both that the initiatives taken previously were not effective and that a smart waste management system needs to be developed to replace the existing infrastructures.

Advances in the field of IoT have made it possible to improve the existing waste management system. Sensor’s implementation in the waste bin together with IoT connectivity allows for real-time monitoring, which is absent in the existing waste management system. Data such as filling level, temperature, humidity, and any necessary data can be collected from the sensors. These data can then be transferred to the cloud for storage and processing. The processed data can then be used to study and access the limitations of the existing waste management system and therefore improve the system’s efficiency. IoT application in the waste bin is one step towards a smart city.

Classification of waste is a crucial step before the separation of waste can be performed. A deep learning method such as a convolutional neural network allows for the extraction of unique features from the image and then classifies them into each class with high accuracy. Tensorflow is an open-source, deep-learning library used for machine learning applications. It is capable of speech recognition, image classification, object detection, text classification, etc. With the intelligence gained from deep learning and an IoT, which integrates millions of smart devices together, the existing infrastructure for waste management systems can be improved.

The existing plan has many advantages, it is also credible enough to be implemented in every street of a developing nation. The advantages lie in its easy and valuable functioning. This will not only improve the streets we live in but also make its way for a better working system.

##### Proposed Solution

The implementation of a Smart Waste Segregation and Monitoring system in the bin aims to reduce the operating cost and improve the waste management system. At the same time, we are eager to develop the Smart City Architecture as it has been tested and executed in other Advanced Countries. In the future, the waste detection model is to be improved by increasing the number of waste images in the dataset to increase the flexibility of the system in identifying waste. Moreover, an automated routing system can be developed to identify and pinpoint the shortest path to the bin for the purpose of maintenance and helps the Employee to reach his desired location without wasting time and fuel. The existing waste management system can be improved and bring society towards a greener and healthier life.

In this scenario, garbage bins are installed with IoT Component (microcontroller). Each component consists of multiple sensors i.e.: Load Cell, and an Ethernet /Wi-Fi adapter interact with the Cloud Server to process the collected information. A database will be maintained containing the information about which dustbin to be placed in which area by their corresponding ids.

The collected data in the cloud will be analyzed through various platforms, and useful information regarding waste management will be extracted. From the collected data, users will get to know about real-time garbage levels and the corresponding waste location will appear to map if it is not empty, and the garbage collection van can find an optimized route for the collection of garbage. Whenever the garbage level crosses the threshold level, the alert will be generated for the urgent collection of garbage. The simple Web GUI will help the user to use this system efficiently.

**TECHNOLOGIES**:

⮚ Arduino UNO

⮚ Ethernet Shield(ESP 8266)

⮚ Google Maps (AI Optimized Path Finding)

⮚ End-User Application /Navigation App

⮚ Admin Panel Toolkit

## Chapter 2: Software Requirement Specification

##### 2.1 Introduction

###### 2.1.1 Purpose

1. The purpose of SRS is to analyze and understand the system in detail so that features and constituent components of at least one feasible solution are identified and documented.
2. This document will provide a roadmap to progressively transform the requirements through several stages into the final stage by describing the new structure of the system to be implemented.
3. The intended audience for the SRS is the buyers of industries who prefer to convert the normal waste Management system to IoT based system.
4. The idea of the following document is to excel in the waste management and industrial task that will succeed us in building a smart city infrastructure.

###### 2.1.2 Scope

This waste Management System using IOT has implemented the management of waste in real-time using a smart dustbin to check the fill level of the dustbin to check if it is full or not. This system also helps to monitor the fake reports and hence can reduce the corruption in the overall management system.

By implementing this system resource optimization, cost reduction, effective usage of smart dustbins can be done. Provide basic infrastructure, Quality of life, Clean and sustainable environment. Apply Smart Solutions.

###### 2.1.3 Definitions, acronyms, and abbreviations

1. Arduino /Raspberry PI
2. ES: Ethernet Shield (ESP 8266)
3. GM: Google Maps (AI Optimized Path Finding)
4. EUSR Application: End-User Application
5. APT: Admin Panel Toolkit
6. AWS Cloud Server: Amazon Web Services Cloud server

##### 

##### 2.2 Overall description

###### 2.2.1 Product perspective

This research aims to develop a smart waste management system and to achieve the efficiency and maintenance of the waste management sector that initiates the keeping of a clean and healthier environment. With the development of the Internet of Things (IoT) and Artificial Intelligence(AI), the traditional waste management system can be replaced with smart sensors embedded into the system to perform real-time monitoring and allow for better waste management.

• **System interfaces**

An Administrator Panel Toolkit will be required as the independent platform to monitor real-time working and simulation of the waste management task.

• **User interface**

Web-based Android applications are developed to interface with a web server to provide information from sensors monitoring bin status, amount of waste in the bin, and time of waste collection. The data are processed by a graph theory optimization algorithm

• **Hardware interfaces**

The hardware dependencies are deprived of a various embedded system that includes a basic Arduino UNO microcontroller along with ESP 8266 Standard Ethernet shield.

**Communications interfaces**

The standard ESP8266 Ethernet shield is required that allows it to communicate simultaneously with the help of **TCP/IP** and **UDP** protocols for the seamless transmission of data

• **Software interfaces**

The usage of System Base API is Required for the persistent monitoring of the System.

• **Memory**

The sensor retrieved data are store in the AWS cloud-based platform allows the secured and consistent backup of the data.

**Operations**

The host that will purchase the application for online availability of the Application Server will provide the facility of backup and recovery of the database if the application will crash.

• **Site adaptation requirements.**

The Map Driven mobile application will run on every version of the Android Platform. So does the APT (Admin Panel Toolkit) imply the same functionality while delivering the Interface to the user

###### 2.2.2 Product functions

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID: | **FR\_01** | | | | | | | |
| Name | **Account Login to Mobile Application** | | | | | | | |
| Description | Input | | Output | | Requirements | | Basic Work Flow | |
| Create account | Name, Email, Password, etc. | | Account created | | Internet Connectivity required | | Enter the correct information and click submit button. The system saves the record in the database | |
|  |  |  | |  | |  | |  | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID: | **FR\_02** | | | | | | | |
| Name | **Check Garbage Level** | | | | | | | |
| Description | Input | | Output | | Requirements | | Basic Work Flow | |
| Open  Mobile Application  Truck Driver | Regulate the  Level of  Smart bin | | Load Sensor | | Active system | | The sensor will generate a message whether it is filled or not. | |
|  |  |  | |  | |  | |  | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID: | **FR\_03** | | | | | | | |
| Name | **Checking Smart bin Status** | | | | | | | |
| Description | Input | | Output | | Requirements | | Basic Work Flow | |
| Observe the level of Garbage | Sensing the filling level of the bin. | | Load Sensor | | Active system | | These sensors send information and alert the processing system when a threshold limit is reached, which enables the processing system to generate an optimized schedule according to which the smart trucks will collect waste from these bins. | |
|  |  |  | |  | |  | |  | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID: | **FR\_04** | | | | |
| Name | **Connecting the nearest Smart bin** | | | | |
| Description | Input | | Output | Requirements | Basic Work Flow |
| Searching for the nearest Smart Bin | Accepting the receiving request from the bin | | Bin Location appears on the map | Active system | If the bin has reached its limit on the system then it will appear on the map |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID: | **FR\_05** | | | | |
| Name | **Fetching the Smart bin Location** | | | | |
| Description | Input | | Output | Requirements | BasicWork Flow |
| Retrieving the GPS pinpoint of the smart bins. | Check the following level and assign the GPS location accordingly | | If the desired condition is met, then display the location on the map | Mobile Phone/GPS Sensor | The collected dataset allows the Driver Application to find the shortest route for the vehicle to track the Waste Can which are marked as a pinpoint on the Map. Only filled waste can are labeled as traceable locations on the Map. |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID: | **FR\_06** | | | | |
| Name | **Interacting with AWS cloud** | | | | |
| Description | Input | | Output | Requirements | Basic Work Flow |
| Retrieving information from the sensor | Collect Information from the sensors | | Gathering the sensor information to store it to a cloud server | Wi-Fi/Ethernet  Shield | The microcontroller directly interacts with the AWS Cloud platform to store the collected information |
|  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID: | **FR\_07** | | | | | | | |
| Name | **Assigning GPS locations** | | | | | | | |
| Description | Input | | Output | | Requirements | | Basic Work Flow | |
| Identifying the smart bin locations the map | A GPS receiver in your phone attends for these signals | | Bin Location on the map | | Wi-Fi | | It works by measuring the distance to each pinpoint location with the help of a pathfinding algorithm | |
|  |  |  | |  | |  | |  | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID: | **FR\_08** | | | | | | | |
| Name | **Making the System Adaptive for Maintenance** | | | | | | | |
| Description | Input | | Output | | Requirements | | Basic Work Flow | |
| Adding the basic functionality of version upgrade if required in the future | Collecting the data suitable for making changes in the system | | Verifying the  System Changes | | Desktop Application (APT) | | This helps to support the maintenance and up-gradation of the system | |
|  |  |  | |  | |  | |  | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID: | **FR\_09** | | | | |
| Name | **Admin Panel Toolkit** | | | | |
| Description | Input | | Output | Requirements | Basic Workflow |
| Real-time monitoring of IoT components | Receiving the sensor data | | Taking Control of  System Components | Desktop | Enter the correct information to log in to the APT(admin Panel Toolkit) |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID: | **FR\_10** | | | | |
| Name | **Making a System Flexible for Update** | | | | |
| Description | Input | | Output | Requirements | Basic Workflow |
| Add Upgrading feature to  the APT | Make Changes to Existing components | | Taking Control of  System Components | Desktop | The System will able to handle the changes made  In the System |
|  |  |  |  |  |  |

**2.2.3** **User characteristic**

The user should be able to understand the basics of Google Maps and GPS tracking methodologies to interact with the Smart Bin Tracking System.

###### 2.2.4 Constraints

This subsection of the SRS should provide a general description of any other items that will limit the developer’s options. These include

* The user must have an active internet connection on a smart device.
* The Smart bins must have a feasible Ethernet /GSM connection respectively to interact with the other neighboring devices.
* The Admin Panel Supports the usage of the Desktop Application while having the stable internet connection to interact with the Cloud server and Service Components
* The system supports the flexibility of the Version Control System for Maintenance and Redesigning purposes
* Smart Bins must follow the desired conditions to be detected by the Bin Tracking System

###### 2.2.5 Assumptions and dependencies

· User must have a stable internet connection

· User Application supports every version of the Android OS

· The service is used on a desktop, laptop, or mobile.

###### 2.2.6 Apportioning of requirements

The android application will be developed which will available on the play store.

##### 2.3 Specific requirements

This section will describe the functional and non-functional requirements of the System at a sufficient level of detail for the designers to design a system satisfying the User requirements and tests to verify that the system satisfies the requirements.

###### 2.3.1 Functional Requirement

* Login to Driving Portal Application
* Determining the Garbage Level according to preset Value.
* Check the status of Smart bin and the System will Perform Accordingly
* Communicating with nearest Smart bin via ESP8266
* Fetching the location of the Smart bins through AWS cloud
* Selecting the AWS server to Interact with the IoT Component
* Assigning the GPS Pinpoint Location
* Calculating the shortest route Map
* Selecting the Suitable Language Platform for the APT(Admin Panel Toolkit).
* Making the System Flexible for System Maintenance and Update.

##### Non-Functional Requirement

###### Usability

From the usability perspective, our system is as easy to use as it has the simplest

Interface. We have followed the design according to the Human-Computer Interaction perspective.

###### Reliability

The app is reliable because we will provide maintenance and recovery for error

###### Performance

The system response time will be good by which the user will get efficient results.

###### 

###### Design Constraints

Language being Used: java

I Interface: sensor system, GUI, GPS

Client: mobile application user

Development Tools: android studio, Arduino Circuit

Programming Language: XML, java

###### Portability

The application should be able to run on any android mobile phone.

###### 

###### Maintainability

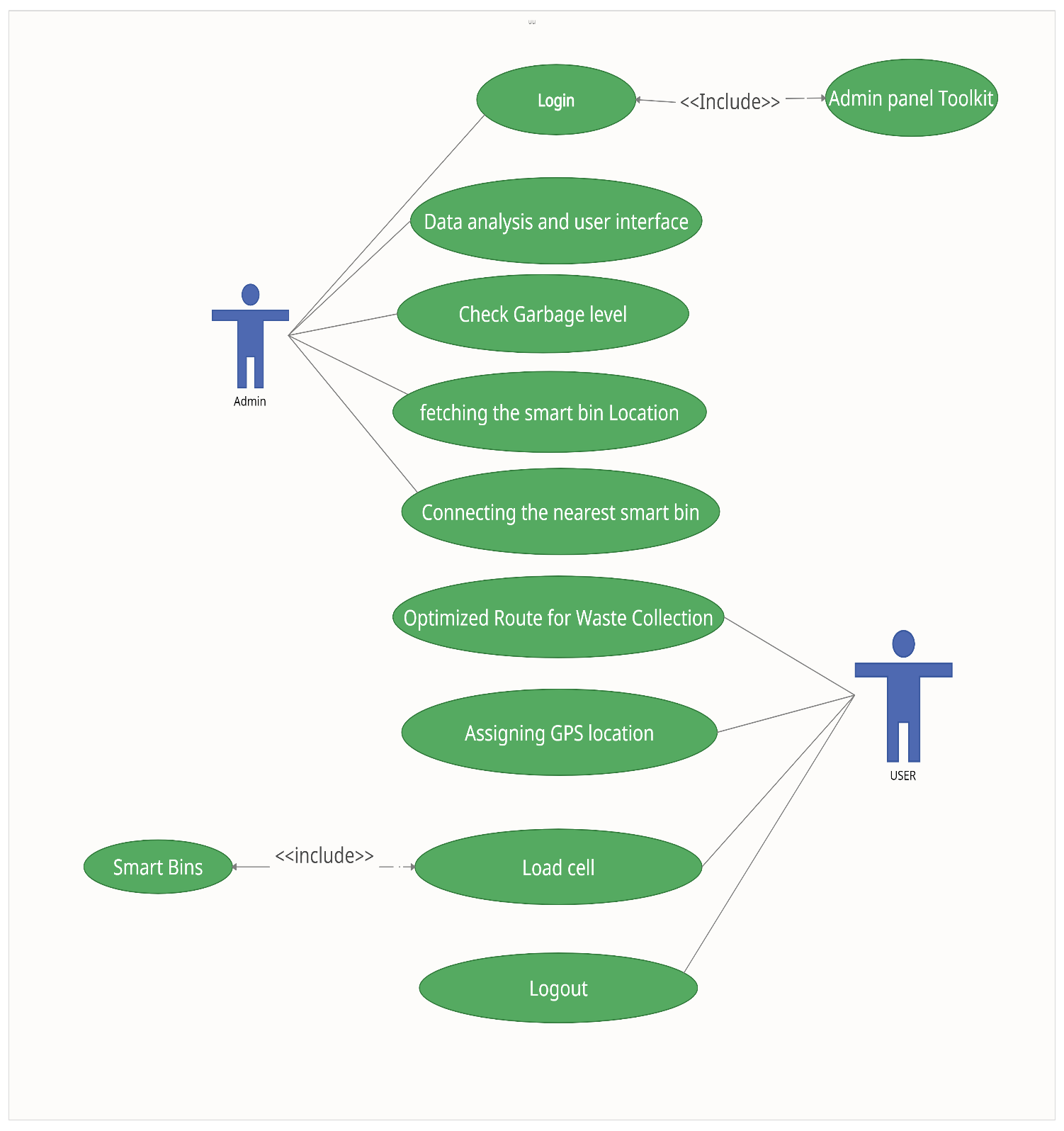
For maintenance purposes, the members are available. They ensure the changes made after launching the first version. The app will be updated with new features and extended versions in the future.

###### License Agreement

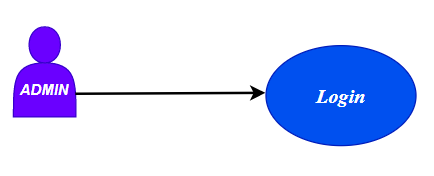
The patent for this system should define the copyrights of this system, ethically.

the system will be highly secure and will not share any user information with others.

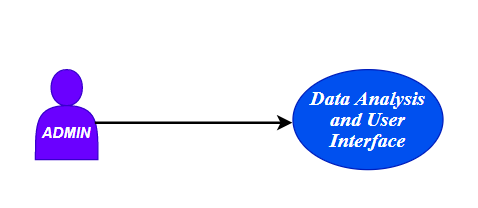
## Chapter 3: Use Case Analysis

****

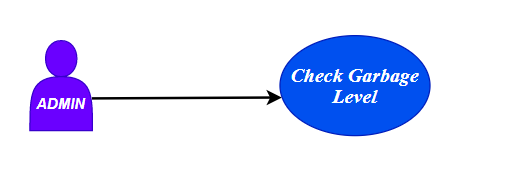
**Use Case Diagram Detail**



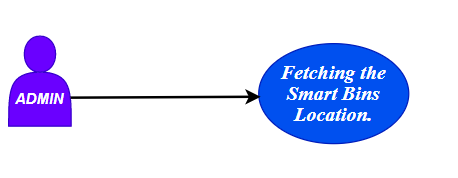
|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_01** | |
| Use Case Name | **Login** | |
| Description | Users will log in and will redirect to the panels | |
| Primary Actor | Admin | |
| Secondary Actor | Staff | |
| Pre-Condition | The user must have an active internet connection | |
| Post-Condition | The user account will be created | |
| Basic Flow | Actor Action | System Action |
|  | The user will enter the name, e-mail address and set the password | The system will add the username and password into the database and register the account of the Admin Panel Account |
| Alternate Flow | if the wrong password is entered by the user, he will not gain access to the system | |



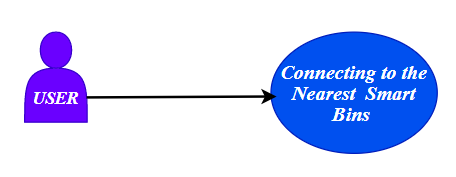
|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_02** | |
| Use Case Name | **Data Analysis and User Interface** | |
| Description | Various sensors and Cloud Platform will be used for the Assessment  Of sensor-based information | |
| Primary Actor | Admin | |
| Secondary Actor | Staff | |
| Pre-Condition | The corresponding sensors should be installed and synchronized properly to the system. | |
| Post-Condition | The GPS Driving App will be able to track the smart bins. | |
| Basic Flow | Actor Action | System Action |
|  | The user ensures the working and performance of the system whether it is adaptive to the system. | The system should be able to compile the information that has collected from the system successfully |
| Alternate Flow | N/A | |



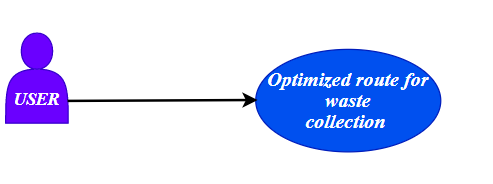
|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_03** | |
| Use Case Name | **Check Garbage Level** | |
| Description | Observe the level of Garbage | |
| Primary Actor | Load cell | |
| Secondary Actor | Driving Application | |
| Pre-Condition | Check whether the level of the bin is either empty or not | |
| Post-Condition | If the condition is satisfied, then the Bin Location appears on the map. | |
| Basic Flow | Actor Action | System Action |
|  | The user will Observe  the Garbage level | The sensor will generate a message whether it is filled or not |
| Alternate Flow | NA | |



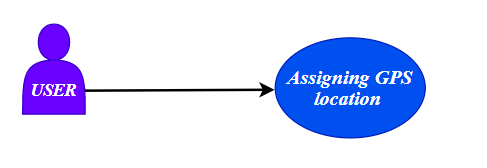
|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_04** | |
| Use Case Name | **Fetching the Smart Bins Location.** | |
| Description | The GPS coordinates and the information regarding the location of the bin should be communicated to the system | |
| Primary Actor | Smart Bin | |
| Secondary Actor | Driving Application | |
| Pre-Condition | Check whether the level of the bin is either empty then pass its location through the cloud server | |
| Post-Condition | If the condition is satisfied, then the Bin Location appears on the map. | |
| Basic Flow | Actor Action | System Action |
|  | The user will be able to track the Smart bin for the nearest location. | The GPS coordinates of the bin help to find the shortest path on the map. |
| Alternate Flow | NA | |



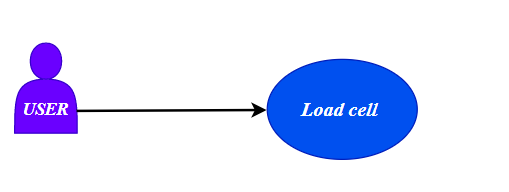
|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_05** | |
| Use Case Name | **Connecting to the Nearest Smart Bins** | |
| Description | The Ethernet Shield Installed to the circuit helps the Smart to interact with  the Nearest Bin location to form up the shortest distance on the map | |
| Primary Actor | Smart Bin | |
| Secondary Actor | Driving Application | |
| Pre-Condition | The Arduino along with its dedicated Communication link finds the nearest located Waste Component. | |
| Post-Condition | The Map will able to distinguish between one or more Smart bins. | |
| Basic Flow | Actor Action | System Action |
|  | The user will be able to track the Smart bin for the nearest location. | The GPS coordinates of the bin help to find the shortest path on the map. |
| Alternate Flow | NA | |



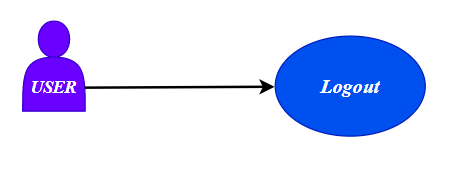
|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_06** | |
| Use Case Name | **Optimized route for waste collection** | |
| Description | The user will select the optimized route for waste collection | |
| Primary Actor | Application user | |
| Secondary Actor | NA | |
| Pre-Condition | The user successfully login into the application | |
| Post-Condition | The user perceives the google map in the application | |
| Basic Flow | Actor Action | System Action |
|  | The user will enter the direction toward the bin | The system will show the optimized route toward the bin |
| Alternate Flow | NA | |



|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_07** | |
| Use Case Name | **Assigning GPS location** | |
| Description | The system will assign the GPS location of the user | |
| Primary Actor | Application user | |
| Secondary Actor | NA | |
| Pre-Condition | the user has an internet connection | |
| Post-Condition | GPS catches the current location of the user | |
| Basic Flow | Actor Action | System Action |
|  | The user will open the Google map through the App | The system will show the location of the user on a Google map |
| Alternate Flow | NA | |



|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_08** | |
| Use Case Name | **Load cell** | |
| Description | The sensor tells the level of waste in the bin in the app | |
| Primary Actor | Application user | |
| Secondary Actor | NA | |
| Pre-Condition | The user must be connected to the internet | |
| Post-Condition | The sensor is attached to the bin | |
| Basic Flow | Actor Action | System Action |
|  | The user will see the level of waste in the bin through the app | The system will show the bin which has waste in the App |
| Alternate Flow | NA | |



|  |  |  |
| --- | --- | --- |
| Use Case ID | **UC\_09** | |
| Use Case Name | **Logout** | |
| Description | The user can logout from the system | |
| Primary Actor | Application user | |
| Secondary Actor | NA | |
| Pre-Condition | the user must be signed in | |
| Post-Condition | The user sign-out successfully | |
| Basic Flow | Actor Action | System Action |
|  | The user will click on the sign-out button | The system will destroy the session |
| Alternate Flow | NA | |

# 

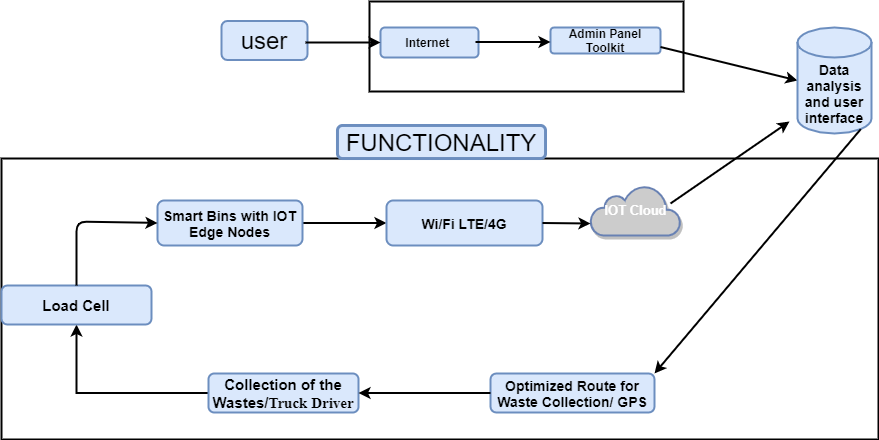
# Chapter 4: Design

In this section, we provide the design analysis of our modules including the following designs

1. Architecture Diagram
2. ERD with data dictionary
3. Data Flow diagram
4. Class Diagram
5. Activity Diagram
6. Sequence Diagram
7. Collaboration Diagram
8. State Transition Diagram
9. Component Diagram
10. Deployment Diagram

#### 4.1 Architecture Diagram

Define the graphical representation of the concepts, their principles, elements, and components that are part of your project.



*Figure 1 Architecture Diagram*

#### 4.2 ERD with data dictionary

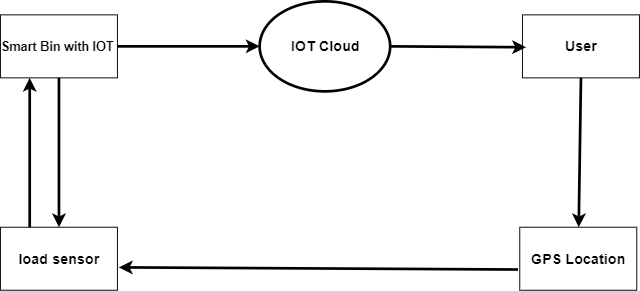
EntityRelationshipDiagram with complete relations with dependencies of your project

#### 4.3 Data Flow diagram

The data flow diagram includes two levels

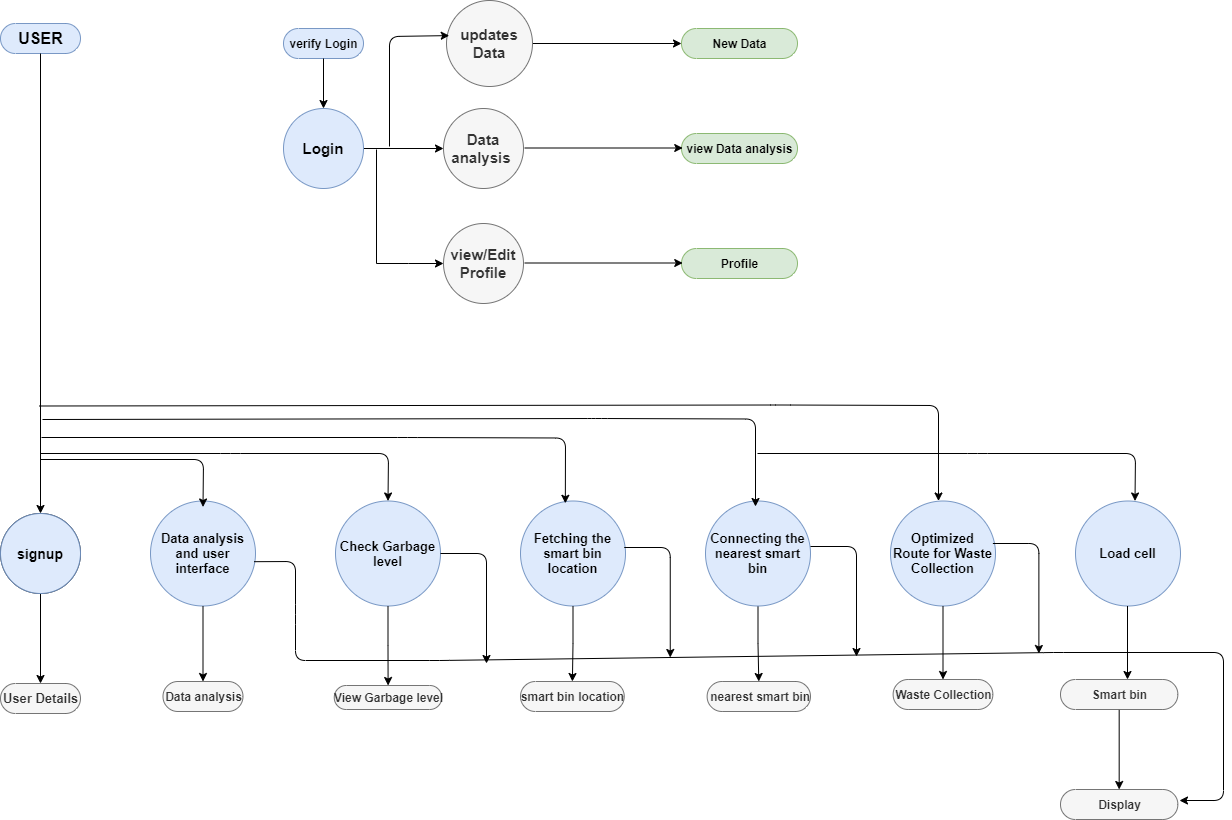
###### 4.3.1 The level 0

The flow of information inside the system is defined at this level

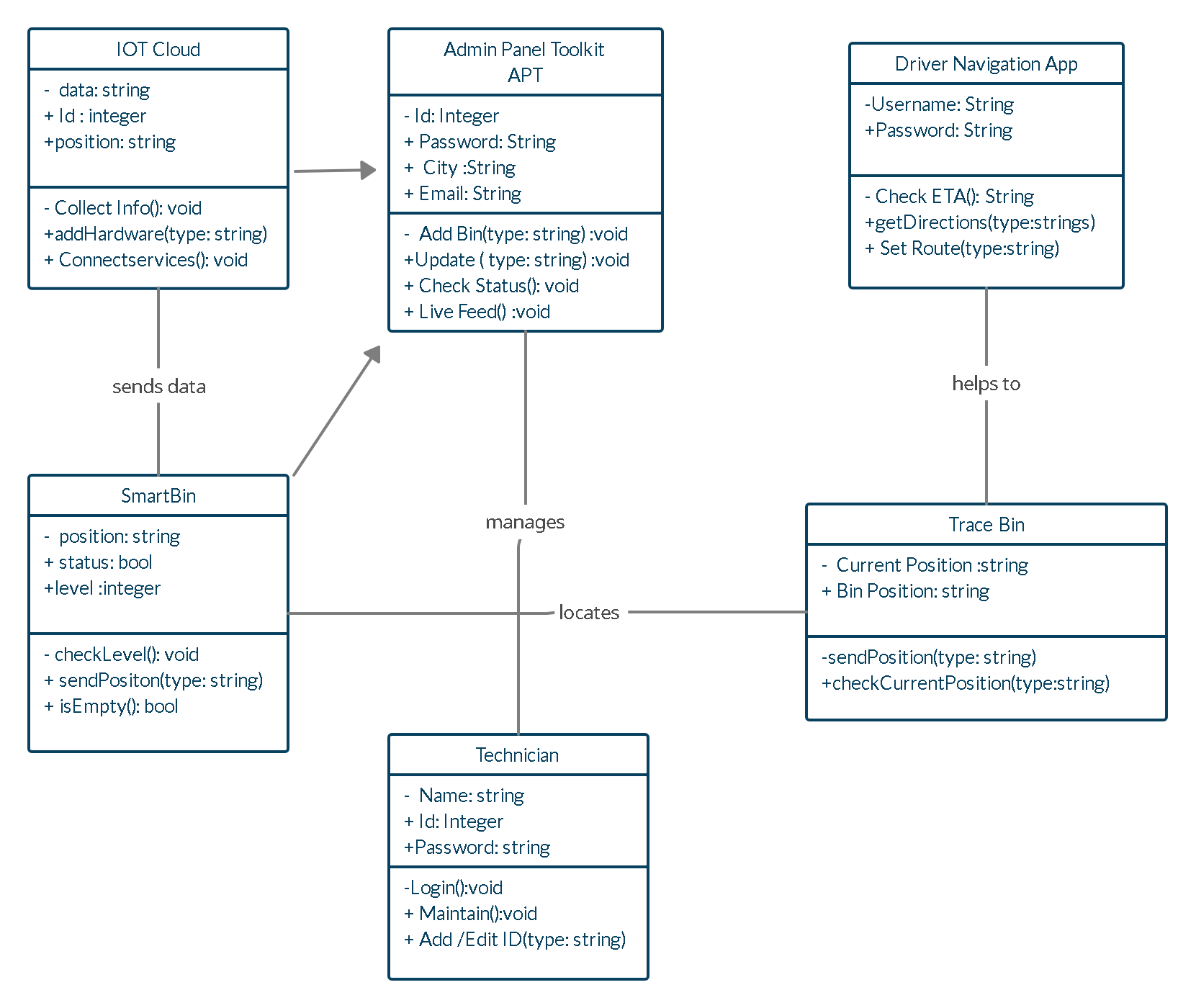


###### 4.3.2 The level 1

The flow of information outside the system is defined in this level



#### Class Diagram

Describe the structure of a project by showing the systems classes, their attributes, operations (or methods), and the relationships among objects.

#### Activity Diagram

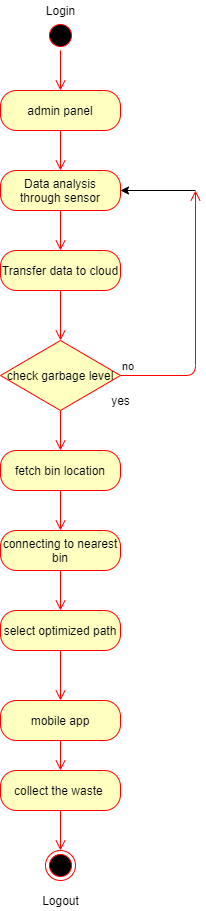
This diagram includes all the activity diagrams of the functional requirements of your project along with the aggregated activity diagram

The Activity diagrams can be made by using Visual Paradigm

1. Select **Diagram > New** from the application toolbar.
2. In the **New Diagram** window, select **Activity Diagram**.
3. Click **Next**.
4. Enter the diagram name and description.
5. Click **OK**.

For example, yours create account activity looks like this and you have to follow this template for writing your activity diagrams

Activity diagram for create account



#### 4.6 Sequence Diagram

This diagram includes all the Sequence diagrams of the functional requirements of your project along with the aggregated Sequence diagram

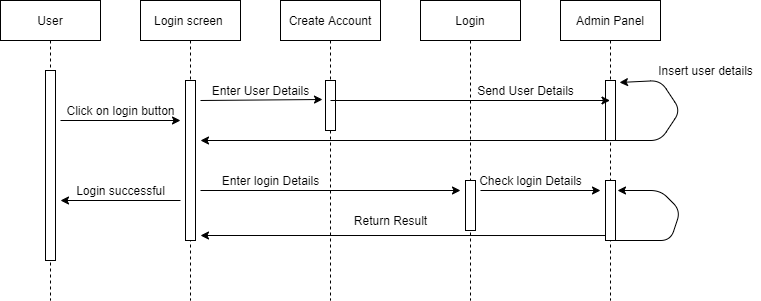
The Sequence diagrams can be made by using Visual Paradigm

1. Select **Diagram > New** from the application toolbar.
2. In the **New Diagram** window, select **Sequence** **Diagram**.
3. Click **Next**.
4. Enter the diagram name and description.
5. Click **OK**.

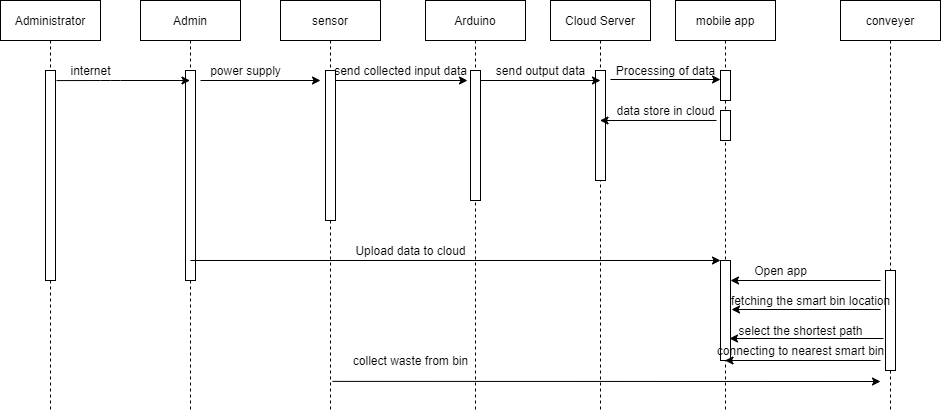
For example, yours create account Sequence look like this and you have to follow this template for writing your Sequence diagrams

Sequence diagram for create account

**Register and login:**



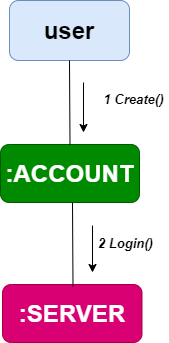
**Usage:**

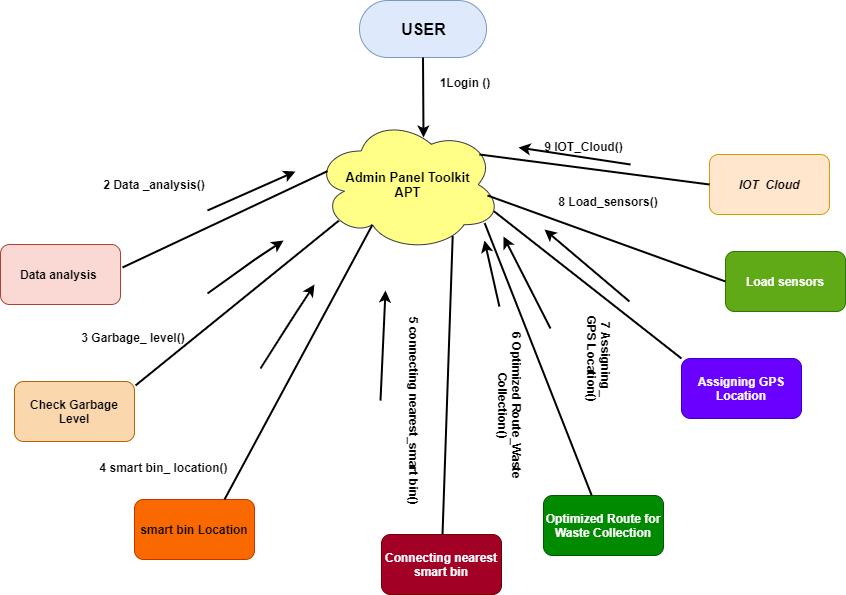


#### 4.7 Collaboration Diagram

It shows the object organization as shown below. Here in collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram.

**Register and login:**



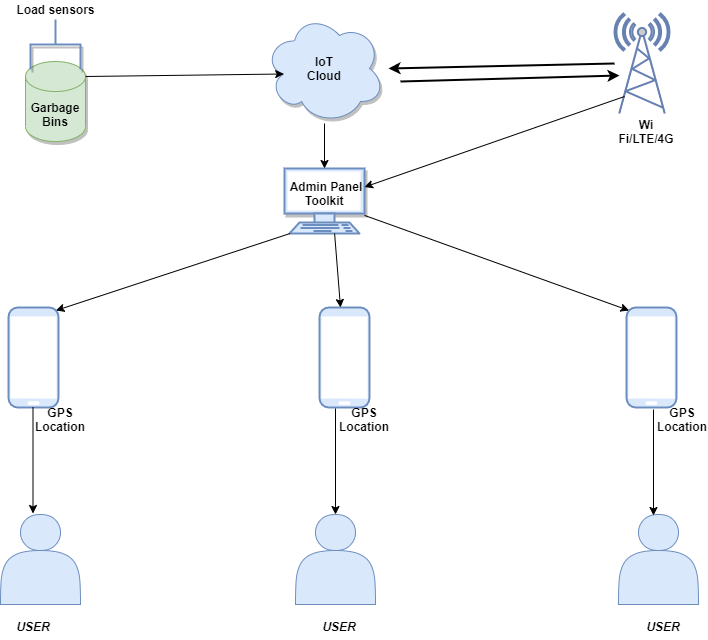
**Usage:**

#### Deployment Diagram

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hardware’s used to deploy the application.

Deployment diagrams are useful for system engineers. An efficient deployment diagram is very important because it controls the following parameters

1. Performance
2. Scalability
3. Maintainability
4. Portability

****