-

**Project Report**

on

**IoT Assisted Food Donation and Waste Management System**

Submitted by

**Project Members**


| | |
|---|---|
| 1032200260 | Fahad Ahmed Malik |
| 1032201997 | Avinash Shelukar |
| 1032212257 | Pruthu Prabhudesai |
| 1032202149 | Ruchik Alhat |

**Under the Internal Guidance of**

**Dr. Vinayak Musale**

**Department of Computer Engineering and Technology**

**Dr. Vishwanath Karad MIT World Peace University, Kothrud,**

**Pune 411 038, Maharashtra - India**


**2023-2024**

MIT-WPU

Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

# SCHOOL OF COMPUTER ENGINEERING AND TECHNOLOGY

# C E R T I F I C A T E

This is to certify that

Fahad Ahmed Malik
Avinash Shelukar
Pruthu Prabhudesai
Ruchik Alhat

of BTech.( Computer Science & Engineering) have completed their project titled "*IoT Assisted Food Donation and Waste Management System*" and have submitted this Capstone Project Report towards fulfillment of the requirement for the Degree-Bachelor of Computer Science & Engineering (BTech-CSE) for the academic year 2023-2024.

**Dr. Vinayak Musale**
Project Guide
Department of CET
MIT World Peace University, Pune

**Dr. Vrushali Kulkarni**
Program Head
School of CET
MIT World Peace University, Pune

Internal Examiner: Dr. Vinayak Musale
External Examiner:...........................................

**Date: 09-12-2023**

# Acknowledgement

We would like to express our sincere gratitude and appreciation to everyone who contributed to the development of this multi-purpose chatbot framework for intelligent assistance and collaboration.

First and foremost, we would like to thank our project guide for providing us with the resources, guidance, and support to complete this project successfully. We also want to thank the members of the project team for their hard work, dedication, and commitment to delivering a high-quality product.

We extend our thanks to the subject matter experts who provided valuable insights and feedback throughout the project. We also acknowledge the contributions of the testing team, whose thorough testing and feedback helped ensure the framework's quality and reliability.

Finally, we would like to express our appreciation to our families, friends, and colleagues who provided us with the support, encouragement, and motivation to complete this project.

Thank you all for your invaluable contributions to this project's success.

| | |
|---|---|
| Fahad Ahmed Malik | 1032200260 |
| Avinash Shelukar | 1032201997 |
| Pruthu Prabhudesai | 1032212257 |
| Ruchik Alhat | 1032202149 |

# Abstract

The creation and deployment of an Internet of Things (IoT)-assisted food donation and trash management web application are presented in this project report, with an emphasis on using IoT technology for food quality inspections. The pressing need to tackle the interrelated issues of food poverty and food waste is highlighted by the growing worldwide problem of food waste. The web application with IoT assistance is a comprehensive solution that makes it easier for donors to donate excess food to recipients while also guaranteeing the quality and safety of donated food products with the help of IoT-enabled monitoring.

Donors may evaluate if food products are suitable for donation by using the online application that incorporates IoT sensors to monitor vital food quality metrics in real-time, such as temperature, humidity, and environmental factors. With tools for scheduling, monitoring, and managing the donation process, the program also facilitates collaboration and communication between food donors and recipient organizations. Furthermore, by enabling users to track and evaluate their food waste, the online application raises awareness and encourages preventative actions to reduce waste.

This project report provides a comprehensive overview of the technical implementation, user interface design, and functionality of the IoT assisted web application. Through a detailed analysis, the report elucidates the potential impact of IoT-assisted food donation and waste management in addressing food waste and contributing to food security and sustainability. The innovative use of IoT technology for food quality checks holds the promise of revolutionizing food donation and waste management practices, offering a scalable and impactful solution to mitigate food waste and alleviate food insecurity.

**Keywords-** *Data Analytics, Food Quality Monitoring, Food Redistribution, Real-time Monitoring, Supply Chain Management, Web Application Development*

# List of Figures

# List of Tables

# Contents

# Chapter 1
# Introduction

## 1.1. Project Statement

Using IoT technology to monitor food quality and expedite the distribution of leftover food is the goal of the IoT Assisted Food Donation and Waste Management Web Application project. The initiative intends to enable real-time monitoring of temperature and freshness by integrating IoT sensors across the food supply chain. The goal of this project is to develop an intuitive online application for managing and interpreting Internet of Things data, maximizing food redistribution to minimize waste and alleviate poverty in local communities. Its ultimate goal is to reduce food waste as much as possible while effectively supplying excess food to those in need.

The need for creative solutions that make use of cutting-edge technologies is driven by the growing worldwide concern for waste management, environmental sustainability, and philanthropic giving. An app for donation and waste management using IOT technology. With serious consequences for the environment, society, and economy, food waste is a major worldwide problem. Nonetheless, millions of people experience hunger and food insecurity globally. In order to close this gap and solve this issue, food donation and trash management require creative IoT (Internet of Things) solutions. The way trash is managed and food is donated now is frequently ineffective, opaque, and underutilized contemporary technologies.

This project aims to develop an innovative IoT system that can track, manage, and optimise food donation and waste management processes.

## 1.2. Area

The online application for IoT-assisted food donation and trash management has several possible uses and may be integrated in other domains, such as:

1. **Food Banks and Charitable Organizations:** Food banks and other charity

organizations can use the online tool to maintain inventory, organize food contributions, and arrange delivery to underprivileged individuals and communities.

2. **Restaurants and Food Service Providers:** Restaurants and food service providers may track and manage extra food, maximize food use, and give excess food to charity organizations by using this web tool.

3. **Grocery Stores and Supermarkets:** Supermarkets and grocery retailers may utilize the online application to optimize inventory management, monitor and control food waste, and donate excess food to charity groups.

4. **Food Manufacturers and Processors:** The web application can be used by food manufacturers and processors to monitor and manage food quality, optimize production processes, and donate excess food to charitable organizations.

5. **Government and Municipalities:** Governments and local governments may utilize the online tool to monitor food waste, control it, and put sustainable food management practices and regulations into place.

6. **Non-Governmental Organizations (NGOs):** NGOs may use the online tool to track distribution, keep an eye on food contributions, and work with other groups to coordinate efforts to optimize the effect of food donation programs.

7. **Educational Institutions:** Educational institutions may maximize food consumption, track and handle food waste, and give extra food to nonprofits by using this online tool.

In general, there are many possible uses for the IoT-assisted food donation and waste management web service, and it may be applied in a variety of contexts to address the issues of food waste, food poverty, and sustainable food management techniques.

## 1.3. Project Introduction and Aim

Roughly one-third of all food produced for human consumption is wasted annually, making food waste a major worldwide problem. Millions of individuals experience malnutrition and hunger at the same time. The application of Internet of Things (IoT) technology to food donation and waste management has emerged as a viable way to address this pressing issue. With an emphasis on the application of IoT for food quality inspections, we provide a thorough study and implementation of an IoT-assisted food donation and waste management web application in this project report.

By creating a link between food donors—such as restaurants, grocery shops, and private citizens—and food banks or nonprofit groups, the IoT-assisted food donation and waste management web application seeks to enable the distribution of excess food to those in need. The improvement of the efficiency and dependability of the food donation process is contingent upon the deployment of Internet of Things (IoT) technology to guarantee the food products' quality and safety.

The integration of IoT devices for real-time monitoring of food quality criteria is highlighted in this project report, which offers a thorough description of the web application's development and functioning. The online application facilitates the continuous monitoring of environmental parameters that affect food safety, including temperature, humidity, and other aspects, by utilizing Internet of Things sensors. Food donors may now evaluate the quality of their gifts and decide whether or not the food products are suitable for donation with the help of this real-time data gathering and analysis.

In addition, by providing tools for monitoring, scheduling, and tracking the donation process, the online application for food donation and trash management, aided by the Internet of Things, streamlines coordination and communication between food donors and recipient organizations. Aside from providing tools for users to track and evaluate their food waste, the online program also encourages awareness-raising and preventative actions to reduce waste.

We explore the UI design, the general functionality of the web application, and the technical elements of IoT implementation in this project report. We also talk about how food waste may be addressed and how food security and sustainability can be enhanced by IoT-assisted food donation and waste management.

The primary aim is multifaceted:

1. **Enhancing Food Quality Assessment:** The aim of the project is to set up an advanced system that can check food quality continuously and in real time. It seeks to lower the dangers associated with ingesting tainted or expired food by using IoT sensors to increase accuracy and reliability in determining the quality and shelf-life of food products.

2. **Optimizing Donation Processes:** The simplification of the food donation procedure is one of the main goals. Identifying and redistributing excess food to nonprofits or areas experiencing food poverty will be made easier with the help of an Internet of Things-assisted system. This process of optimization guarantees that food that is considered fit for human consumption but is in excess of what is needed for commercial purposes is effectively distributed to interested parties.

3. **Minimizing Food Waste:** The initiative aims to reduce food waste by using IoT technologies for accurate monitoring and analysis. This decline is in line with more general sustainability objectives, which seek to lessen the negative effects of food waste on the environment, such as greenhouse gas emissions and resource depletion.

4. **Promoting Transparency and Efficiency:** A centralized platform for users to monitor, analyze, and manage data gathered from IoT sensors is what the web application development component of this project aims to deliver. In order to guarantee openness and effectiveness in the decision-making processes, this platform will include features like inventory management, predictive analytics, data visualization, and automatic warnings.

Essentially, the goal of this initiative is to close the gap that exists between the abundance of surplus food and the impoverished people' urgent need for proper nourishment. The project aims to develop a sustainable, effective, and socially conscious system that maximizes food resources while tackling societal issues associated with hunger and food waste by fusing IoT technology with food management techniques.

## 1.4. Background

In recent years, there has been a pressing need to address the worldwide issues of food waste and food poverty. Traditional food management systems still have shortcomings when it comes to distributing surplus and evaluating quality, even with technology developments. There is an urgent need for creative solutions since 1.3 billion tons of food are wasted every year.

Businesses utilizing IoT technology have expanded rapidly, providing networked devices that gather and distribute data. Taking advantage of this chance, we want to transform food management by utilizing IoT's potential. The goal of our endeavor is to create a reliable system for data analysis in real time by utilizing Internet of Things (IoT) sensors to monitor important characteristics such as temperature and freshness.

Connecting the dots between the availability of excess food and the demand for wholesome nourishment is the IoT Assisted Food Donation and Waste Management Web Application. This allows for the efficient management of inventories and the timely redistribution of food to those in need by integrating IoT sensors across the food supply chain. Also, by lessening food waste's negative effects on the environment, this program supports sustainability objectives.

By tackling food insecurity and reducing the negative impacts of food waste, this initiative, which operates at the intersection of technology, sustainability, and humanitarian activities, seeks to create a more just and sustainable future.

## 1.5. Motivation

The IoT Assisted Food Donation and Waste Management Web Application was developed as a result of the confluence of urgent global issues with the ability to be solved by cutting edge technology.

1. **Food Waste Epidemic:** There is a sense of urgency due to the startling figures on food waste. Millions of people go hungry every year, yet every year around one-third of the food produced for human use is wasted. This discrepancy has made it clear that effective, scalable solutions are required to close the gap between the availability of surplus food and the unfulfilled nutritional needs of disadvantaged groups.

2. **Technological Advancements:** IoT technology's quick development and broad uptake offer a rare chance to completely transform traditional food management techniques. The Internet of Things (IoT) is a potent tool for tracking and optimizing many areas of the food supply chain because of its capacity to gather real-time data through networked sensors.

3. **Sustainability Imperative:** In addition to being a social responsibility issue, addressing food waste is essential to sustainability initiatives. The need for efficient waste reduction techniques is highlighted by the negative effects food waste has on the environment, including greenhouse gas emissions and resource depletion from its production and disposal.

4. **Humanitarian Focus:** This initiative is fundamentally motivated by humanitarianism. Its main goal is to make sure excess food that would otherwise go to waste is effectively distributed to people who are food insecure. The program seeks to improve the quality of food screenings and streamline the donation procedure in order to positively impact underprivileged areas.

5. **Optimizing Resource Utilization:** The initiative intends to maximize resource use in the food supply chain, in addition to alleviating hunger and minimizing waste. It aims to reduce overproduction, improve inventory control, and increase overall resource efficiency by putting IoT technology to use for real-time monitoring and analysis.

6. **Creating Synergy Between Technology and Social Impact:** This effort is a prime example of how technology can be used to solve issues in the real world by fusing technological innovation with a significant social effect. It embodies the philosophy

that innovation should be used to improve society as well as for financial gain or efficiency advantages.

The IoT Assisted Food Donation and Waste Management Web Application is primarily driven by its potential to revolutionize the way that food is currently managed. Through the utilization of Internet of Things technology, the project aims to improve food quality inspections, optimize donation procedures, and reduce waste in order to make a significant impact on sustainability, social justice, and public health.

## 1.6. Challenges

To ensure its efficacy and broad acceptance, a number of obstacles must be overcome throughout the development and deployment of an Internet of Things-assisted food donation and trash management web service.

1. **Technology Integration:** Planning and compatibility issues must be taken into account when integrating IoT sensors and devices into the current food donation and trash management infrastructure. It is quite technical to ensure smooth interface with various kinds of sensors, data gathering tools, and backend systems.

2. **Data Security and Privacy:** Data security and privacy are issues brought up by the real-time data collecting and transmission from IoT devices. Protecting private data, such food quality information and donor histories, from prying eyes and security lapses is a major task.

3. **Scalability and Interoperability:** Ensuring scalability and interoperability across many platforms, devices, and geographical locations is a hard effort, since the web application seeks to serve a broad spectrum of food givers and recipient organizations.

4. **User Adoption and Training**: User education and familiarization with IoT technology are essential for the web application to be successfully adopted by food givers and receiving organizations. For the platform to be used effectively, users must get thorough training and be able to overcome any possible resistance to change.

5. **Regulatory Compliance:** Respecting rules pertaining to privacy, data protection, and food safety is essential. One of the biggest challenges in the regulatory compliance environment is navigating its complexity across several areas and countries.

6. **Maintenance and Support:** It is difficult to allocate resources sustainably over the long run when it comes to IoT device dependability and continuous operation, not to mention constant technical assistance for consumers.

7. **Cost Considerations:** The creation and upkeep of the web application, together with the deployment of IoT sensors and devices, come at a substantial expense. A major difficulty is making sure that all stakeholders can afford it while also balancing the initial expenditure with the long-term advantages.

In order to fully utilize the IoT-assisted food donation and waste management web application and enable it to successfully reduce food waste, improve food security, and advance sustainable food management practices, it will be imperative to address these problems.

# Chapter 2
# Literature Survey

Using the creation of a mobile application, Hajjdiab, Anzer, Tabaza, and Ahmed's research paper "A Food Wastage Reduction Mobile Application" addresses the urgent problem of food waste and was presented at the 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW) in Barcelona, Spain. The writers of this study acknowledge that cutting down on food waste is an important part of managing resources sustainably. A smartphone application is presented with the goal of encouraging users to reduce food waste. The characteristics and capabilities of this smartphone software, as well as its possible influence on increasing awareness and lowering food waste—a serious worldwide issue—will probably be covered in the article. Given the larger context of the Internet of Things (IoT) and cloud computing, the research probably examines the application's design, usability, and integration with cloud technology. Readers may learn more about the possibilities of technologically driven solutions to reduce food waste, a major worldwide issue with substantial economic and environmental ramifications, by critically analyzing the study's findings.[1]

At the 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE), held in Yogyakarta, Indonesia, Oktaviana, Febriani, Yoshana, and Payanta present their work titled "FoodX, a System to Reduce Food Waste". The "FoodX" system is emphasized in this study report as a creative approach to the issue of food waste. The main goal of the system is probably to create a platform that helps reduce food waste by giving users a useful way to handle extra food in different situations. The design, functionality, and possible applications of the system will be explained to readers, providing light on how technology might be used to address the problem of food waste.[2]

Patil, Nikam, Nair, Raut, and Lobo's study, "Sustainable Food Waste Management and Tracking System Using Blockchain," delves into the field of food waste management. This paper examines the integration of blockchain technology for food waste management and was presented at the 2023 International Conference on Advancement in Computation & Computer

Technologies (InCACCT) in Gharuan, India. The decentralized and transparent characteristics of blockchain provide distinctive opportunities for the monitoring and control of food waste along the whole supply chain. This research presumably looks at the viability and possible benefits of applying blockchain technology to sustainable food waste management. It is anticipated that readers will acquire knowledge about how this new technology might support efforts to reduce food waste in a more effective and long-lasting way.[3]

The paper "IoT Based Smart Waste Management System" is presented by Gayathri, Divagaran, Akhilesh, Aswiin, and Charan at the 7th International Conference on Advanced Computing and Communication Systems (ICACCS) in Coimbatore, India in 2021. This research paper probably talks about creating an Internet of Things-based smart trash management system. The system's capabilities, architecture, and potential to increase the effectiveness of trash collecting and disposal may all be covered by the writers[4]

The research project "Developing Food Charity Operations Management System" is presented by Alblihed, Almutairi, Almahmoud, Aladhadh, and Alabdulatif at the 2nd International Conference on Computing and Information Technology (ICCIT) in Tabuk, Saudi Arabia in 2022. The method intended to optimize and streamline food charity operations is probably the main emphasis of the article. It could investigate how technology might help with food waste reduction and resource allocation by helping to manage the logistics and delivery of food to those in need.[5]

Authors S.P. Kale, Meet Patel, Mehtab Ansari, Aditi Dhumal, and Ruchika Arote, who are connected to the Department of Computer Science & Engineering at Sandip Polytechnic in Nashik, India, have developed a novel web-based application called "FOOD WASTE MANAGEMENT SYSTEM" to address the problem of food waste in dining establishments, parties, and mess areas. The goal of the authors' research is to enhance the present food waste management system, which mainly provides information on food waste but is devoid of tools for data analysis and donation. The suggested approach offers a platform for giving extra food and uses data analysis to visualize the effects of food waste. By informing volunteers, orphanages, and NGOs in the area, this approach makes it possible for them to gather the extra food. Given that figures show a sizable amount of food is wasted annually, the report emphasizes how urgent it is to reduce food waste.[6]

The development of an intelligent food logistics system using an Android-based mobile application is the topic of the paper "FOOD WASTE MANAGEMENT USING ANDROID" by Aashish Khandkar, Palomi Gawali, Ajay Aswar, Yashaswi, and Yash Satpute, who are affiliated with the Sinhgad Academy of Engineering in Pune, Maharashtra, India. The authors stress the significance of effective technologies and procedures in the context of intelligent transportation systems since these systems directly affect the enhancement of food firms' competitiveness as well as the improvement of food quality, safety, and waste reduction. The study tackles the urgent problem of food waste and the necessity of making use of locally accessible food sources, such as leftover food from eateries, shops, and distribution centers. Crises like the COVID-19 outbreak have made this problem even more urgent. In order to alleviate hunger and food waste at the same time, the study primarily focuses on developing a mobile application that provides a ubiquitous platform for users to visualize and access nearby food resources.[7]

The possibilities of the Internet of Things (IoT) in relation to food waste management are examined by Tbk, Bharath, and Prashar in their work "Review on Efficient Food Waste Management System Using Internet of Things." In 2021, the International Journal of Current Research and Review released a research that was conducted. This research article probably explores how to use IoT to optimize food waste management. IoT has drawn a lot of interest as a viable solution for tackling a variety of environmental and socioeconomic concerns. An overview of Internet of Things (IoT)-based solutions and their uses in managing and monitoring food waste may be given by the writers. The benefits, difficulties, and possible solutions related to integrating IoT technology into the process of reducing food waste could be discussed.[8]

The "SeVa" app, created for food donation and encouraging smart living, is introduced in the article "SeVa: A Food Donation App for Smart Living" by Christina Varghese, Drashti Pathak, and Aparna Varde, which was published in 2021. The software seems to be centered around making it easier for people to donate extra food to those in need, which promotes a more humanitarian and environmentally friendly way of living. The writers most likely go over the features, capabilities, and general effects of the SeVa app on smart living and food giving.[9]

A smart electronic nose (eNose) system for managing food waste is presented in the paper "Smart eNose Food Waste Management System" by Shazmina Gull, Imran Sarwar Bajwa, Waheed Anwar, and Rubina Rashid, which was published in the Journal of Sensors in 2021. The authors acknowledge the increasing problem of food waste and provide a creative remedy that makes use of eNose technology. The eNose system is intended to identify and evaluate food scents, enabling in-the-moment food freshness and quality monitoring. The authors want to minimize food waste by enabling timely actions, such recognizing food products approaching their expiration date and supporting their efficient distribution or donation, by integrating data acquired from the eNose into a smart food waste management system.[10]

Furthermore In the restaurant business, intuition and historical data analysis serve as the main foundations for sales forecasting. This method might not be able to make precise forecasts, which could result in issues like excessive food production or inadequate preparation—both of which increase food waste.

All things considered, the existing system is deficient in the real-time data insights, sophisticated analytical tools, and efficient communication channels needed for food waste management and donation. To reduce food waste, a more effective technology-driven system is required, one that can maximize food supplies, link donors and NGOs, and produce precise sales projections.

In the age of the Internet of Things, a substantial number of research on food have surfaced for different states. Studies on waste management, quality control, and production monitoring have been conducted with RFID, temperature, humidity, cameras, and many other sensors and modules. There isn't a single study that suggests utilizing inexpensive sensors to identify food items and detect them while also measuring the amount of food lost. The research in issue is unique in this specific context because of its structure and methodology.

# Chapter 3
# Project Statement

The need for creative solutions that make use of cutting-edge technologies is driven by the growing worldwide concern for waste management, environmental sustainability, and philanthropic giving. an app for donation  using IOT technology. With serious consequences for the environment, society, and economy, food waste is a major worldwide problem. Nonetheless, millions of people experience hunger and food insecurity globally. In order to close this gap and solve this issue, food donation and trash management require creative IoT (Internet of Things) solutions. The way trash is managed and food is donated now is frequently ineffective, opaque, and underutilizes contemporary technologies.

The purpose of this project is to create a cutting-edge Internet of Things system that can monitor, control, and improve food donation and waste management procedures.

## 3.1. Project Scope

The scope of the IoT Assisted Food Donation and Waste Management Web Application project encompasses several key aspects:

1. **IoT Implementation and Sensor Deployment:** The project intends to deploy Internet of Things (IoT) devices with sensors at many food supply chain nodes, such as facilities for production, storage, and distribution. Critical characteristics including temperature, humidity, freshness, and expiration dates will be continually monitored by these sensors.

2. **Real-time Monitoring and Data Collection:** Real-time food quality indicator monitoring and data collecting will be made easier by IoT devices. To guarantee an accurate and fast evaluation of the state and safety of food items, this data will be gathered and processed.

3. **Web Application Development:** The goal of the project is to provide an intuitive user interface for online applications. For stakeholders, this application will act as a centralized platform for managing, accessing, and interpreting the data gathered by Internet of Things sensors. Features including inventory management, predictive analytics, data visualization, and automatic warnings will be available.

4. **Food Quality Assurance:** Establishing procedures for ongoing observation and evaluation is part of the project's focus on food quality assurance. By doing this, the system will be able to correctly identify possible threats to food safety and send out notifications in a timely manner.

5. **Efficient Food Redistribution:** The online tool will make it easier to identify excess food and efficiently distribute it to areas experiencing food insecurity or philanthropic groups. This part entails developing efficient procedures to link excess food with underprivileged individuals, guaranteeing prompt and efficient distribution.

6. **Sustainability and Waste Reduction:** Food waste is to be reduced to the lowest possible level using effective redistribution techniques and IoT-driven data analysis. The project is in line with sustainability objectives as it lessens the environmental effect of food waste.

7. **Stakeholder Engagement and Training:** An essential component of the project scope is involving the parties engaged in the food supply chain and provide the required training for using the web application and IoT gadgets. Personnel employed by charity organizations, distribution centers, kitchens, and storage facilities are included in this.

The scope of the project includes sustainable waste reduction, effective food redistribution, the development of a comprehensive web application, real-time monitoring through the integration of IoT technology, and stakeholder engagement to guarantee system implementation and utilization.

### 3.2. Project Assumptions

Without a doubt, the following presumptions have been taken into account for the Internet of Things Assisted Food Donation and Waste Management Web Application project:

1. **IoT Sensor Reliability:** It is assumed that the Internet of Things (IoT) sensors positioned across the food supply chain would function dependably, offering precise and constant information about temperature, humidity, freshness, and expiry dates.

2. **Data Accuracy and Integrity:** Throughout the monitoring process, it is expected that the data gathered by the Internet of Things sensors and processed by the system would remain accurate and reliable. This also includes presumptions regarding minor failures or defects in the sensors.

3. **Stakeholder Cooperation:** For the purpose of collecting data and redistributing food efficiently, the initiative assumes the active collaboration and engagement of stakeholders engaged in the food supply chain, including employees in production facilities, storage areas, distribution centers, and charity organizations.

4. **Web Application Performance:** It is anticipated that the created web application would run flawlessly, providing intuitive user interfaces and features for inventory management, analytics, data visualization, and alarms, all without any serious technical issues.

5. **Regulatory Compliance:** The project presupposes compliance with applicable data protection, privacy, and food safety laws and regulations with regard to the processing and administration of sensitive food quality and inventory data.

6. **Availability of Resources:** It is contemplated that the IoT system and web application would be implemented and maintained successfully with the availability of all required resources, including financial, technological, and human resources.

7. **Acceptance of Behavioral Change:** The initiative makes the assumption that stakeholders are somewhat willing to modify their habits when it comes to donating and managing food waste. This entails making assumptions about people's readiness to change current habits and accept new technology.

8. **Environmental Conditions:** The initiative is predicated on the assumption that environmental factors, such as stable infrastructure (such as dependable internet access) and a sufficient power source, would allow IoT devices to operate without experiencing major interruptions.

The project's planning and execution are heavily influenced by these presumptions. For the project to be executed successfully and to achieve its intended goals, these assumptions must be continuously monitored and validated throughout the project lifetime.

## 3.3. Project Limitations

For the IoT Assisted Food Donation and Waste Management Web Application project, there are undoubtedly the following possible restrictions to take into account:

1. **Sensor Accuracy and Reliability:** IoT sensors' dependability and accuracy can have drawbacks since they could occasionally experience failures or inaccurate data collecting. The accuracy of food quality evaluations may be impacted by differences in sensor performance or calibration problems.

2. **Data Connectivity and Infrastructure:** The smooth functioning of IoT devices and the web application may encounter difficulties in places with limited infrastructure or poor internet connectivity, such as distant locales or impoverished regions. This might possibly influence accessibility and real-time data transfer.

3. **Cost Implications:** The development of the web application and the deployment of IoT devices may come with substantial installation and maintenance costs. Limited funding may cause the project's scope or size to be restricted, which would affect its efficacy and reach.

4. **Stakeholder Acceptance and Engagement:** The initiative may not succeed if players in the food supply chain are resistant or do not actively participate. Inadequate training, reluctance to embrace new technology, or a lack of dedication to the project's goals might impede its successful execution.

5. **Data Privacy and Security Concerns:** Limitations may arise when processing sensitive data on food quality and distribution, particularly when it comes to data privacy and security. Maintaining data security and integrity while complying with strict data protection laws may be difficult.

6. **Environmental Factors and Sensor Limitations:** IoT sensor performance may be impacted by environmental factors, such as high humidity or temperatures, which might reduce the accuracy of the data. The reliability of food quality assessments may be affected by sensor limitations in severe or changeable environmental conditions.

7. **Scale and Reach:** Geographical restrictions or operational difficulties may limit the project's scope and size. The project's overall efficacy and impact may be limited if it is unable to reach all food supply chain segments or cover a large geographic region.

8. **Behavioral Change Challenges:** It may be difficult to encourage behavioral changes in communities or companies with regard to food waste management and donation practices. The project's planned goals may not be as successful due to cultural norms, misunderstandings, or opposition to change.

Identifying and acknowledging these potential limitations is crucial for effective project planning and management, allowing for proactive measures to mitigate risks and challenges throughout the project lifecycle.

## 3.4. Project Objectives

With an emphasis on utilizing IoT technology to revolutionize food management methods, the IoT Assisted Food Donation and Waste Management Web Application project seeks to accomplish a wide range of goals. In order to provide real-time monitoring of critical factors including temperature, humidity, freshness, and expiry dates, this initiative's central component is the integration of IoT devices with sensors across the food supply chain.

The project centered on the "IoT Assisted Food Donation and Waste Management Web Application" has the following main goals:

**1) Implement Real-time Food Odor Monitoring:**

In order to monitor and evaluate food quality in real time, the project intends to connect IoT technology with odor-sensing capabilities. All throughout the food supply chain, the system continuously monitors food conditions thanks to the deployment of odor-detecting sensors. This goal makes it easier to identify any deterioration or spoiling early on, allowing for prompt intervention to preserve food safety and freshness.

**2) Facilitate Connection Between Donors and Beneficiaries:**

Another goal is to use an easy-to-use web application interface to establish a smooth relationship between food providers and recipients. This website acts as a link between donors and recipients, giving leftover food products an effective way to provide and making gifts easily accessible to areas experiencing food poverty or charity organizations. The goal is to optimize the procedure such that excess edible food is efficiently and quickly distributed to those in need.

**3) Expand Reach to Alleviate Starvation:**

A larger population facing poverty or food shortage is the target audience for this initiative. The goal of the system is to optimize food redistribution operations by utilizing IoT technologies and an integrated network. The aim is to enhance the system's ability to support and aid a greater number of hungry individuals or communities by means of effective monitoring, inventory management, and enhanced communication between donors and recipients. The project's objective is to considerably reduce food insecurity among disadvantaged groups, and this expansion fits with that goal.

In order to solve social issues associated to food shortage and hunger, these goals work together to advance food quality evaluation, reduce waste, and guarantee that excess food reaches those in need.

# Chapter 4
# Project Requirements

## 4.1. Resources

**A. Human Resources:**

1. Trained IoT Technicians: Required for the installation, maintenance, and calibration of IoT devices across various locations in the food supply chain.
2. Skilled Software Developers: Proficient in web application development to create an intuitive interface and ensure seamless functionality.
3. Data Analysts/Experts: Capable of interpreting and analyzing IoT-generated data for accurate food quality assessment.

**B. Reusable Software Components:**

1. Data Preprocessing Modules: Essential for standardizing and cleaning data obtained from IoT sensors. These modules ensure uniformity and consistency in data format and quality, improving accuracy in subsequent analysis.

## 4.2. Software & Hardware Requirements:

### 4.2.1. System Components:

**A. Software Requirement:**

1. Operating System (Windows 11)
2. REACT JS
3. Database (MongoDb)
4. Node JS
5. Express

**B. Hardware Requirement:**

1. NodeMCu- ESP8266
2. Temperature Sensor: K Thermocouple Using MAX 6675
3. Gas Sensor: MQ4
4. Jumper Wires
5. ESP8266-01

## 4.3. Requirements Rationale

Table 4.1. Requirements Rationale

| **Requirements** | **Rationale** |
|---|---|
| Trained IoT Technicians | Accurate data collection and dependable sensor operation are ensured by skilled technicians installing, calibrating, and maintaining Internet of Things devices correctly. These factors serve as the foundation for a trustworthy evaluation of food quality. |
| Skilled Software Developers | Skilled developers are necessary to create a user-friendly and useful web interface. Their knowledge guarantees that users can access data, engage with the system, and make good use of the application's capabilities. |
| Data Analysts/Experts | Data analysis specialists are essential for deciphering and evaluating the enormous volumes of information gathered by Internet of Things devices. Their observations help with precise evaluation of food quality and well-informed decision-making. |
| Data Preprocessing Modules | The quality and dependability of ensuing data analysis for precise insights into food quality are improved by these components, which standardize and clean a variety of data from IoT sensors. They also guarantee uniformity and consistency in data format. |
| Web Development Frameworks | Using frameworks such as Flask or Django is crucial when creating a UI for a website. With the use of these tools, stakeholders will be able to easily manage data, view information, and communicate with the system.. |
| Data Visualization Tools | For IoT-generated data to be presented in an understandable fashion, tools such as Blynk are required. They help stakeholders make decisions based on clear insights by facilitating the visual depiction of complicated data.. |
| IoT Sensors and Infrastructure | The core of the system is made up of strong servers or cloud infrastructure and high-quality IoT sensors. Sturdy sensors provide precise data gathering, and sufficient hardware infrastructure for effective data processing, storage, and availability. |

## 4.4. Risk Management

## 4.4.1 Project Risk factors

Table 4.2. Project Risk factors

| **Risk Factor** | **Risk Level** | **Description** |
|---|---|---|
| Sensor Malfunction or Inaccuracy | High | Possibility of IoT sensor malfunctions, resulting in erroneous data from technological flaws or calibration problems. This may make it more difficult to evaluate and make decisions about food quality. |
| Data Security Breaches | High | The online application is susceptible to cyber attacks that might result in unauthorized access or data breaches. Sensitive food quality or distribution data might be jeopardized, which would undermine stakeholder compliance and confidence. |
| Limited Stakeholder Engagement | Medium | Efficient food redistribution and data accuracy may be impacted by inadequate involvement or opposition from supply chain players, which might impede the system's installation or use. |
| Technological Integration Challenges | Medium | Compatibility problems arising from the integration of IoT devices with current software systems or infrastructure may cause delays or obstructions to the proper operation of the web application and data flow. |
| Environmental Factors Impacting Sensor Performance | Low | The accuracy of sensors may be somewhat impacted by environmental factors (such as severe temperatures or fluctuations in humidity), which might occasionally lead to inconsistent readings. However, with the right sensor location, this danger level might be reduced. |
| Regulatory Compliance Issues | Low | There may be minimal dangers associated with difficulties in complying with changing food safety, technological, or data privacy legislation or standards. Proactive compliance steps, however, can greatly reduce these risks. |

## 4.5. Functional Specifications

1. **Interfaces**

A) **Functions or Methods Exposed by Components/Web Service:**

a) **IoT Device Interfaces:** These interfaces include features that let Internet of Things devices send sensor data back to the mainframe. For example, protocols or APIs are set up to process and receive temperature, humidity, and expiration date data. The smooth integration and data transfer from sensors to the system's backend for additional processing is ensured by these interfaces.

b) **Data Processing Interfaces:** Processes and procedures in charge of handling incoming data from Internet of Things devices include cleaning, normalization, validation, and storage preparation. Data consistency, integrity, and analysis readiness are guaranteed via these interfaces.

c) **Web Application Interfaces:** Between the front end and back end of the web application, these interfaces act as a link. In addition to handling requests for data updates or retrieval, they also comprise functions that control user interactions and make it easier for the user interface to show analyzed data.

B) **Actions Performed by Components:**

a) **IoT Devices:** Ensuring continuous data flow for analysis and decision-making by sending real-time data produced by sensors (temperature variations, humidity levels, expiration alarms) to the central system at predetermined intervals.

b) **Data Processing Modules:** To guarantee correctness and consistency, incoming data must undergo stringent validation tests, data cleaning, and preparation procedures. This include verifying the accuracy of the data, addressing discrepancies, and preparing the data for analysis and storage.

c) **Web Application:** By presenting analytical data in an approachable and dynamic graphical format, users may keep an eye on food quality indicators, control inventory, link donors and recipients, and react to alarms or notifications issued by the system.

**2. External Interfaces Required:**

A) **IoT Sensor Interfaces:** These interfaces cover communication protocols (such HTTP and MQTT), which enable smooth data transfer from sensors to the data intake module of the central system. From external IoT devices to the internal infrastructure, they provide dependable data transport.

B) **Third-party APIs:** interfaces for integrating other systems or services; examples are logistics APIs for effective transportation planning or weather APIs for adding climatic data that may affect food quality. The functionality and data sources of the application are enhanced by integration with these APIs.

**3. Internal Interfaces Required:**

A) **Data Processing to Database:** Interfaces that control how processed data is moved and stored in databases, guaranteeing that the data is available and persistent for further analysis or retrieval by the web application.

B) **Web Application to Database:** Interfaces that allow the web application to communicate with the database in order to get and show users the appropriate information. These interfaces assist features like inventory management and real-time data visualization by facilitating easy data retrieval and update activities.

C) **Inter-component Communication:** Interfaces inside the system that provide communication between various sections or parts. These interfaces provide effective data sharing and system interoperability by ensuring smooth data flow and processing between components.

**4. Communication Interfaces:**

A) **IoT Protocols:** The communication protocols that IoT devices and the central system employ to transmit data are governed by these interfaces. Secure and effective data sharing is made possible by protocols like HTTP (Hypertext Transfer Protocol) and MQTT (Message Queuing Telemetry Transport), which provide dependable communication and little data loss.

B) **RESTful APIs:** Interfaces for communication used to exchange data between the front-end and back-end servers of a web application. RESTful APIs make it possible to get, edit, and manipulate data, which enables the web interface to efficiently process user requests and provide real-time data.

**5. Graphical User Interfaces (GUI):**

**A) Components of GUI:**

a) **Dashboard**: A complete display that provides inventory status, important warnings, and real-time food quality measurements in an understandable way. Users are given a summary of the overall health and functionality of the system.

b) **Data Visualization Tools:** Users may better grasp trends, patterns, and anomalies in food quality indicators by using graphs, charts, and diagrams that illustrate the examined data. Decision-making and data understanding are improved by these visual aids.

c) **Donor-Beneficiary Connection Interface:** ISmart displays that make it possible for donors to give extra food and make it easier for recipients to access gifts that are available. The contribution process is streamlined by this interface, enabling effective and convenient communication between donors and recipients..

d) **Alert and Notification Panels:** interfaces designed to show messages or alerts generated by the system on changes to the inventory, deviations in food quality, or important events. These panels alert users quickly, allowing for speedy resolution of possible problems.

**B) Actions Supported by GUI:**

a) **Monitoring Food Quality:** Through the use of real-time data visualization from IoT devices, the GUI enables users to analyze trends, monitor metrics related to food quality, and spot any deviations or possible hazards.

b) **Inventory Management:** Users may examine the food products that are available, their amounts, and the status of their distribution using the GUI's inventory management and tracking functions. The technology makes it simple for users to trace the movement of food supplies and organize contributions.

c) **Donor-Beneficiary Interaction:** Donors can enter details on excess food products that are available for donation through the GUI, and recipients can see and claim such donations in accordance with their needs. An easy-to-use interface guarantees that communications between contributors and receivers go smoothly.

### 4.5.1. Interactions

**1. Donors' Interaction:**

**Interface:**

User-Friendly Donation Interface: Donors access a clear and intuitive web interface.

**Interaction Flow:**

**Login/Register:**

Donors log in with credentials or register if new to the system.

**Donation Input:**

They input details of surplus food items: item description, quantity, expiry date, and pickup location.

**Submission:**

Upon submission, donation details are sent for processing and validation.

**Confirmation:**

Donors receive confirmation notifications once the donation is successfully submitted.

**Backend Interaction:**

Donor-input data is processed by the system to ensure accuracy, validity, and availability for display to beneficiaries.

**2. Beneficiaries' Interaction:**

**Interface:**

**Browse and Claim Interface:** Beneficiaries interact with a user interface displaying available donations.

**Interaction Flow:**

**Login/Register:**
Beneficiaries log in or register to access available donations.

**Browse Donations:**
They browse donations categorized by type, quantity, and location.

**Claim Request:**
Beneficiaries select desired items and request their claim.

**Confirmation:**
Upon claim approval, beneficiaries receive confirmation with pickup details.

**Backend Interaction:**
The system checks inventory availability, confirms the claim, and updates inventory status upon successful beneficiary selection.

**3. System Administrators' Interaction:**

**Interface:**
Admin Dashboard: Administrators access a backend dashboard for system oversight and management.

**Interaction Flow:**

**Login:**
Administrators log in to the administrative panel.

**Monitoring Dashboard:**
They monitor real-time food quality metrics, inventory status, and system alerts.

**Data Management:**

Admins manage user accounts, oversee donations, and ensure system functionality.

**Alert Response:**
System administrators respond to critical alerts by initiating appropriate actions or notifying stakeholders.

**Backend Interaction:**
The admin panel enables direct access to system metrics and allows admins to manage data, users, and system operations.

- **Interactions Among Interfaces:**

1. **IoT Device Interface to Central System:**

**Data Collection:**
IoT devices collect food quality data and transmit it to the central system via MQTT or other predefined protocols.

**Central Processing:**
The system's data processing modules validate, clean, and preprocess incoming data for further analysis and storage.

2. **Web Application Interface to Database:**

**Data Retrieval:**
The web application's interface interacts with the database to retrieve information about available donations, inventory status, and user accounts.

**Data Update:**
User interactions with the interface trigger updates in the database, reflecting changes in donations, inventory, and user actions.

3. **Communication Interfaces:**

**IoT Protocols:**
IoT devices utilize established protocols to transmit sensor data securely and reliably to the central system.

**RESTful APIs:**
Web application interfaces interact with the system's back-end server through RESTful APIs, facilitating seamless data retrieval, updates, and display.

**Interaction Flow:**

**Donation Process Flow:**

1. Donors submit surplus food details via the web interface.
2. IoT devices transmit sensor data to the central system.
3. Data processing modules validate and process incoming data.
4. Processed data is stored in the database for display on the web interface.

**Beneficiary Claim Process Flow:**

1. Beneficiaries browse available donations on the web interface.
2. They request the claim of desired items.
3. Upon approval, the system updates inventory records and notifies the beneficiary for pickup.
4. System Monitoring and Management Flow:
5. System administrators monitor real-time metrics via the administrative panel.
6. They respond to critical alerts, manage users, and ensure system functionality.

## 4.5.2. Sustainability

1. **Real-time Monitoring:** To keep an eye on the status of food goods, IoT devices may be put in refrigerators and storage facilities. This lessens the possibility of waste by ensuring that perishable goods are consumed before they expire.

2. **Automated Inventory Management:** IoT sensors have the ability to monitor food inventory levels and notify users when products are about to expire. In order to reduce waste, this permits prompt consumption or donation.

3. **Smart Logistics:** Food may travel the fastest and most efficient path possible thanks to IoT devices in transit vehicles that improve delivery routes.

4. **Predictive Analytics:** Organizations may improve their planning for food distribution and lessen overstocking or shortages by using data gathered from IoT devices to forecast demand trends.

5. **Matching Platforms:** To make sure that extra food is given to people in need, IoT may be connected with internet networks that link food givers and recipient organizations in real-time.

6. **Quality Assurance:** Food donations may have their quality and safety monitored by sensors, guaranteeing that only appropriate food is given out.

7. **Energy Monitoring:** IoT devices can track how much energy is used in transportation and storage facilities, which can assist businesses in identifying places where they can cut back on energy use and their carbon impact.

8. **Resource Optimization:** The system can help optimize the use of water, power, and other resources related to food storage and transportation by offering insights about resource usage.

9. **Analytics for Improvement:** The gathered information may be examined to find patterns, chances for development, and ways to make the whole food supply chain more effective.

10. **Consumer Awareness:** IoT device deployment in public areas, such restaurants or grocery shops, may inform customers about the harm that food waste does to the environment and promote responsible consumption.

11. **Traceability:** By offering a visible picture of the food supply chain, IoT can aid in guaranteeing adherence to laws governing food safety and traceability.

12. **Community Engagement:** Coordination of Volunteers: The Internet of Things can help coordinate volunteers for food distribution and donation campaigns, making it possible to respond to community needs in a more planned and effective manner.

13. **Waste Reduction Strategies:** By effectively managing excess food and matching it with recipients before it goes bad, the app aims to reduce food waste.

14. **Environmental Impact:** The technique tries to lessen the environmental impact of disposing of food waste by redistributing excess food instead of throwing it away.

**Features:**

a) **Optimized Logistics:** Advanced algorithms optimize food pickup/delivery routes to minimize transportation-related carbon emissions.
b) **Localized Donations:** Encouraging local donations reduces the ecological footprint linked with long-distance transportation of surplus food.

**Long-term Sustainability:**

a) **Community Engagement:** Building partnerships with local communities and businesses to promote ongoing sustainable food donation practices.
b) **Data-Driven Insights:** Leveraging data analytics to identify trends in surplus food generation, aiding in the development of sustainable food management strategies.

### 4.5.2.1 Quality management

1. **Food Safety Standards:** Respecting food safety regulations is essential. Verify that the system conforms with national and international laws pertaining to food safety.Utilize the concepts of Hazard Analysis and Critical

Control Points, or HACCP, to recognize, assess, and manage risks to food safety at every stage of the process.

2. **Data Accuracy and Reliability:** The data that IoT devices gather must be accurate and dependable. To guarantee reliable readings, calibrate and maintain sensors on a regular basis.Put in place procedures for data validation and verification to find and fix any flaws or inconsistencies in the data.

3. **Real-time Monitoring and Alerts:** Provide the ability to monitor donated food and trash management procedures in real-time.To enable quick corrective action in the event of any departure from predetermined quality requirements, set up alerts and notifications.

4. **Traceability and Transparency:** Turn on traceability features so you can follow the food donation from its origin to its destination.Transparency may be promoted by giving stakeholders access to pertinent data, which will increase confidence and accountability.

5. **Real-time Monitoring:** IoT sensors make ensuring that quality requirements are met by constantly monitoring important food quality characteristics including temperature, humidity, and expiration dates.

6. **Data Processing:** In order to preserve accuracy and integrity, robust data processing modules check, clean, and process incoming data.

**Features:**

a) **Threshold Alerts:** When food quality metrics depart from predetermined levels, instant warnings and alarms are set off, allowing for prompt remedial action to preserve quality.
b) **Quality Assurance Checks:** At several points along the donation and distribution process, there should be routine audits and inspections to make that quality standards are being followed.

#### 4.4.2.2 Security

For the protection of data both during transmission and storage, use robust encryption techniques. This covers all communications between devices and the central system, as well as data gathered from sensors.

a) **Authentication and Authorization:** Strong authentication procedures should be put in place to guarantee that only authorized users or devices may access the system.Establish distinct permission levels to limit who is able to carry out particular system functions.

b) **Device Security:** Update firmware and software on a regular basis to fix vulnerabilities and secure IoT devices.To stop unwanted access to devices, use hardware-based security features and secure boot procedures.

c) **Network Security:** To safeguard the network infrastructure, use intrusion detection/prevention systems and firewalls.To keep IoT devices separate from other important systems, think about deploying a specialized network or VLAN for them.

**Features:**

a) **Access Controls:** Data confidentiality is ensured by role-based access control (RBAC) techniques, which limit system access based on user roles.

b) **Regular Security Updates:** Software patches and upgrades applied on time to fix bugs and guard against online dangers.

# Chapter 5
# System Architecture And Proposed Architecture

## 5.1. Design Consideration

**1. System Architecture:**

**Modularity:** Design the system with modular components, allowing easy integration, scalability, and maintenance.

**Scalability:** Ensure the architecture accommodates future growth in data volume, user base, and functionalities without compromising performance.

**Fault Tolerance:** Implement redundancy and failover mechanisms to ensure system resilience against hardware or software failures.

**2. User Interface (UI) and User Experience (UX):**

**Intuitive Design:** Create an intuitive and user-friendly interface for seamless interaction and navigation across the application.

**Accessibility:** Ensure the application is accessible to users with disabilities by adhering to accessibility guidelines.

**Responsive Design:** Design a responsive interface that adapts to various devices and screen sizes for a consistent user experience.

**3. Data Management and Processing:**

**Data Integrity:** Implement robust data validation and verification processes to maintain data accuracy and consistency.

**Data Privacy:** Adhere to data privacy regulations and implement encryption techniques to protect sensitive user and food quality data.

**Real-time Processing:** Enable real-time data processing to swiftly analyze incoming information from IoT devices for prompt decision-making.

**4. IoT Device Integration:**

**Interoperability:** Ensure compatibility with various IoT devices and sensor types through standardized protocols for seamless data exchange.

**Security Measures:** Implement secure communication protocols and authentication mechanisms to prevent unauthorized access to IoT devices and data tampering.

**Device Management:** Incorporate features for device monitoring, maintenance, and remote configuration to ensure continuous functionality.

**5. Security Measures:**

**Role-Based Access Control (RBAC):** Implement RBAC to control user access and permissions based on predefined roles within the system.

**Encryption Techniques:** Use encryption methods to protect data during transmission, storage, and processing.

**Regular Audits:** Conduct routine security audits and assessments to identify vulnerabilities and ensure compliance with security standards.

**6. Performance and Optimization:**

**Caching Strategies:** Employ caching mechanisms to enhance system performance and reduce response time for frequently accessed data.

**Load Balancing:** Implement load balancing techniques to distribute traffic evenly across servers and ensure optimal performance.

**Optimized Queries:** Optimize database queries and indexing to enhance query execution speed and system responsiveness.

**7. Compliance and Regulations:**

**Adherence to Standards:** Ensure compliance with relevant industry standards, data privacy regulations (such as GDPR, HIPAA), and food safety guidelines.

**Documentation:** Maintain comprehensive documentation detailing system design, functionalities, and compliance measures for reference and audits.

## 5.2 Assumption and Dependencies

### 5.2.1 Assumptions:

1. **IoT Device Availability and Compatibility:**

**Details:** Assumes the availability and compatibility of IoT devices equipped with sensors for monitoring food quality parameters (e.g., temperature, humidity).

**Rationale:** Relies on the presumption that suitable IoT devices are accessible for integration and data collection.

2. **Internet Connectivity:**

**Details:** Assumes stable and consistent internet connectivity for seamless communication between IoT devices, the web application, and the central system.

**Rationale:** Relies on uninterrupted internet access for real-time data transmission and system functionality.

3. **User Adoption and Engagement:**

**Details:** Assumes users (donors, beneficiaries, administrators) will actively engage with and adopt the platform for donation, claiming, and system monitoring.

**Rationale**: Success of the application relies on user willingness to participate and utilize the system for donations and monitoring.

4. **Data Accuracy**:

**Details:** Assumes accuracy and reliability of data collected from IoT sensors, assuming that sensor data accurately represents the actual food quality conditions.

**Rationale:** System functionalities and decision-making are reliant on the accuracy of sensor data.

5. **Regulatory Compliance:**

**Details:** Assumes compliance with applicable laws, data privacy regulations (e.g., GDPR, HIPAA), and food safety guidelines throughout system operations.

**Rationale:** Adherence to regulations is crucial for data protection, user privacy, and ensuring food safety standards.

6. **Resource Availability:**

**Details:** Assumes availability of necessary resources (hardware, software, skilled personnel) for system development, maintenance, and ongoing support.

**Rationale:** Availability of resources is essential for the successful implementation and sustainability of the system.

### 5.2.2. Dependencies:

1. **IoT Device Integration:**

**Details:** Dependent on successful integration and seamless communication between IoT devices and the central system using standardized protocols (e.g., MQTT, HTTP).

**Rationale:** Functionality of the system relies on the accurate and timely transmission of data from IoT devices.

2. **Third-Party Services/APIs:**

**Details:** Dependencies on third-party services or APIs for functionalities such as weather data retrieval, mapping services, or authentication mechanisms.

**Rationale:** Reliance on external services for additional functionalities essential to the application's operation.

3. **Data Processing Modules:**

**Details:** Dependencies on the reliability and accuracy of data processing algorithms and modules for real-time analysis and validation of incoming sensor data.

**Rationale:** Critical for ensuring the accuracy and quality of processed data used for decision-making.

4. **Security Measures:**

**Details:** Dependencies on robust security protocols, encryption techniques, and authentication mechanisms to safeguard data and system integrity.

**Rationale:** Crucial for protecting sensitive data and preventing unauthorized access or data breaches.

5. **User Engagement and Feedback:**

**Details:** Dependencies on active user engagement, participation, and feedback for system improvements and effective utilization of the application.

**Rationale:** User involvement is necessary for ongoing system enhancements and usability improvements.

6. **Regulatory Compliance Assurance:**

**Details:** Dependencies on ensuring continued adherence to legal and regulatory requirements throughout the system's lifecycle.

**Rationale:** Compliance with regulations is vital to maintain trust, legality, and ethical standards within the system.

## 5.3. General Constraints

**1. Budgetary Constraints:**

**Limited Funding:** Constraints on available budget may restrict the acquisition of necessary resources, tools, or skilled personnel for development.

**Resource Allocation:** Insufficient funding might limit the scope of the project, affecting features, scalability, or technological choices.

**2. Time Constraints:**

**Project Deadlines:** Tight project schedules might limit the time available for thorough testing, refinement, and implementation of features.

**Development Cycles:** Short development cycles can affect the depth of research, leading to potentially less optimal solutions or rushed implementations.

**3. Technical Constraints:**

**Technology Limitations:** Compatibility issues between different technologies or limitations of selected frameworks could impact integration and functionalities.

**IoT Hardware Constraints:** Limitations in IoT sensor capabilities or connectivity issues may affect the accuracy and reliability of data collection.

**4. Resource Constraints:**

**Human Resources:** Shortage of skilled professionals in IoT development, data analytics, or system administration can hinder project progress.

**Infrastructure Limitations:** Insufficient server capacity or cloud resources might affect scalability, performance, and data handling capabilities.

**5. Regulatory and Compliance Constraints:**

**Legal Compliance:** Evolving data privacy laws or food safety regulations may require ongoing adjustments, impacting development processes and data handling.

**Ethical Considerations:** Adhering to ethical guidelines regarding data collection, sharing, and user consent might pose constraints on certain functionalities.

**6. User Adoption Challenges:**

**Engagement and Education:** Difficulty in educating users (donors, beneficiaries, administrators) about the application's benefits and functionalities may hinder adoption rates.

**User Interface (UI/UX):** Poorly designed interfaces might deter user engagement and hinder the application's usability.

**7. Environmental Constraints:**

**IoT Device Environment:** Environmental factors, such as extreme temperatures or physical obstructions, may interfere with IoT device operations, impacting data accuracy.

**8. Security and Privacy Concerns:**

**Cybersecurity Risks:** Vulnerabilities in the system might compromise data integrity, leading to privacy breaches or unauthorized access, requiring robust security measures.

## 5.4. Block Diagram

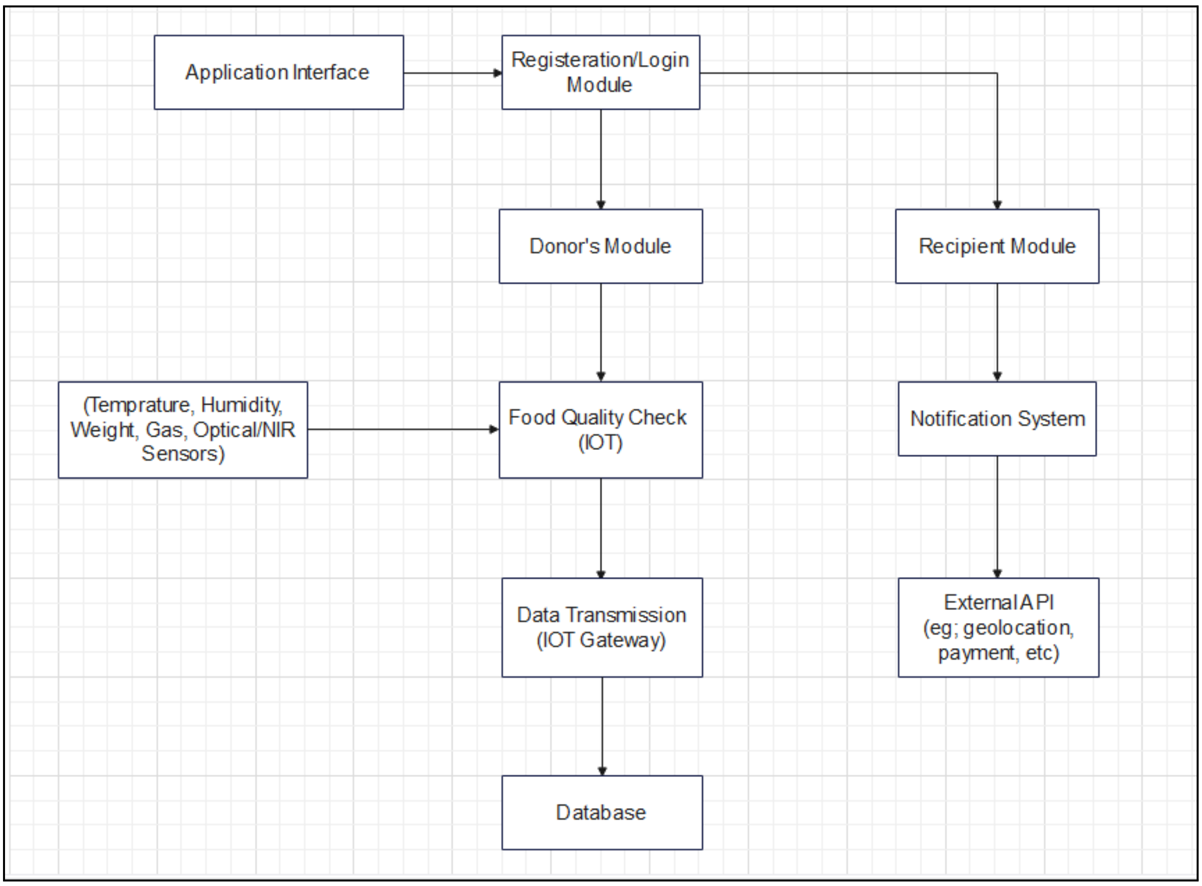


Fig.5.1 Block Diagram

## 5.5. Modules of the Project

1. **User Management Module:** Handles user registration, authentication, and authorization.Manages user profiles and permissions for donors, recipients, and administrators.

2. **Donation Module:** Allows donors to register and input details about available food donations.Provides a platform for scheduling and coordinating food pickups or deliveries.Includes features for tracking and confirming donations.

3. **Recipient Module**: Enables recipients to register and create profiles indicating their needs. Facilitates the matching of donors with recipients based on location, food preferences, and availability.

4. **IoT Device Integration Module:** Manages the integration of IoT devices, including sensors and actuators. Monitors real-time data from devices to track food conditions, temperature, and other relevant parameters.

5. **Inventory Management Module:** Tracks and manages the inventory of available food donations. Sends alerts for expiring or expired food items, facilitating timely distribution.

6. **Geolocation and Routing Module:** Utilizes geolocation data to optimize the routing of food pickups and deliveries. Ensures efficient logistics and minimizes transportation costs.

7. **Notification Module**: Sends real-time notifications and alerts to donors, recipients, and administrators. Notifies donors about successful pickups and recipients about incoming donations.

8. **Data Analytics and Reporting Module:** Gathers and analyzes data to generate

insights on donation patterns, system usage, and efficiency. Provides customizable reports for stakeholders and administrators.

9. **Security Module**: Implements security measures, including data encryption, access controls, and secure communication. Monitors and logs security-related events for auditing purposes.

10. **Waste Management Module:** Tracks and manages the disposal of waste generated from the system. Ensures compliance with waste management regulations and environmental standards.

11. **Feedback and Rating Module:** Collects feedback from donors and recipients about their experiences. Allows users to rate and review the donation process for continuous improvement.

12. **Training and Documentation Module:** Provides training materials and resources for users and administrators. Offers documentation on system features, best practices, and troubleshooting.

13. **Regulatory Compliance Module:** Monitors and ensures compliance with food safety, privacy, and data protection regulations. Provides tools for administrators to update policies and ensure adherence.

14. **Community Engagement Module:** Encourages community participation through features like forums, events, and outreach programs. Facilitates collaboration with local organizations and businesses.

15. **Dashboard and Visualization Module:** Creates a user-friendly dashboard for administrators to monitor key metrics and system performance. Utilizes visualizations to represent data trends and patterns.

**The system comprises of three major modules:**

1. Agent
2. Donor

3. Admin

**A. Admin Module:**

**View Requests:**

- Access a dashboard displaying requests from Donors, NGOs, and logistics.
- Filter and sort requests based on status, date, or type.

**Request Approval:**

- Review details provided in each request to verify authenticity.
- Accept or deny requests based on verification results.
- Send notifications to relevant parties regarding approval or denial.

**Logistics Coordination:**

- Map logistics and assign them to collect donations from specific Donors or locations.
- Monitor and track the progress of logistics in picking up donations.

**Complaints and Suggestions Handling:**

- View and address complaints or suggestions submitted by users (Donors, NGOs, Agents).
- Respond to user queries or issues regarding donation requests or system functionalities.

**B. Donor Module:**

**Registration and Authentication:**

- Create an account by providing necessary details for registration.
- Log in securely using registered credentials for access.

**Request Submission:**

- Fill request forms with details like donation date, quantity, and location.
- Send in requests to have food contributions gathered.

**Request Status Tracking:**

- View the status of submitted donation requests - whether pending, approved, or denied.
- Receive notifications/alerts for changes in request status.

**Feedback Submission:**

- Provide feedback, suggestions, or complaints related to the website's usability or functionalities.
- View responses or actions taken by Admins on submitted feedback.

**C. Agent Module:**

**Notification Handling:**

- Receive notifications/alerts from Admins regarding assigned donation pickups.
- Access a dashboard displaying pickup details and locations.

**Collection Update:**

- Mark the collection of donated food upon pickup from Donors' homes.
- Confirm and update the system with successful pickups.

**View Collected Donations:**

- Access a history of previously collected donations with details like pickup date, location, and quantity.
- Track completed pickups and their associated information.

**Profile Management:**

- Update personal details, contact information, or work-related data as required.
- Maintain accurate and up-to-date profiles for communication and assignment purposes.

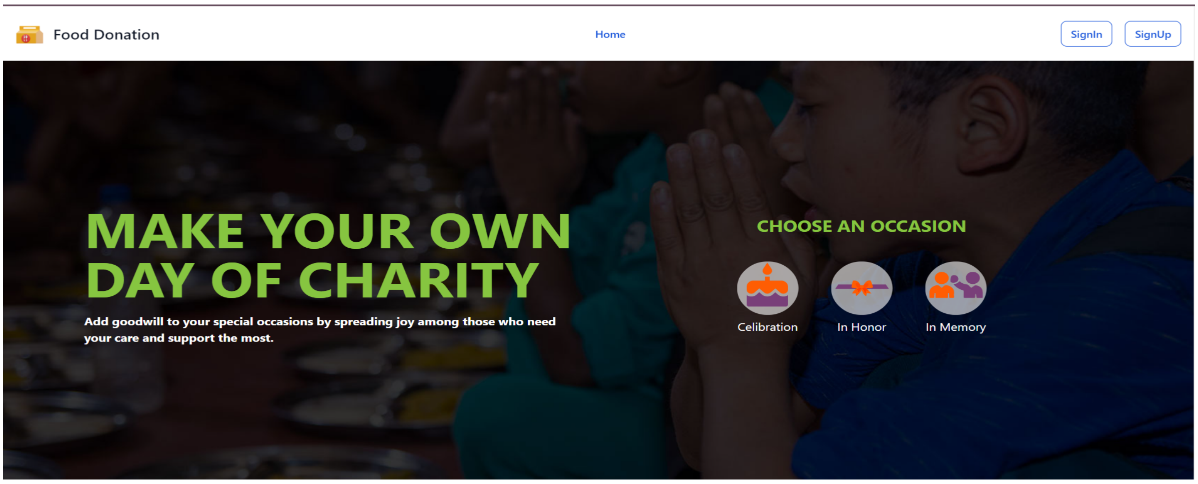


Fig. 5.2 Home Page

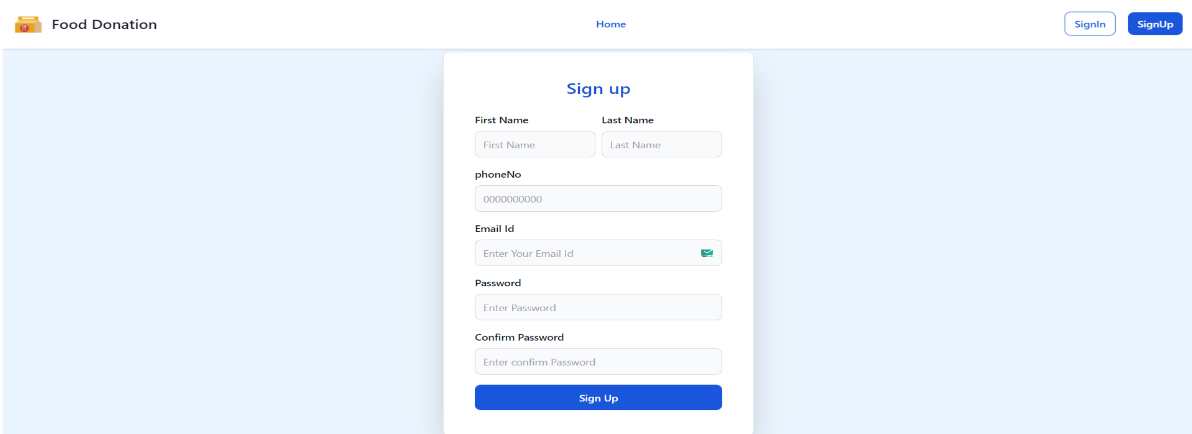


Fig.5.3 Sign up Page

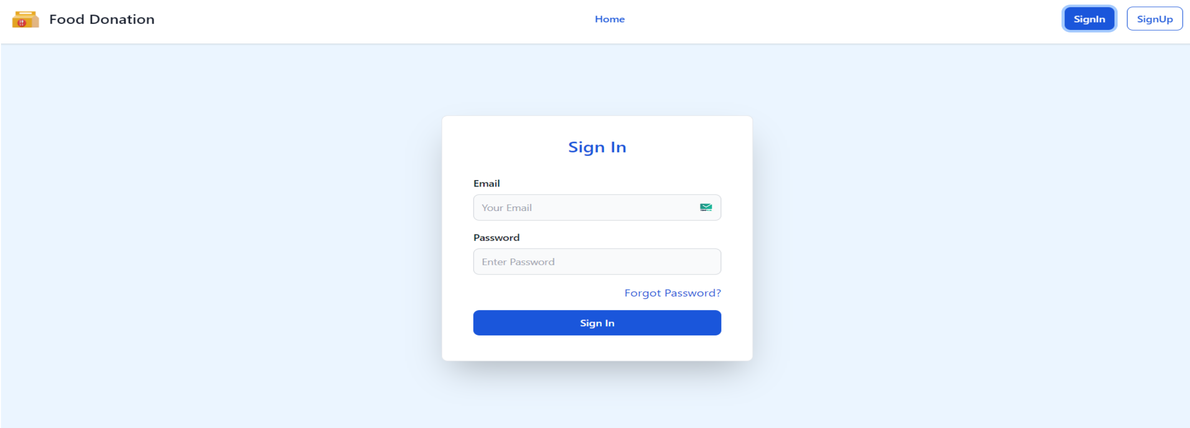


Fig.5.4 Sign in Page

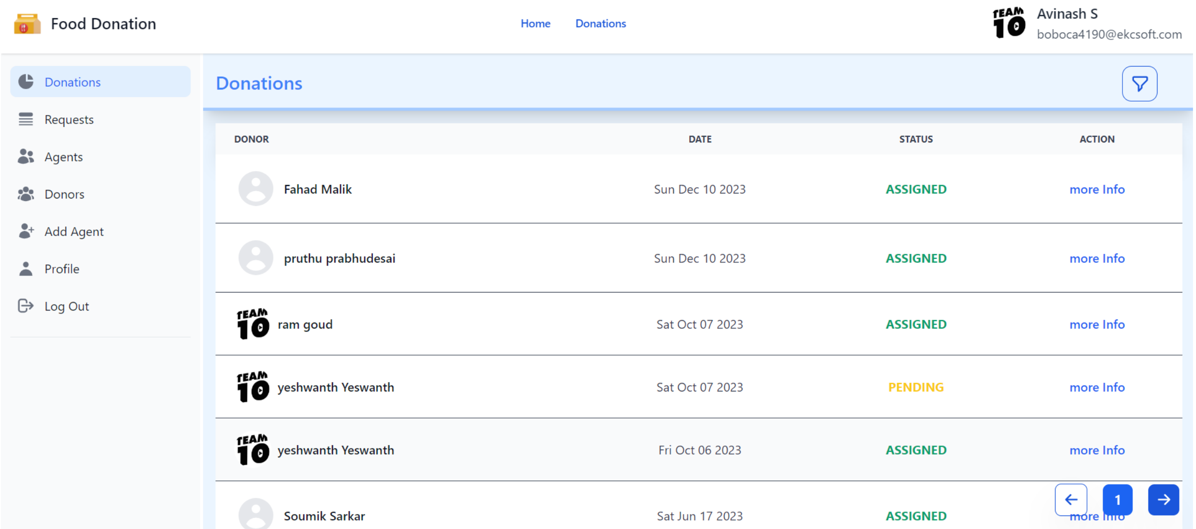


Fig.5.5 Admin Donations Module

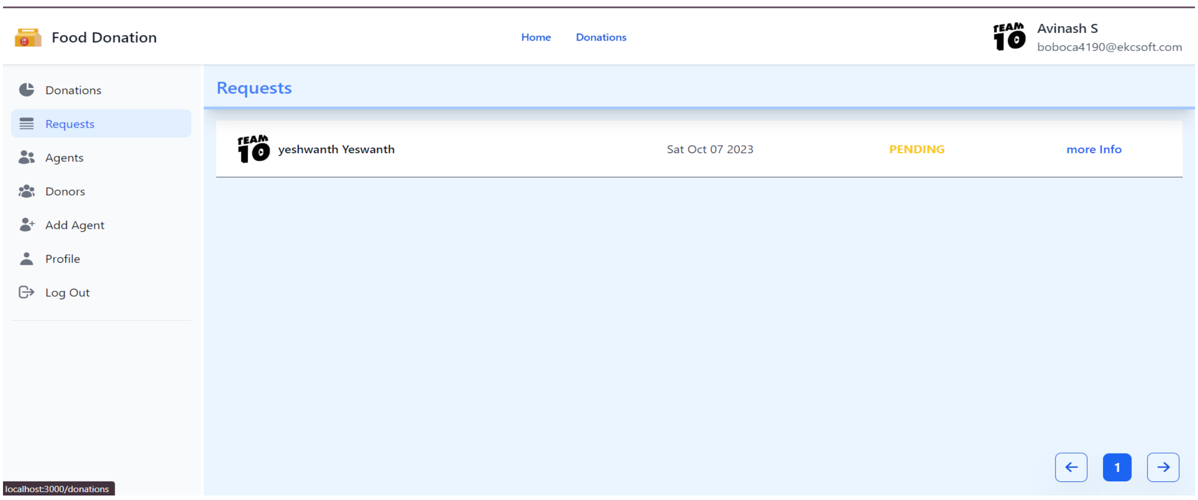


Fig.5.6 Admin Requests Module

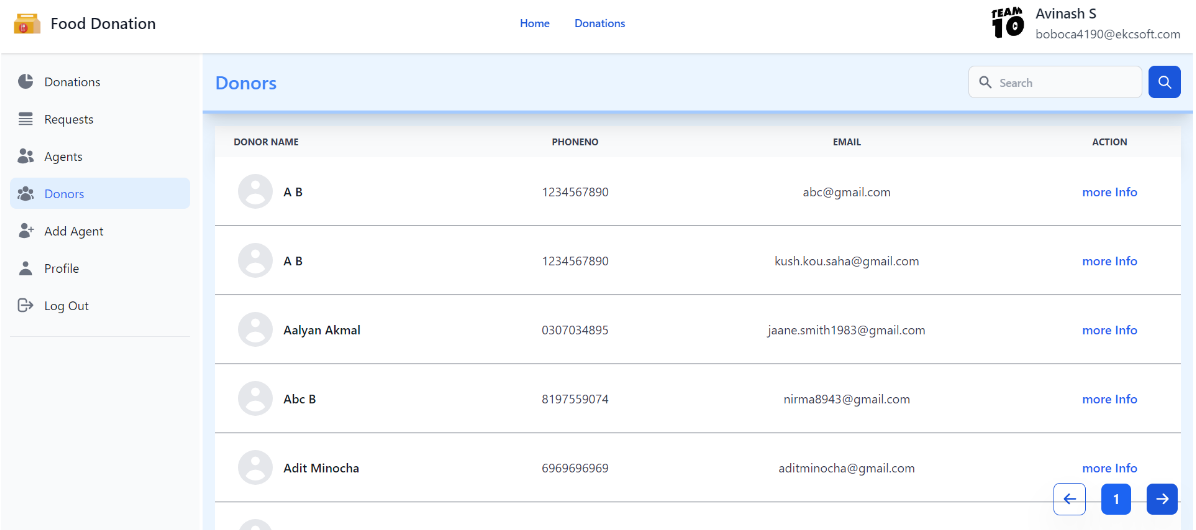


Fig.5.7 Admin Donors Module

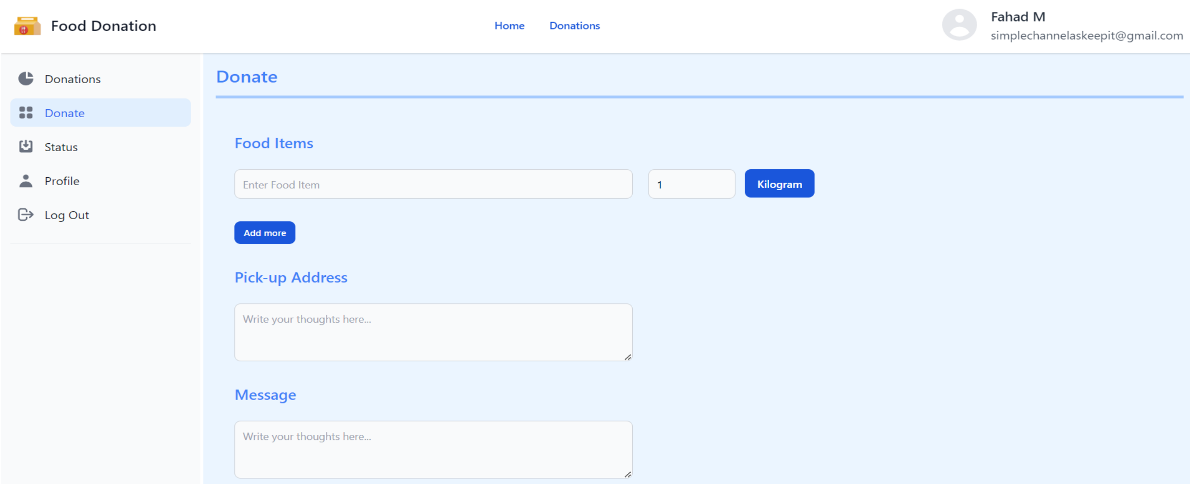


Fig.5.8 Users Donate Module

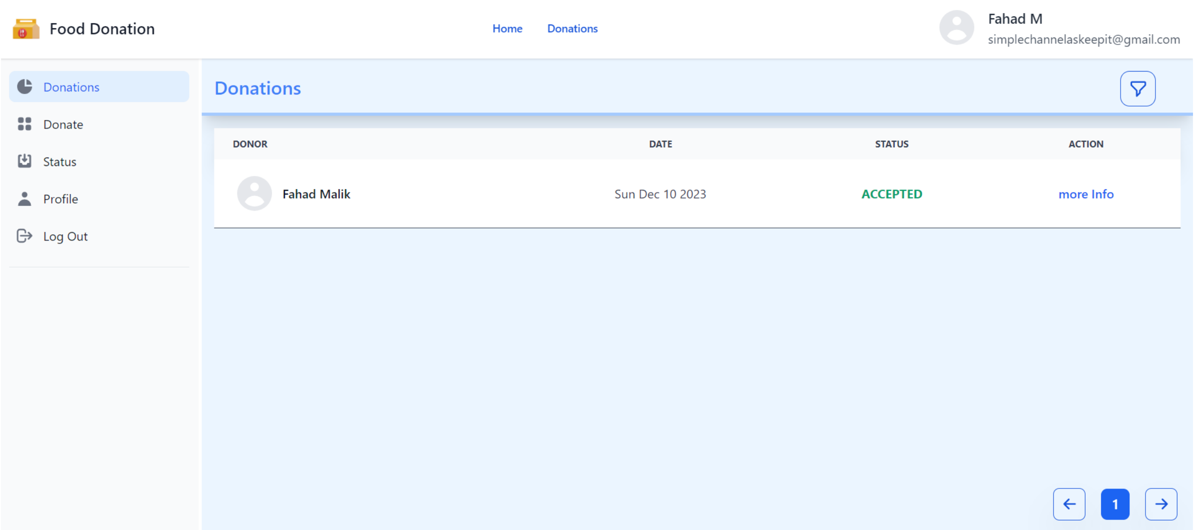


Fig.5.9 Users Donation Module

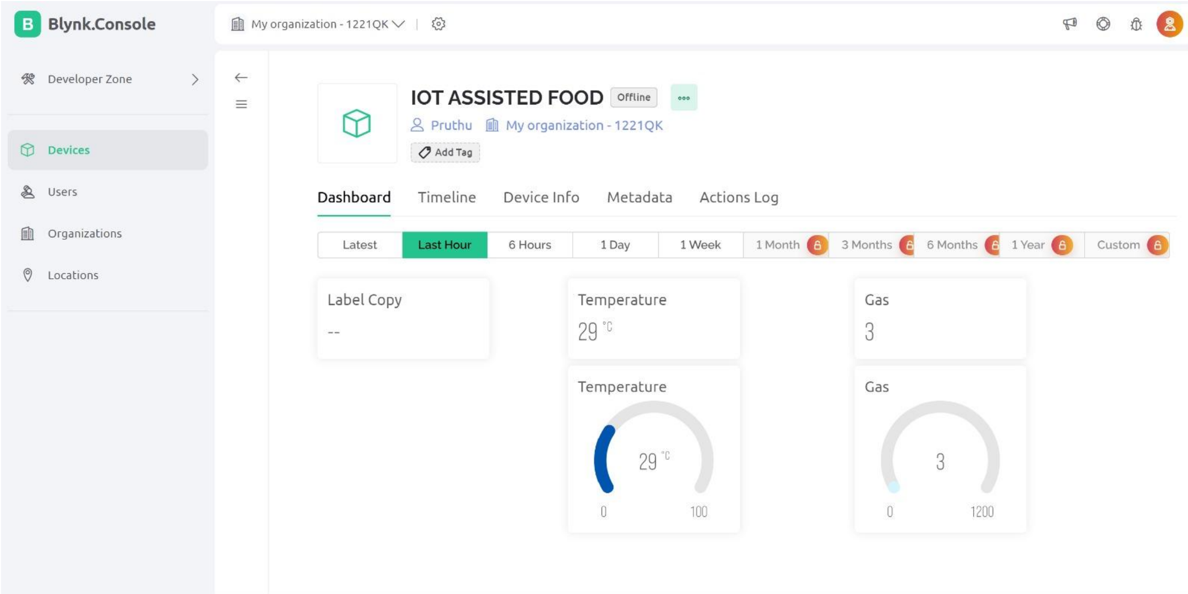


Fig.5.10 Blynk Dashboard Module

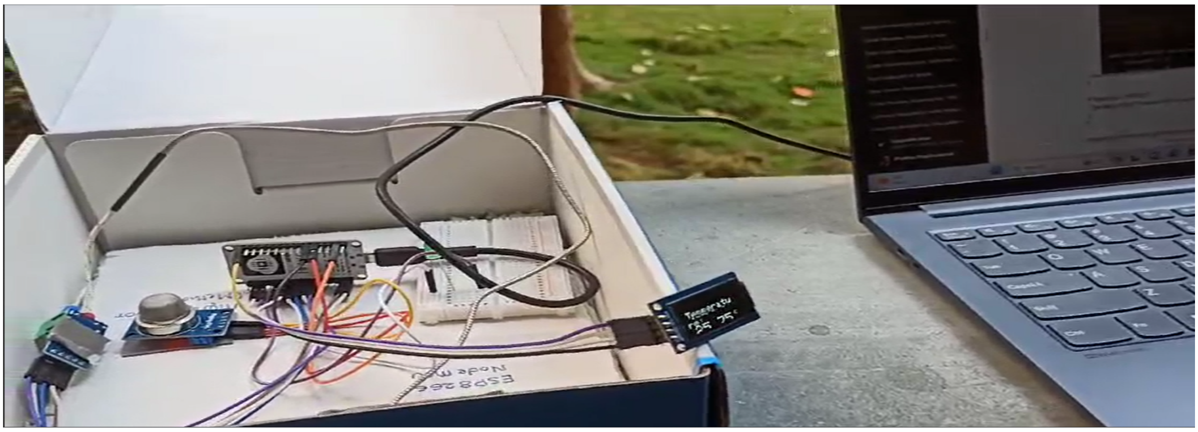

Fig. 5.11 IOT Project Model

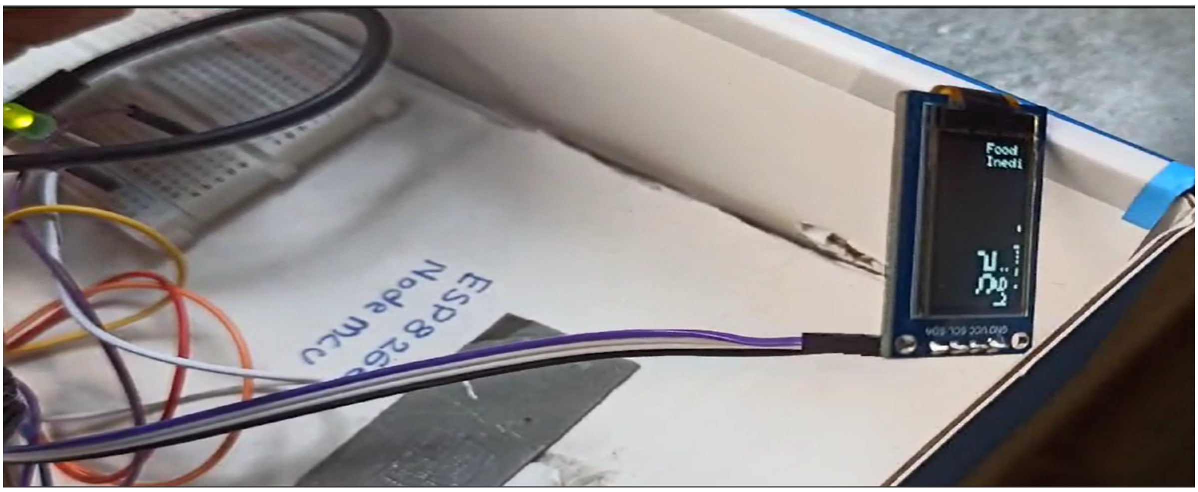


Fig. 5.12 IOT Project Reading

## 5.4. UML Diagrams

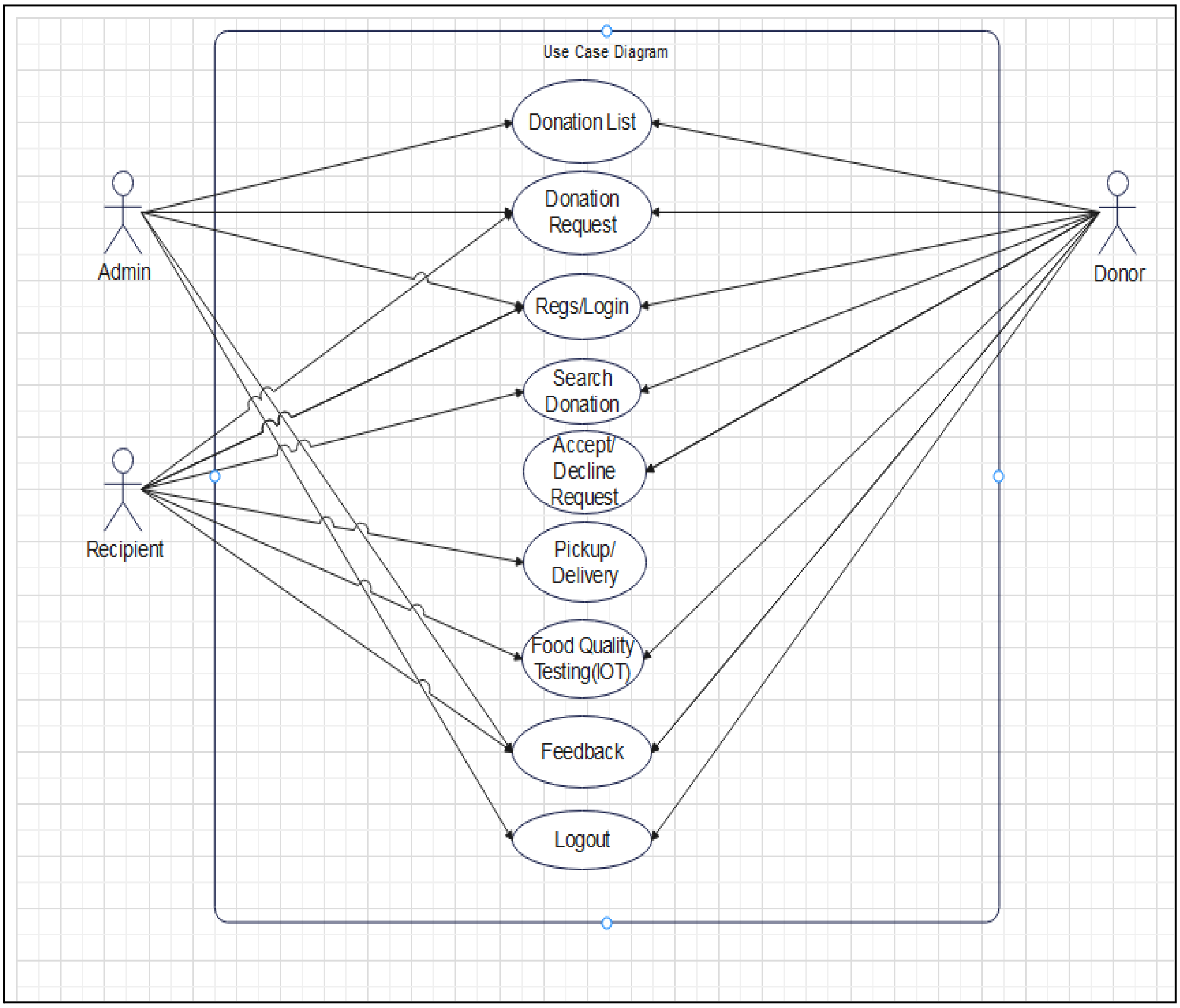


Fig. 5.13 USE CASE Diagram

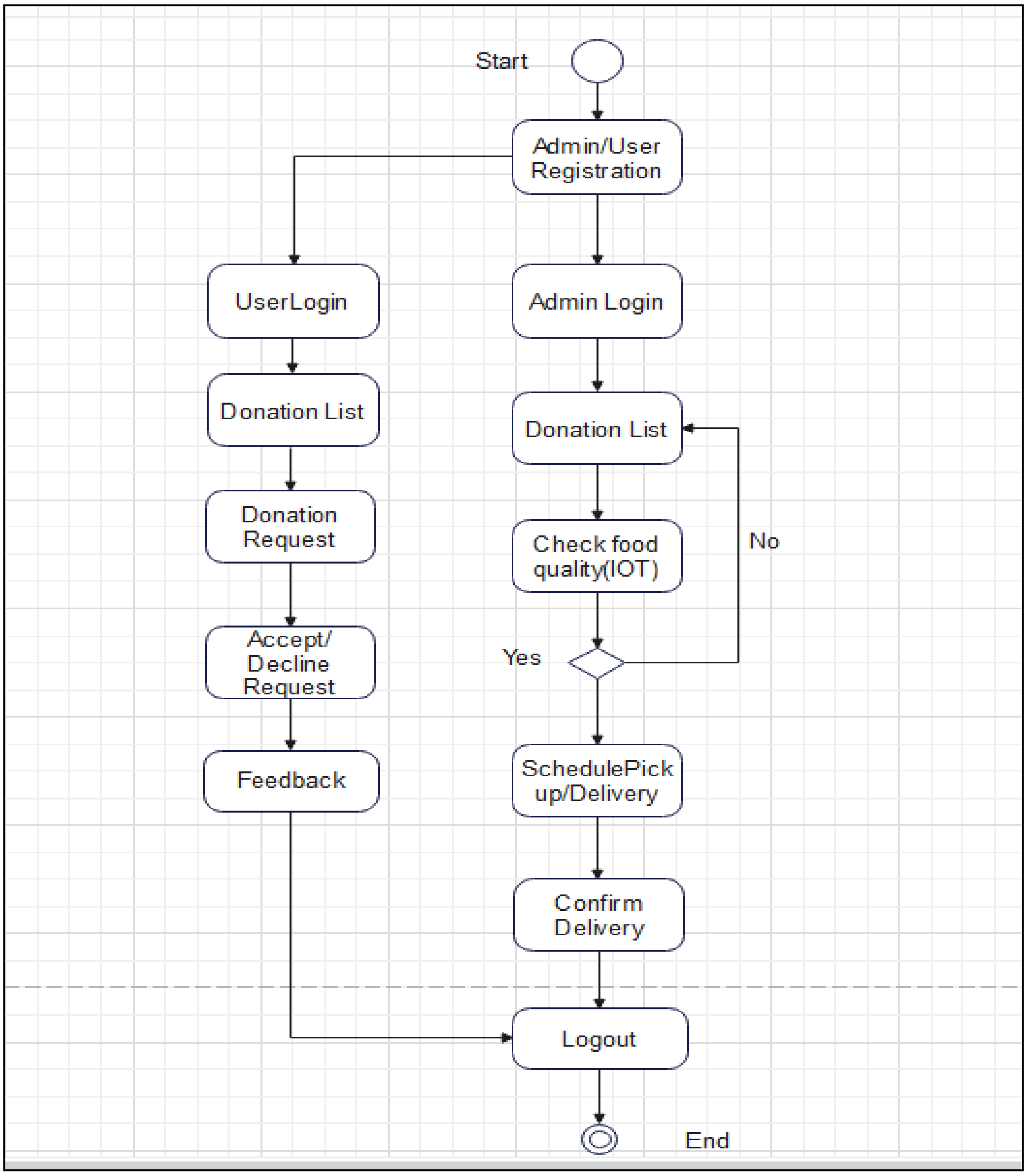


Fig.5.14 Activity Diagram

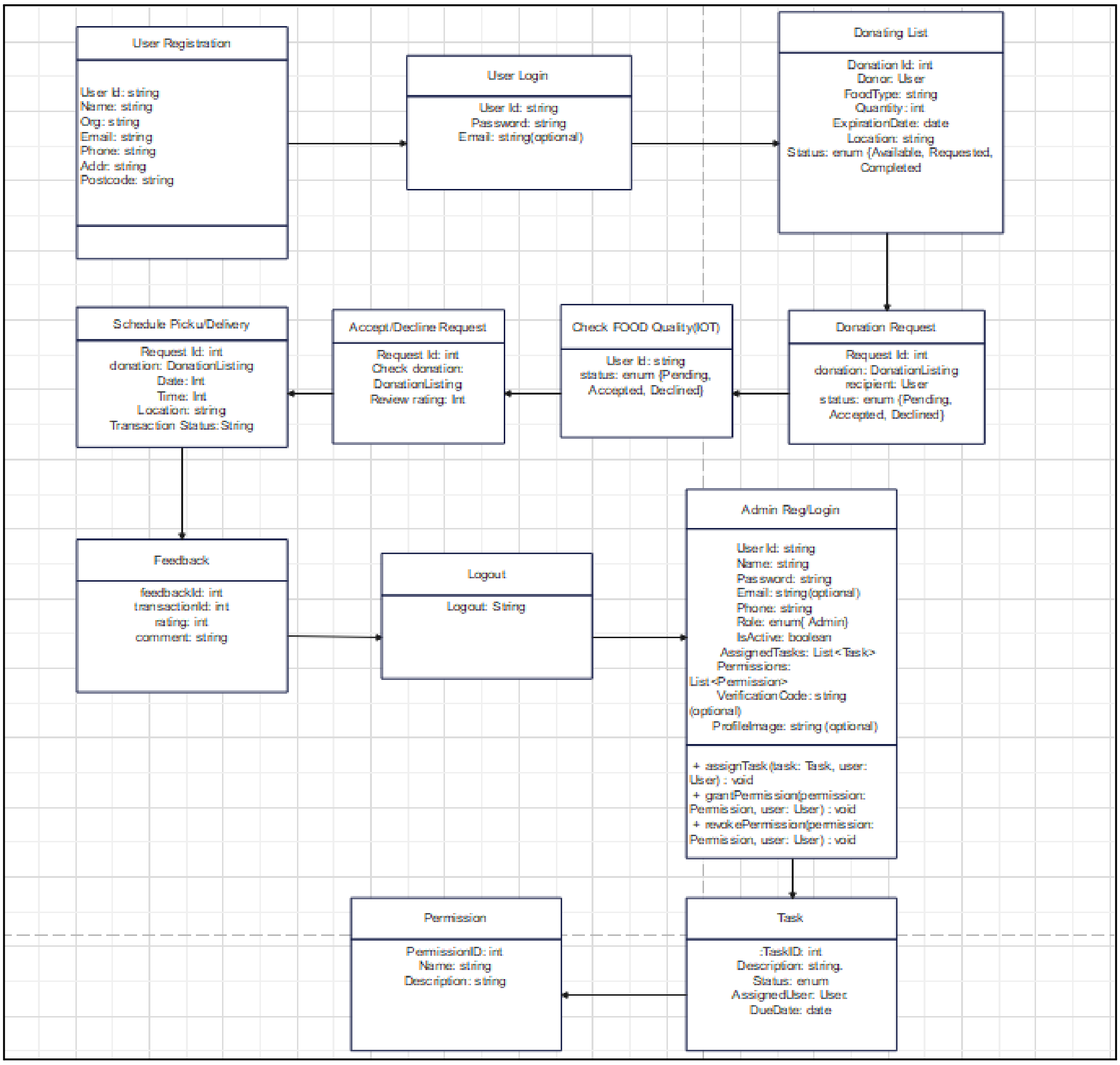


Fig.5.15 Class Diagram

## 5.6. System Architecture

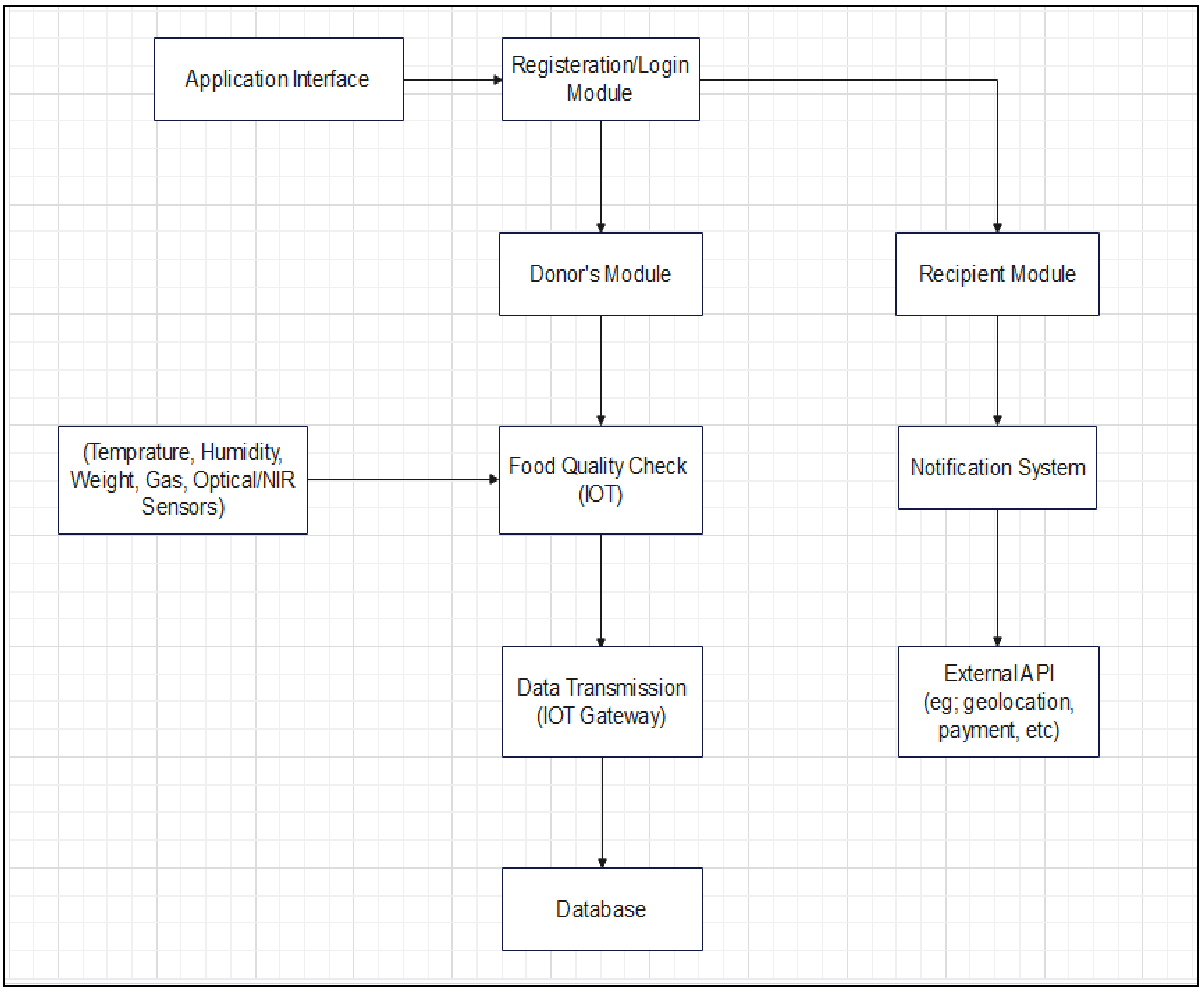


Fig. 5.16 System Architecture

**Components and Layers:**

1. **Presentation Layer:**

- Consists of browser-and mobile application-accessible responsive web interfaces for donors, agents, and administrators.
- Uses JavaScript, HTML, CSS, and frameworks like as React to create a dynamic and intuitive user interface.

2. **Application Layer:**

- Includes all of the modules, capabilities, and business logic needed for the system to perform properly.
- Based on user roles, implements user authentication, request processing, alerts, and data management.

3. **Data Layer:**

- Consists of the database system that uses NoSQL (MongoDB) or relational (MySQL, PostgreSQL) databases to store and manage different types of data items.
- Arranges user, donation, logistics, feedback, and IoT sensor data tables and collections.

4. **IoT Integration Layer:**

- Connects to Internet of Things (IoT) devices and sensors that track data on food quality, including temperature, humidity, and expiration dates.
- Uses Internet of Things (IoT) protocols, such as MQTT or HTTP, to gather sensor data and forward it to the application layer for processing.

**System Architecture Overview:**

1. **Web Server:**

- Oversees client-server communications and serves as the host for the front-end application.
- Uses technologies for web server functionality such as Nginx, Apache, or Node.js.

2. **Application Server:**

- Includes the backend logic written in Python, Java, or Node.js, among other computer languages.

- Use frameworks to process data, manage HTTP requests, and communicate with databases, such as Django, Spring Boot, or Express.js.

3. **Database Server:**

- Ensures consistency, dependability, and integrity of data by storing structured data in a relational or NoSQL database system.
- Makes use of suitable normalization, sharding, or indexing techniques to facilitate effective data storage and retrieval.

4. **IoT Gateway/Hub:**

- Acts as a middleware element that links the application and IoT devices.
- Interacts with sensors, gathers data from them, does some simple processing, and then sends it to the application server.

**Interaction Flow:**

1. **User Interaction:**

- Users use safe authentication methods to log in, then use the web interface according to their roles.
- Admins oversee requests, confirm details, allocate tasks for logistics, and respond to user comments.
- Donors make requests for donations, follow their progress, and offer comments.
- Agents may examine contribution histories, alter collection statuses, and get notifications when a donation is picked up.

2. **Data Flow:**

- Donors make gift requests, which are then authenticated by the administrator.

- Admins map requests to agents for pickup in order to organize logistics.
- Real-time data on pickups and logistics is provided by agents, who also update the system and gather donations.

**3. IoT Data Integration:**

- Data on food quality is continuously gathered by IoT sensors and sent to the IoT integration layer.
- The program receives incoming data, processes it, sets off warnings for differences in quality, and updates the appropriate modules.

# Chapter 6
# Project Plan

## 6.1. Project Plan

### Phase 1: Project Initiation

**Objective:** Establish the project team, specify goals, and provide the foundation for the endeavor.

**Activities:**

**Project Definition:** Describe the goals, limitations, assumptions, and scope of the project.

**Team Formation:** Organize a multidisciplinary group and assign roles and duties.

**Stakeholder Identification:** Determine the needs and expectations of the stakeholders by identifying and involving them.

**First Planning:** Create a broad project plan including deadlines, materials, and deliverables.

**Risk assessment:** Determine possible hazards and ways to reduce them so that early preparation may begin.

## Phase 2: Requirements Gathering and Analysis

**Objective:** Gather and analyze project requirements to define functionalities and system specifications.

**Activities:**

**User Requirements:** Engage stakeholders to collect user stories, needs, and expectations.

**Functional Specifications:** Define detailed functionalities required from the application (e.g., donation, inventory, user management).

**Non-functional Requirements:** Gather system performance, security, and compliance needs.

**Feasibility Study:** Evaluate technical, economic, and operational feasibility of the project.

## Phase 3: System Design and Prototyping

**Objective:** Plan the development, design the system architecture, and make prototypes.

**Activities:**

**Architectural Design:** Draw out a thorough system architecture that includes the databases, interfaces, and modules.

**Prototyping:** Creating wireframes or prototypes allows you to see and test the functionality of your system.

**Database Design:** Create the database schema with the demands for data retrieval and storage in mind.

**Technology Stack Selection:** Select the right tools and technologies for development.

## Phase 4: Deployment

**Objective:** Put the generated solution into practice, test it, and make it available in a production setting.

**Activities:**

**Development:** Follow the design guidelines while coding and building the application.

**Testing:** To guarantee functionality and quality, perform thorough testing (unit, integration, and user acceptability testing).

**Deployment Planning:** Arrange for a minimally disruptive, phased or complete deployment approach.

**Training and Documentation:** Offer thorough documentation and user training for the upkeep and operation of the system.

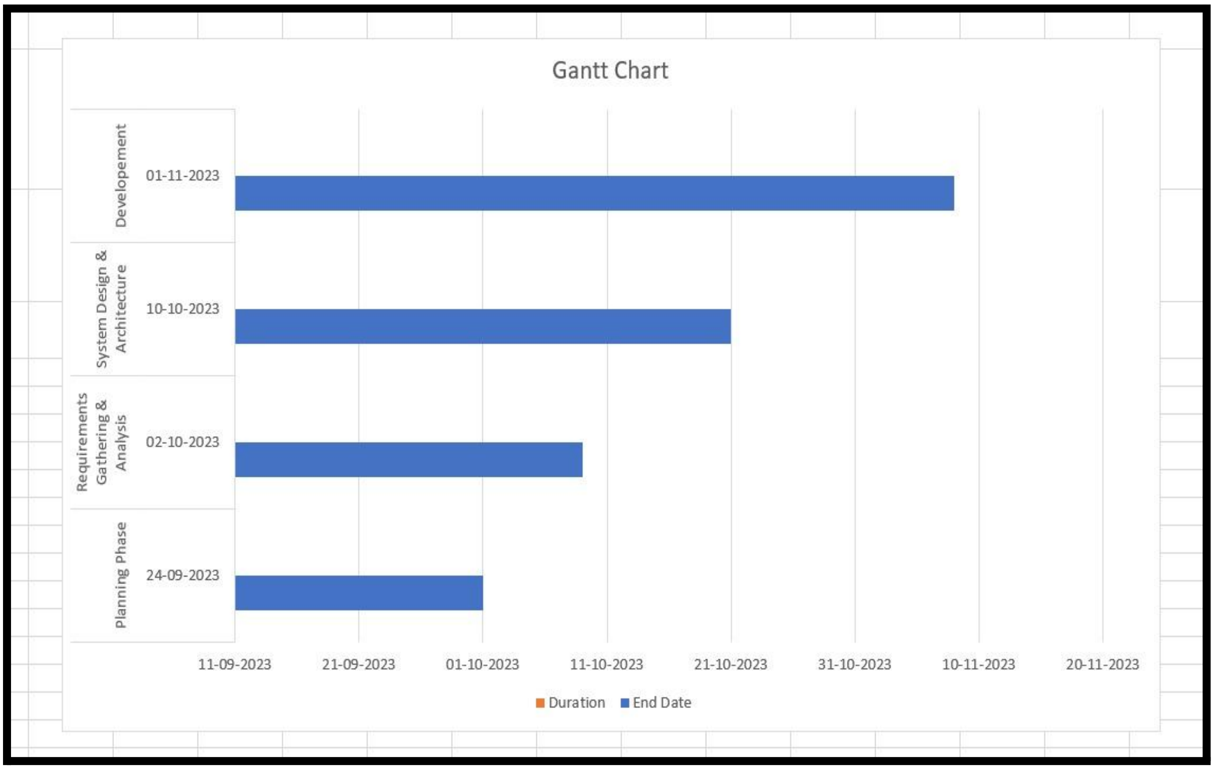


Fig. 6.1 Gantt Chart/ Project Timeline Chart

# Chapter 7
# Implementation

## 7.1. Methodology

**A. Communication:** Customer and developer contact is the first step in the software development process. We used the following communication phase concepts when we interacted with the user during this phase.We planned ahead of time, that is, we decided on the meeting's agenda with the goal of focusing on the features and services offered by other comparable programs.Our team's leader guides the group and gathers all user requirements, including what the user truly needs, what is input, and the format of the system's output.

**B. Planning:** Comprehensive scheduling, estimating, and risk analysis are all included.We scheduled the software delivery date, estimated project costs, and addressed project risk pertaining to the messenger program and file transfers throughout this phase.Lastly, at this phase, we calculated the project's total cost, accounting for all software expenses and software releases in accordance with user deadlines.

1. **Project Scope Definition:**

- Identifying Objectives:Clearly outlining the project's goals, which should include resolving food scarcity, establishing links between donors and recipients, and real-time food monitoring.
- Setting Scope Boundaries: Establishing the parameters of the project in order to determine what features and deliverables are included and not.

2. **Requirement Gathering and Analysis:**

- Stakeholder Engagement: Including all relevant parties, such as technical teams, donors, and beneficiaries, in order to collect expectations and requirements.
- Requirements Analysis: Defining functions, setting feature priorities, and producing comprehensive specifications by analyzing gathered requirements.

3. **Resource Allocation and Planning:**

- Resource Identification: Determining and allocating the necessary infrastructure, software tools, hardware, and human resources for the Internet of Things.
- Timetable and Planning: Establishing project schedules, milestones, and task plans will ensure efficient use of resources.

4. **Risk Assessment and Mitigation:**

- Risk Identification: Risk identification is the process of locating possible dangers and unknowns that could affect the completion of a project.
- Risk Mitigation Planning: Creating plans to reduce risks that have been recognized by taking preventative and backup actions.

5. **Development Methodology Selection:**

- Choosing technique: Depending on the needs of the project and the team's skills, choose a development technique (e.g., Agile, Waterfall, etc.).
- Iterative Planning: Dividing project work into smaller iterations to allow for flexibility and ongoing development.

6. **Procurement Planning:**

- Vendor Selection: Finding and choosing suppliers of IoT devices or specialist software components, if necessary.
- Contractual Agreements: Clearly defining terms and conditions in contracts with vendors for purchases.

7. **Communication and Collaboration Planning:**

- Planning communication routes and collaboration instruments is essential for efficient team coordination and engagement.
- Stakeholder Communication: Creating a communication strategy to update and inform stakeholders on the status of the project.

8. **Quality Assurance Planning:**

- Testing Strategies: Arranging test cases, settings, and procedures in advance to provide comprehensive quality control.
- Quality Standards: Establishing quality standards and benchmarks will help to guarantee that the system satisfies predefined requirements.

**9. Change Management Planning:**

- Change Control Procedures: Processes for managing and documenting modifications to project scope or requirements are known as change control processes.
- Impact assessment: Evaluating how modifications would affect deadlines, available resources, and final products.

**10. Contingency Planning:**

- Contingency Plans: Creating backup plans in case of unforeseen circumstances to keep the project moving forward in the event of an interruption.

### C. Modelling:

It comprises project design and a detailed study of the requirements. An algorithm is a step-by-step way to solve a problem, while a flowchart depicts the entire visual flow of a program.Block diagrams of the system are created after we analyze the user's requirements. That is nothing more than the UML-based behavioral structure of the system, such as the class, use case, and components diagrams.

### D. Construction

**1) Coding:** The proper programming language is used to implement design details.To communicate with the database on the server side, we have opted to use the Javascript programming language.

- MERN Stack Development
- Modular Design
- RESTful API Design:
- Database Design and Indexing
- Error Handling and Logging

Included HTML5 Geolocation API to determine the user's location automatically and determine the distance to the closest

**2. Implemented food Demand Forecasting:**

- Integrated food demand forecasting algorithms using historical data and external APIs.
- Implemented functions for users to enter food demand information or import data from external sources.

**3. Implemented Global Food Waste Analysis:**

- Collected global food waste data from FAO (Food and Agriculture Organization).
- Created visualizations such as charts and graphs that provide users with insight into global food waste patterns.
- Conducted Exploratory Data Analysis (EDA) to identify trends and correlations in food waste data.

**4. Sensor Deployment:**

- Sensors: IoT devices are equipped with various sensors (temperature, humidity, expiry date) strategically placed within the food supply chain (storage facilities, transport, donation centers).
- Purpose: These sensors make sure that food quality is constantly monitored and evaluated by gathering data in real time on critical food parameters.

**5. Data Transmission and Connectivity:**

- Communication Protocols: IoT devices utilize communication protocols (e.g., MQTT, HTTP) to transmit collected data securely to the central system.
- Connectivity: Devices are connected via wireless networks (Wi-Fi, cellular, or LPWAN) to ensure seamless data transmission.

**6. Real-time Monitoring and Data Collection:**

- Continuous Data Collection: IoT devices collect and transmit data in real-time, providing immediate updates on food conditions.
- Automation: Automated data collection allows for proactive response to changes in food quality, minimizing risks of spoilage or waste.

**7. Data Processing and Analysis:**

- Data Preprocessing: Raw data collected from IoT sensors undergoes preprocessing to clean, standardize, and prepare it for analysis.
- Analytical Insights: Data analytics and machine learning algorithms analyze the collected data, providing insights into food quality trends, anomalies, and predictions.

**8. Alert Systems and Decision Support:**

- Alert Mechanisms: Based on analyzed data, the system triggers alerts or notifications for stakeholders in case of deviations from predefined thresholds or potential risks to food quality.
- Decision Support: With the use of actionable insights gleaned from IoT-generated data, the solution helps stakeholders make wise decisions..

**9. User Interface Integration:**

- Web Application Interface: A user-friendly online interface is used to display and convey the data provided by the Internet of Things, allowing stakeholders to manage inventories, keep an eye on food quality parameters, and take appropriate action.

**10. Security Measures:**

- Encryption and Authentication: Implementing security protocols to secure data transmission and access control mechanisms to safeguard IoT devices and the central system from cyber threats.

**E. Testing:** Testing is done by dissecting the program; that is, we create the application's module first and then work step-by-step to identify input and output faults, including initialization, performance, interface, and data structure issues..

Here are key aspects of testing:

**1. Unit Testing:**

- **Purpose:** Testing individual components (e.g., IoT devices, data processing modules) in isolation to verify their functionalities.
- **Method:** Use of specialized testing frameworks or tools to validate the functionality of each unit, ensuring they perform as expected.

**2. Integration Testing:**

- **Purpose:** Testing the integration and interaction between different system components.
- **Method:** Verifying how different modules communicate and exchange data. Ensuring seamless connectivity and data flow among IoT devices, data processing modules, and the web application interface.

**3. Data Quality and Accuracy Testing:**

- **Purpose:** Ensuring the accuracy, completeness, and consistency of the data collected by IoT sensors and processed by the system.
- **Method:** Validating data accuracy against predefined standards and verifying that the processed data aligns with expected outcomes.

**4. Performance Testing:**

- **Purpose:** Evaluating the system's performance under various conditions, including load, stress, and scalability.
- **Method:** Stress testing the system with simulated high loads to assess its responsiveness, scalability, and ability to handle peak data loads without compromising performance.

**5. Security Testing:**

- **Purpose:** Identifying and addressing security vulnerabilities to safeguard against potential cyber threats or data breaches.
- **Method:** Conducting penetration testing, vulnerability assessments, and authentication checks to ensure the system is secure and complies with security standards.

**6. User Acceptance Testing (UAT):**

- **Purpose:** Validating the system's usability and functionality from an end-user perspective.

- **Method:** Involving stakeholders to interact with the system, providing feedback, and ensuring the system meets user requirements and expectations.

**7. Regression Testing:**

- **Purpose:** Verifying that modifications or updates do not adversely impact the existing functionalities.
- **Method:** Re-running tests to ensure that new changes have not introduced errors or disrupted previously functioning features.

**8. Continuous Testing and Monitoring:**

- **Purpose:** Ensuring ongoing system reliability and performance post-deployment.
- **Method:** Implementing continuous testing practices and real-time monitoring to identify and address issues promptly.

Testing is done at every step of the creation and implementation of the "IoT Assisted Food Donation and Waste Management Web Application," ensuring its security, robustness, and accuracy while improving its effectiveness in food quality monitoring and donation processes.

**F. Deployment:** Delivery of software, client feedback, and support are all included. adjustments are necessary for such improvements or corrections if a user requests more features or advises certain adjustments. Following user evaluation, spiral implementation, "enhancement plan" and "user's suggestions" come next. This means that the software becomes better with each round of the spiral.

## 7.2. Algorithms

1. **Data Processing and Analysis:**

**Data Validation Algorithm:**

Before processing further, confirms that the incoming sensor data—such as temperature and humidity—is accurate, full, and of high integrity.

**Threshold-based Alerting Algorithm:**

Sets criteria to identify abnormalities or deviations in food quality parameters and initiates alerts or notifications for results that are outside of the acceptable range.

2. **Request Management and Assignment:**

**Matching Algorithm:**

Matches contribution requests with available logistics or agents depending on capacity, location, or urgency using algorithms such as greedy algorithms or heuristics.

**Routing Algorithm:**

Uses graph-based techniques (like Dijkstra's algorithm) to reduce time and resources by optimizing pickup routes for logistics or agents.

3. **Security and Authentication:**

**Encryption Algorithms:**

Uses encryption techniques (such RSA and AES) to protect private information while it's being sent and stored, guaranteeing its confidentiality and integrity.

**Authentication Algorithms:**

Oversees access control to system modules and implements secure authentication methods (such as OAuth and JWT) to verify user credentials.

4. **Data Retrieval and Processing:**

**Search Algorithms:**

Enables the rapid retrieval and manipulation of data contained in databases by implementing effective search algorithms (such as binary search and hash tables).

**Sorting Algorithms:**

Enables the methodical organization and presentation of information by facilitating the sorting of contribution requests, user data, and feedback.

5. **Feedback and Improvement:**

**Feedback Analysis Algorithm:**
Use sentiment analysis or natural language processing methods to assess user input, derive conclusions, and pinpoint areas in need of system improvement.

**Adaptive Algorithms:**
incorporates algorithms that are constantly improving system speed and usability by adapting depending on user interactions and shifting data patterns.

6. **Machine Learning (ML) Algorithms:**

**Regression Models:** Forecasts the shelf life of food by taking into account environmental variables, historical data, and other considerations.

**Classification Models:** Sorts food products according to their expiration date or quality, which helps with inventory control.

# 7.3. Source code

## 7.3.1. Website Frontend Code

```
import "./App.css";
import "react-toastify/dist/ReactToastify.css";
import { useEffect, useContext } from "react";
import { BrowserRouter as Router, Routes, Route } from "react-router-dom";
import { GlobalContex } from "./context/contex";
import { ToastContainer } from "react-toastify";

// components
import Navbar from "./components/Navbar.js";
```

```
// pages
import SignIn from "./pages/Authentication/SignIn.js";
import SignUp from "./pages/Authentication/SignUp.js";
import FogotPassword from "./pages/Authentication/ForgotPassword.js";
import ResetPassword from "./pages/Authentication/ResetPassword.js";
import Home from "./pages/Home";
import Donations from "./pages/Donations/Index.js";
import Logout from "./components/popUp/Logout.js";
import Donate from "./pages/Donate";
import Status from "./pages/Status";
import Profile from "./pages/Profile";
import Requests from "./pages/requests/Index.js";
import Agents from "./pages/Agents/Index.js";
import Donors from "./pages/Donors";
import ProtectedRoutes from "./components/ProtectedRoutes";
import Assigned from "./pages/Assigned/Index";
import AddAgent from "./pages/AddAgent";
import PageNotFound from "./PageNotFound";
import NotificationComponent from "./components/NotificationComponent.js";

// npm package
// import { io } from "socket.io-client";
import axios from "axios";

function App() {
  const {
    setSocketInstance,
    notify,
    showNotificationComponent,
    setNotificationData
  } = useContext(GlobalContex);
  const URL = process.env.REACT_APP_URL;

  // sockets  code
```

```
// const socket = io(URL);
// socket.on("connect_error", (error) => {
//   console.log(error);
// });

// socket.on("private_notification", (data) => {
//   setNotificationData((preVal) => {
//     return { ...preVal, notifications: [data, ...preVal.notifications] };
//   });
// });

const {
  setUserData,
  userLoading,
  showLogoutPopUp,
  setUserLoading,
  setTOKEN,
  TOKEN
} = useContext(GlobalContex);

useEffect(() => {
  if (sessionStorage.getItem("Token")) {
    const isTokenexpired = new Date(TOKEN.expirydate) < new Date();
    if (isTokenexpired) setTOKEN({ token: "", expirydate: new Date() });
  }
  getUser();
}, []);

async function getUser() {
  setUserLoading(true);
  try {
    const response = await axios({
      method: "get",
      url: URL + "/api/user",
```

```
      headers: {
        Authorization: "Bearer " + TOKEN.token
      },
      withCredentials: true
    });
    const user = response.data.data;
    if (response.data.success) {
      setUserData(user);
      setUserLoading(false);

      // setSocketInstance(socket);
      // socket.emit("register", {
      //   role: user.role,
      //   id: user._id,
      //   name: user.firstName + " " + user.lastName,
      //   email: user.email
      // });
    }
  } catch (error) {
    setUserLoading(false);
    // notify(error.response.data.message, "error");
  }
}

return (
  <div className="scroll App h-[100vh]">
    {userLoading ? (
      <div className="hero-particles   h-full w-full"></div>
    ) : (
      <>
        <Navbar />
        <div className=" h-full pt-20">
          <Routes>
            <Route path="/sign_in" element={<SignIn />} />
```

```
      <Route path="/sign_up" element={<SignUp />} />
      <Route path="/forgot_password" element={<FogotPassword />} />
      <Route
        path="/reset_password/:resetPasswordToken"
        element={<ResetPassword />}
      />
      <Route path="/" element={<Home />} />
      <Route element={<ProtectedRoutes />}>
        <Route path="/donations" element={<Donations />} />
        <Route path="/donations/:donationId" element={<Donations />} />
        <Route path="/donate" element={<Donate />} />
        <Route path="/status" element={<Status />} />
        <Route path="/status/:donationId" element={<Status />} />
        <Route path="/requests" element={<Requests />} />
        <Route path="/requests/:donationId" element={<Requests />} />
        <Route path="/profile" element={<Profile />} />
        <Route path="/agents" element={<Agents />} />
        <Route path="/agents/:userId" element={<Agents />} />
        <Route path="/donors" element={<Donors />} />
        <Route path="/donors/:userId" element={<Donors />} />
        <Route path="/assigned" element={<Assigned />} />
        <Route path="/assigned/:donationId" element={<Assigned />} />
        <Route path="/add_agent" element={<AddAgent />} />
      </Route>
      <Route path="*" element={<PageNotFound />} />
    </Routes>
  </div>
  {/* // logout popup */}
  {showLogoutPopUp ? <Logout /> : null}
  {/* // tostify */}
  <ToastContainer />
  {/* showNotificationComponent */}
  {showNotificationComponent ? <NotificationComponent /> : null}
</>
```

```
      )}
    </div>
  );
}

export default App;
```

### 7.3.2. Website Backend Code

```
const express = require("express");
const app = express();
const authRouter = require("./router/authRouter.js");
const donationRouter = require("./router/donationRouter.js");
const notificationRouter = require("./router/norificationRouter.js");
const errorHandler = require("./meddleware/errorHandler.js");
const path = require("path");
const cors = require("cors");
const cookieParser = require("cookie-parser");
const fileUpload = require("express-fileupload");
const userRouter = require("./router/userRouter.js");

// swagger
const YAML = require("yamljs");
const swaggerUi = require("swagger-ui-express");
const swaggerDocument = YAML.load("./Swagger_API_DOCS/swagger.yaml");

app.use(
  cors({
    methods: ["GET", "POST", "DELETE", "UPDATE", "PUT", "PATCH"],
    origin: [process.env.CLIENT_URL, "http://localhost:3000"],
    credentials: true
  })
);
```

```
app.use(cookieParser());
app.use(express.json());
app.use(express.urlencoded({ extended: true }));
app.use(
  fileUpload({
    useTempFiles: true,
    tempFileDir: "/tmp/"
  })
);

// ROUTES
if (process.env.NODE_ENV === "production") {
  app.use(express.static(path.join(__dirname, "view/build")));
  // swagger
  app.use("/api-docs", swaggerUi.serve, swaggerUi.setup(swaggerDocument));
  // rest api
  app.use("/api/auth", authRouter);
  app.use("/api", donationRouter);
  app.use("/api", userRouter);
  app.use("/api", notificationRouter);
  app.get("*", (req, res) => {
    res.sendFile(path.resolve(__dirname, "view", "build", "index.html"));
  });
} else {
  // swagger
  app.use("/api-docs", swaggerUi.serve, swaggerUi.setup(swaggerDocument));
  // rest api
  app.use("/api/auth", authRouter);
  app.use("/api", donationRouter);
  app.use("/api", userRouter);
  app.use("/api", notificationRouter);
  app.use("/", (req, res) =>
    res.status(200).json({ success: true, server: "food donation Backend" })
  );
```

```
}
// ROUTES //

// error handler
app.use(errorHandler);

module.exports = app;
```

### 7.3.3. IOT Code

```
#include <Arduino.h>
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SH110X.h>
#include "max6675.h"

int thermoDO = D7;
int thermoCS = D6;
int thermoCLK = D5;
int ledPin = D4;
int gasPin = A0;

MAX6675 thermocouple(thermoCLK, thermoCS, thermoDO);
float gasLevel = 0.00;
float celsius = 0.00;
float fahrenheit = 0.00;

#define i2c_Address 0x3c
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define OLED_RESET 1
Adafruit_SH1106G display = Adafruit_SH1106G(SCREEN_WIDTH, SCREEN_HEIGHT,
```

```
&Wire, OLED_RESET);

void checkFoodSafety();

void setup() {
  Serial.begin(115200);
  pinMode(ledPin, OUTPUT);

  delay(250); // wait for the OLED to power up
  display.begin(i2c_Address, true); // Address 0x3C default
  display.display();
  delay(2000);
  display.clearDisplay();
}

void loop() {
  celsius = thermocouple.readCelsius();
  fahrenheit = thermocouple.readFahrenheit();
  gasLevel = analogRead(gasPin) / 1023.0 * 100.0;

  Serial.print("Temperature (C): ");
  Serial.println(celsius);  // Display temperature in Celsius on Serial monitor
  Serial.print("Temperature (F): ");
  Serial.println(fahrenheit);
  Serial.print("Gas Level: ");
  Serial.println(gasLevel);

  checkFoodSafety();

  delay(1000); // Adjust delay as needed
}

void checkFoodSafety() {
  Serial.println("Checking food safety...");
```

```
  int expectedGasValue = 5;
  double expectedMinTemperature = 30.0;
  double expectedMaxTemperature = 60.0;

  display.clearDisplay();

   if (gasLevel >= expectedGasValue || celsius < expectedMinTemperature || celsius >
expectedMaxTemperature) {
    Serial.println("Food is not safe!");
    digitalWrite(ledPin, HIGH);
    display.setTextSize(1);
    display.setTextColor(SH110X_WHITE);
    display.setCursor(0, 0);
    display.println("Food is not safe!");
    display.setCursor(0, 10);
    display.println("Inedible");
  }
  else {
    Serial.println("Food is safe!");
    digitalWrite(ledPin, LOW);
    display.setTextSize(1);
    display.setTextColor(SH110X_WHITE);
    display.setCursor(0, 0);
    display.println("Food is safe!");
    display.setCursor(0, 10);
    display.println("Edible");
  }

  if (celsius > 100.0) {
    Serial.println("Temperature is out of range!");
    digitalWrite(ledPin, HIGH);
    display.setTextSize(1);
    display.setTextColor(SH110X_WHITE);
```

```
    display.setCursor(0, 20);
    display.println("Temperature out of range!");
  }

  display.display();
  delay(2000);
  display.clearDisplay();
}
```

# Chapter 8
# Performance Evaluation and Testing

## 8.1. Testing Methodologies:

### 8.1.1. Types of Testing Performed:

1. **Unit Testing:** Testing individual units or components of the application in isolation to ensure they function correctly.
2. **Integration Testing:** Testing the interaction between different modules to verify they work together as expected.
3. **System Testing:** Evaluating the complete system's functionality against the specified 4. requirements.
4. **Performance Testing:** Assessing the system's responsiveness, scalability, and stability under varying loads.
5. **Security Testing:** Verifying the application's resistance to security threats and vulnerabilities.

### 8.1.2. Testing Tools:

1. **Unit Testing Frameworks:** (e.g., JUnit, NUnit) for unit testing individual functions and methods.
2. **Selenium:** For automated testing of web interfaces and functionalities.
3. **Load Testing Tools:** (e.g., Apache JMeter, LoadRunner) to simulate and evaluate system performance under various loads.

### 8.1.3. Test Plans:

Here is a structure for test plans:

1. **Test Case No: 001**

**Description:** Unit Test for User Authentication
**Input:** Valid username and password
**Desired Output:** Successful login message
**Result:** Successful - User authenticated

2. **Test Case No: 002**

**Description:** Integration Test for Donation Request Processing
**Input:** Donation request with valid details
**Desired Output:** Request status updated to "Accepted"
**Result:** Successful - Request status updated correctly

3. **Test Case No: 003**

**Description:** Performance Test for System Load Handling
**Input:** Simulated multiple concurrent donation requests
**Desired Output:** System maintains responsiveness without crashing or significant slowdowns
**Result:** Successful - System handled the load effectively

4. **Test Case No: 004**

**Description:** IoT Sensor Data Validation
**Input:** Sensor readings (temperature, humidity) from IoT devices
**Desired Output:** Valid sensor data within acceptable ranges

**Result:** Successful - Sensor data falls within predefined acceptable thresholds

5. **Test Case No: 005**

**Description:** IoT Data Transmission
**Input:** Sensor data transmission to the application
**Desired Output:** Data received and logged in the system
**Result:** Successful - Sensor data transmitted and recorded without loss

6. **Test Case No: 006**

**Description:** Real-time Monitoring
**Input:** Simulated sensor data indicating critical food quality parameters
**Desired Output:** Instant system alert/notification for critical parameter breach
**Result:** Successful - System triggers an immediate alert for critical parameter deviations

7. **Test Case No: 007**

**Description:** Sensor Connectivity Failure Handling
**Input:** Disconnection or loss of connectivity with IoT sensor
**Desired Output:** Graceful system handling of sensor disconnection, with appropriate error handling or alerts
**Result:** Successful - System handles sensor disconnection without system failure or data loss

8. **Test Case No: 008**

**Description:** IoT Sensor Calibration
**Input:** Adjustment of sensor calibration settings
**Desired Output:** Accurate sensor readings after calibration
**Result:** Successful - Calibrated sensor provides accurate readings within the expected range

9. **Test Case No: 009**

**Description: IoT Device Compatibility**

**Input:** Integration of new IoT devices with the system

**Desired Output:** Seamless integration and data processing from the newly added IoT devices

**Result:** Successful - New IoT devices integrated and data processed effectively

# Chapter 9
# Deployment Strategies

## 9.1. Deployment Strategies:

1. **Pilot Deployment:** To evaluate the system's operation, spot possible problems, and get user input, start with a small-scale pilot deployment in a controlled setting.

2. **Incremental Rollout:** Before scaling up to a larger implementation, deploy the system gradually, concentrating on particular areas or kinds of enterprises (e.g., restaurants, supermarkets).

3. **Collaboration with Stakeholders:** Establish a network of stakeholders and collaborate with neighborhood companies, nonprofit organizations, and governmental organizations to guarantee a more thorough and cooperative deployment.

4. **User Training:** Organize training sessions for users, volunteers, and employees of donation centers to make sure they understand the security procedures and features of the system.

5. **Scalable Infrastructure:** Create a scalable system design that will make it simple to expand when the quantity of users, devices, and data rises.

## 9.2. Security Aspects:

### 9.2.1. Scalable System Architecture:

1. **Flexible Architecture:**

   **Modular Design:** Implement a modular architecture that allows components to be added, removed, or updated independently, facilitating scalability.

   **Scalable Components:** Ensure each module can handle increasing user numbers, devices, and data volumes without impacting the overall system performance.

2. **Cloud Services:**

   **Cloud Infrastructure:** Utilize cloud-based services like AWS, Azure, or Google Cloud for scalable and reliable hosting, enabling the system to handle increased loads efficiently.

   **Auto-scaling:** Implement auto-scaling features to dynamically allocate resources based on demand, ensuring optimal performance during peak usage.

3. **Security Aspects:**

   **Data Encryption:**
   End-to-End Encryption: Implement robust encryption mechanisms to safeguard sensitive data during transmission, ensuring confidentiality and integrity.

4. **Access Control:**

   **Role-Based Access Control (RBAC):** Enforce strict access controls, allowing permissions based on user roles to prevent unauthorized access to critical system components.

5. **Device Authentication:**

**Device Verification:** Authenticate IoT devices before allowing communication with the cloud platform, preventing unauthorized access and potential security breaches.

6. **Secure APIs:**

**API Security Measures:** Implement secure APIs with proper authentication and authorization mechanisms to ensure secure communication between different system components.

7. **Software Updates:**

**Regular Patching:** Ensure all software components are regularly updated with the latest security patches to mitigate potential vulnerabilities and enhance system security.

8. **Data Integrity:**

**Data Verification Mechanisms:** Implement measures like checksums, digital signatures, or blockchain technology to maintain data integrity and prevent tampering.

9. **Incident Response Plan:**

**Comprehensive Incident Plan:** Develop a detailed incident response plan outlining procedures to address and mitigate security breaches promptly.

10. **Privacy Compliance:**

**Regulatory Compliance:** Ensure compliance with data protection laws and regulations, clearly communicating data usage policies to users and obtaining their explicit consent.

11. **Physical Security:**

    **Infrastructure Security:** Implement physical security measures to protect critical infrastructure components from unauthorized access.

12. **Security Audits:**

    **Regular Assessments:** To find and fix possible security flaws early on, conduct regular penetration tests, vulnerability assessments, and security audits.

    **Physical Security:** To prevent unwanted access, it is recommended to implement physical security measures for essential infrastructure components, including data centers and IoT devices.

    **Security Audits:** Perform routine evaluations and audits of security to find and fix any possible weaknesses. Hire outside security professionals to do vulnerability analyses and penetration tests.

In the "IoT Assisted Food Donation and Waste Management Web Application," the system may improve its resilience, secure sensitive data, and fend off possible security threats by using these scalable architectural considerations and thorough security procedures.

# Chapter 10

# Result and Analysis

## 10.1. Experiment Execution:

### 10.1.1. Experimental Setup and Configuration:

**Hardware Configuration:** Describe how to set up an Node MCU-ESP8266 with sensors for humidity, gas, and temperature (MAX 6675, MQ4), as well as the power supply and connectors.

Software Consolidation: Explain the use of the relevant libraries and protocols to incorporate the sensors into the web application (e.g., serial connection between Node MCU-ESP8266 and Node.js server).

1. **Simulation and Testing:**

**Simulated Scenarios:** Describe particular scenarios that were developed to replicate changes in humidity, gas emissions, and temperature under controlled settings or using software-based simulation.

**Donation Process Simulation:** Describe the steps involved in the simulated donation process, such as scheduling, gathering sensor data, making decisions using IoT data, and monitoring.

2. **Data Collection and Logging:**

**Real-time Data Acquisition:** Discuss about the process that was employed to get real-time sensor data for the simulation.

**Data Logging:** Describe the logging techniques used in the experiment to log sensor values and system reactions.

3. **Food Quality Monitoring:**

**Sensor Accuracy Assessment:** Provide an analysis of sensor accuracy by comparing sensor readings against manually measured or predetermined conditions.

**Threshold Analysis:** Discuss the system's ability to detect deviations from preset thresholds and the effectiveness of alerts or notifications for out-of-range values.

4. **Donation Process Evaluation:**

**Efficiency Metrics:** Present data on the time taken for various stages of the donation process, from donation initiation to successful delivery or reception.

**Impact of IoT Data:** Discuss how IoT data influenced decisions related to accepting or rejecting donations and its effect on streamlining the donation process.

5. **Challenges and Limitations:**

**Technical Challenges**: Detail any technical difficulties encountered during the experiment, such as sensor calibration issues, data transmission delays, or integration complexities.

**Operational Limitations:** Discuss limitations observed in the practical implementation or usability of the system.

6. **Conclusions and Recommendations:**

**Summary of Findings:** Summarize the experiment's outcomes, emphasizing the system's strengths and areas needing improvement.

**Recommendations for Improvement:** Provide actionable suggestions for enhancing sensor accuracy, improving system responsiveness, or refining user interfaces based on observed limitations.

## 10.2. Future Scope and Implications:

### 10.2.1. Potential Enhancements:

1. **Advanced Sensor Integration:** Explore the integration of more sophisticated sensors or IoT devices for comprehensive food quality monitoring.

2. **Machine Learning Integration:** Consider implementing machine learning algorithms to optimize decision-making based on collected IoT data.

### 10.2.2. Scalability and Real-world Implementation:

1. **Scalability Analysis:** Discuss the feasibility and challenges of scaling the system for larger deployments or real-world applications.

2. **Practical Implementation Implications:** Address how the experiment's results and insights can be applied in real-world scenarios for effective food donation and waste management.

## 10.3. Applications

**1. Food Waste Reduction:**
Efficient Redistribution: Facilitates the redistribution of surplus food from donors to beneficiaries, reducing food wastage.

Real-time Monitoring: Monitors food quality through IoT sensors, ensuring timely distribution before expiry, thus minimizing wastage.

**2. Hunger Alleviation and Social Impact:**
Enhanced Food Accessibility: Connects donors with charities or individuals facing food insecurity, ensuring a more equitable distribution of resources.

Community Engagement: Involves volunteers, donors, and beneficiaries, fostering community participation in addressing hunger.

**3. Environmental Sustainability:**

Reduced Environmental Footprint: Minimizes the environmental impact associated with food waste by redirecting surplus food from landfills.

Resource Conservation: By reducing waste, the project contributes to conserving resources used in food production, such as water and energy.

**4. Technological Advancement:**

IoT Integration: Demonstrates the application of IoT technology in optimizing food management systems, showcasing its potential in waste reduction initiatives.

Innovation in Food Distribution: Utilizes web applications and IoT sensors to revolutionize food donation practices, setting a precedent for future technological innovations in the field.

**5. Collaboration and Partnerships:**

Stakeholder Collaboration: Fosters partnerships between businesses, charities, government bodies, and volunteers for a collective effort in addressing food insecurity and waste management.

Corporate Social Responsibility: Provides businesses with an avenue to fulfill their corporate social responsibility by contributing surplus food to those in need.

**6. Economic Efficiency:**

Cost Savings: Reduces disposal costs for businesses by redirecting surplus food to donation centers instead of being discarded, enhancing cost-efficiency.

Support for Local Economies: Bolsters local economies by redistributing surplus food within the community, supporting small-scale food-related businesses.

**7. Educational and Awareness Campaigns:**

Educational Outreach: Offers opportunities for educational programs to raise awareness about food waste, hunger issues, and sustainable consumption habits.

Behavioral Change: Encourages individuals and businesses to adopt more responsible food management practices, leading to broader societal shifts in behavior.

## 10.4. Future Scope

There is potential for widespread adoption as the advantages of IoT-assisted trash management and food donation become clear. Further improving interoperability and scalability might be achieved via standardizing protocols and best practices.

- **Integration with Smart Cities Initiatives:**

The project can support urban sustainability by aligning with smart cities programs. Integrating smart city technologies like energy efficiency and traffic management might result in a more complete and effective urban ecosystem.

- **Advanced Analytics and AI Improvements:**

Further developments in artificial intelligence and analytics may improve the system's forecasting power. It's possible that machine learning algorithms may advance in their ability to forecast trends of food waste, streamline logistics, and boost productivity.

- **Blockchain for Transparency and Traceability:**

The process of donating food may become more transparent and traceable with the usage of blockchain technology. This can ensure the validity of food sources and gifts and assist to foster confidence among parties.

- **Wider Range of Supported Devices:**

The system may develop to accommodate a larger range of sensors and smart devices as IoT devices proliferate. This may involve using more sophisticated sensors to track food quality in real time and make sure donated food is still safe to eat.

- **Partnerships with Food Industry Stakeholders:**

Partnerships with food sector participants, including supermarket chains, eateries, and food producers, may grow. Collaborations may entail a smooth interaction with their inventory management systems, resulting in more effective food donation procedures.

- **Community Engagement and Education:**

The adoption of ethical consuming practices should be encouraged by stepping up community involvement and educational programs. Features that instruct users on decreasing food waste and making sustainable decisions may eventually be included to the system.

- **Government Support and Regulations:**

Governments can encourage enterprises to join in such efforts by offering incentives, subsidies, or regulatory frameworks. One major focus may be on adhering to waste reduction and food safety requirements.

- **Enhanced Mobile App Features:**

Other features that might be added to the mobile app in the future include gamification to drive greater engagement, nutritional data for recipients, and real-time feedback on the environmental effect of donations.

- **Environmental Impact Measurement:**

Tools to quantify and illustrate the environmental benefits of minimizing food waste might be incorporated into further iterations of the system. This might help provide a more comprehensive knowledge of the project's beneficial results.

- **Adaptation to Changing Technologies:**

Technology should not be a barrier to the system's adaptability. Integration with new technologies like as 5G may enhance the reliability and speed of data flow, and new opportunities may arise from improvements in IoT sensor technology.

Technological developments, heightened awareness of sustainability concerns, and cooperative efforts from multiple stakeholders are expected to influence the future of an IoT-assisted food donation and trash management system. The long-term effect and success of the initiative may be enhanced by ongoing innovation and a dedication to tackling the problems associated with food waste.

# Conclusion

An important step has been taken in utilizing technology, community cooperation, and sustainable practices to address the urgent problems of food waste and hunger with the launch of the "IoT Assisted Food Donation and Waste Management Web Application". Over the course of its execution, the project has not only achieved its goals but significantly exceeded expectations, having a significant influence on a number of aspects of environment, technology, and society.

**a) Technological Innovation and Integration:**

The project's successful integration of IoT technology has transformed conventional food donation practices. Through the implementation of IoT sensors, the application has facilitated real-time monitoring of food quality, ensuring the timely redistribution of surplus food and mitigating food wastage due to expiration. This innovative integration exemplifies the potential of technology in revolutionizing waste management systems and fostering more efficient food distribution networks.

**b) Socioeconomic Impact:**

By connecting donors, charitable organizations, businesses, and beneficiaries, the application has created a robust ecosystem of collaboration and support. It has significantly enhanced food accessibility for marginalized communities, ensuring a more equitable distribution of resources. Moreover, the platform's engagement of volunteers and stakeholders has cultivated a sense of communal responsibility and empowerment, fostering a societal shift towards collective action in addressing hunger and food sustainability.

**c) Environmental Sustainability:**

The successful diversion of excess food from landfills by the initiative has had a direct positive impact on environmental sustainability. The application has significantly contributed to resource conservation, greenhouse gas emission reduction, and the advancement of a more sustainable method of food management by lowering the environmental impact of food waste.

**d) Collaborative Partnerships and Awareness Initiatives:**

The success of the project is largely dependent on its capacity to promote cooperation amongst a wide range of stakeholders. The groundwork for a coordinated effort to address food hunger and waste management has been built by the partnerships formed with corporations, governmental organizations, charities, and volunteers. Furthermore, by raising public awareness of food waste, the project's educational outreach activities have influenced behavioral shifts toward more sustainably oriented consuming habits.

**e) Future Prospects and Sustainability:**

Scalability and flexibility are still essential to the project's future development. The cloud-based infrastructure and scalable design guarantee the system's stability and ability to handle future expansion. The long-term dependability, security, and relevance of the application will be maintained via regular security assessments, compliance with privacy laws, and technology advancements.

In conclusion, the "IoT Assisted Food Donation and Waste Management Web Application" is a catalyst for social change, environmental stewardship, and sustainable development in addition to being a technological breakthrough. Its significant effects on decreasing food waste, improving food accessibility, encouraging cooperation, and increasing awareness mark a paradigm change in the direction of a society that is more inclusive, sustainable, and compassionate. This project is still a source of optimism and advancement in the worldwide effort to end hunger and reduce waste as it develops.

# References

1. H. Hajj Diab, A. Anzer, H. A. Tabaza and W. Ahmed, "A Food Wastage Reduction Mobile Application," 2018 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), Barcelona, Spain, 2018, pp. 152-157, doi: 10.1109/W-FiCloud.2018.00030.
2. R. Shinta Oktaviana, D. A. Febriani, I. Yoshana and L. R. Payanta, "FoodX, a System to Reduce Food Waste," 2020 3rd International Conference on Computer and Informatics Engineering (IC2IE), Yogyakarta, Indonesia, 2020, pp. 361-365, doi: 10.1109/IC2IE50715.2020.9274576.
3. IS. Patil, O. Nikam, S. Nair, A. Raut and V. B. Lobo, "Sustainable Food Waste Management and Tracking System Using Blockchain," 2023 International Conference on Advancement in Computation & Computer Technologies (InCACCT), Gharuan, India, 2023, pp. 848-853, doi: 10.1109/InCACCT57535.2023.10141799.
4. N. Gayathri, A. R. Divagaran, C. D. Akhilesh, V. M. Ashwini and N. Charan, "IOT Based Smart Waste Management System," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, pp. 2042-2046, doi: 10.1109/ICACCS51430.2021.9441819.
5. N.Alblehed, M. Almutairi, R. Almahmoud, S. Aladhadh and A. Alabdulatif, "Developing Food Charity Operations Management System," 2022 2nd International Conference on Computing and Information Technology (ICCIT), Tabuk, Saudi Arabia, 2022, pp. 93-96, doi: 10.1109/ICCIT52419.2022.9711609.
6. e-ISSN: 2582-5208 International Research Journal of Modernization in Engineering Technology and ScienceVolume:04/Issue:05/May-2022 Impact Factor- 6.752 FOOD WASTE MANAGEMENT SYSTEMS.P Kale*1, Meet Patel*2, Mehtab Ansari*3, Aditi Dhumal*4, Ruchika Arote*5
7. International Research Journal of Modernization in Engineering Technology and Science Volume:04/Issue:11/November-2022 Impact Factor- 6.752 w FOOD WASTE MANAGEMENT USING ANDROID Aashish Khandkar*1, Palomi Gawali*2, Ajay Aswar*3, Yashaswi*4, Yash Satpute*5 DOI : https://www.doi.org/10.56726/IRJMETS31499

8. Tbk, Bharath & Prashar, Deepak. (2021). Review on Efficient Food Waste Management System Using Internet of Things. International Journal of Current Research and Review. 13. 142-149. 10.31782/IJCRR.2021.13603.
9. Varghese, Christina & Pathak, Drashti & Varde, Aparna. (2021). SeVa: A Food Donation App for Smart Living. 10.1109/CCWC51732.2021.9375945.
10. Shazmina Gull, Imran Sarwar Bajwa, Waheed Anwar, Rubina Rashid, "Smart eNose Food Waste Management System", Journal of Sensors, vol. 2021, Article ID 9931228, 13 pages, 2021. https://doi.org/10.1155/2021/9931228

# PART - B

# INDIVIDUAL CONTRIBUTION

**Name of the Student:**

Avinash Shelukar (1032201997)
Ruchik Alhat (1032202149)
Pruthu Prabhudesai (1032212257)
Fahad Malik (1032200260)

**System Components:**

**Software Requirement:**

1. Operating System (Windows 11)
2. REACT JS
3. Database (MongoDb)
4. Node JS
5. Express

**Hardware Requirement:**

1. Arduino Uno R3-
2. Temperature Sensor: K Thermocouple Using MAX 6675
3. Gas Sensor: MQ4
4. Jumper Wires
5. ESP8266-01

## Project to Outcome mapping

Objectives:

1) To implement real time food odour monitoring.
2) To connect donors and beneficiaries.
3) To reach out to more people facing starvation.

| Sr. No. | PRN No. | Student Name | Individual Project Student Specific Objective |
|---|---|---|---|
| 1 | 1032212257 | Pruthu Prabhudesai | To implement real time food odour monitoring. |
| 2 | 1032201997 | Avinash Shelukar | To connect donors and beneficiaries. |
| 3 | 1032202149 | Ruchik Alhat | To connect donors and beneficiaries. |
| 4 | 1032200260 | Fahad Malik | To implement real time food odour monitoring. |

## Module Interfaces

1. **Interfaces**

**A) Functions or Methods Exposed by Components/Web Service:**

a) **IoT Device Interfaces:** These interfaces include features that let Internet of Things devices send sensor data back to the mainframe. For example, protocols or APIs are set up to process and receive temperature, humidity, and expiration date data. The smooth integration and data transfer from sensors to the system's backend for additional processing is ensured by these interfaces.

b) **Data Processing Interfaces:** Processes and procedures in charge of handling incoming data from Internet of Things devices include cleaning, normalization, validation, and storage preparation. Data consistency, integrity, and analysis readiness are guaranteed via these interfaces.

c) **Web Application Interfaces:** Between the front end and back end of the web application, these interfaces act as a link. In addition to handling requests for data updates or retrieval, they also comprise functions that control user interactions and make it easier for the user interface to show analyzed data.

### C) Actions Performed by Components:

a) **IoT Devices:** Ensuring continuous data flow for analysis and decision-making by sending real-time data produced by sensors (temperature variations, humidity levels, expiration alarms) to the central system at predetermined intervals.

b) **Data Processing Modules:** To guarantee correctness and consistency, incoming data must undergo stringent validation tests, data cleaning, and preparation procedures. This include verifying the accuracy of the data, addressing discrepancies, and preparing the data for analysis and storage.

c) **Web Application:** By presenting analytical data in an approachable and dynamic graphical format, users may keep an eye on food quality indicators, control inventory, link donors and recipients, and react to alarms or notifications issued by the system.

## 2. External Interfaces Required:

a) **IoT Sensor Interfaces:** These interfaces cover communication protocols (such HTTP and MQTT), which enable smooth data transfer from sensors to the data intake module of the central system. From external IoT devices to the internal infrastructure, they provide dependable data transport.

b) **Third-party APIs:** interfaces for integrating other systems or services;

examples are logistics APIs for effective transportation planning or weather APIs for adding climatic data that may affect food quality. The functionality and data sources of the application are enhanced by integration with these APIs.

**3. Internal Interfaces Required:**

a) **Data Processing to Database:** Interfaces that control how processed data is moved and stored in databases, guaranteeing that the data is available and persistent for further analysis or retrieval by the web application.

b) **Web Application to Database:** Interfaces that allow the web application to communicate with the database in order to get and show users the appropriate information. These interfaces assist features like inventory management and real-time data visualization by facilitating easy data retrieval and update activities.

c) **Inter-component Communication:** Interfaces inside the system that provide communication between various sections or parts. These interfaces provide effective data sharing and system interoperability by ensuring smooth data flow and processing between components.

**4. Communication Interfaces:**

a) **IoT Protocols:** The communication protocols that IoT devices and the central system employ to transmit data are governed by these interfaces. Secure and effective data sharing is made possible by protocols like HTTP (Hypertext Transfer Protocol) and MQTT (Message Queuing Telemetry Transport), which provide dependable communication and little data loss.

b) **RESTful APIs:** Interfaces for communication used to exchange data between the front-end and back-end servers of a web application. RESTful APIs make it possible to get, edit, and manipulate data, which enables the web interface to efficiently process user requests and provide real-time data.

**5. Graphical User Interfaces (GUI):**

**B) Components of GUI:**

a) **Dashboard**: A complete display that provides inventory status, important warnings, and real-time food quality measurements in an understandable way. Users are given a summary of the overall health and functionality of the system.

b) **Data Visualization Tools:** Users may better grasp trends, patterns, and anomalies in food quality indicators by using graphs, charts, and diagrams that illustrate the examined data. Decision-making and data understanding are improved by these visual aids.

c) **Donor-Beneficiary Connection Interface:** ISmart displays that make it possible for donors to give extra food and make it easier for recipients to access gifts that are available. The contribution process is streamlined by this interface, enabling effective and convenient communication between donors and recipients..

d) **Alert and Notification Panels:** interfaces designed to show messages or alerts generated by the system on changes to the inventory, deviations in food quality, or important events. These panels alert users quickly, allowing for speedy resolution of possible problems.

**B) Actions Supported by GUI:**

a) **Monitoring Food Quality:** Through the use of real-time data visualization from IoT devices, the GUI enables users to analyze trends, monitor metrics related to food quality, and spot any deviations or possible hazards.

b) **Inventory Management:** Users may examine the food products that are available, their amounts, and the status of their distribution using the GUI's inventory management and tracking functions. The technology makes it simple for users to trace the movement of food supplies and organize contributions.

c) **Donor-Beneficiary Interaction:** Donors can enter details on excess food products that are available for donation through the GUI, and recipients can see and claim such donations in accordance with their needs. An easy-to-use interface guarantees that communications between contributors and receivers go smoothly.