A Project on

PRODUCTIVE FOOD WASTE TRACKING APPLICATION

SUBMITTED TO THE SAVITRIBAI PHULE PUNE UNIVERSITY, PUNE IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

ANJALI LONDHE	**Exam No: B191104235**
SAKSHI HOLE	**Exam No: B191104221**
CHETANA PATIL	**Exam No: B191104249**
SAYALI KHAIRE	**Exam No: B191104228**

Under the guidance of

PROF. AMRUTA MORE

Department of Computer Engineering

A.B.M.S.PARISHAD'S

ANANTRAO PAWAR COLLEGE OF ENGINEERING & RESEARCH

Parvati, Pune-09

NAAC Accredited Institute

SAVITRIBAI PHULE PUNE UNIVERSITY

2023-24

AKHIL BHARATIYA MARATHA SHIKSHAN PARISHAD'S
ANANTRAO PAWAR COLLEGE OF ENGINEERING & RESEARCH
Sr. No. 103, Parvati, Pune - 411 009.
Tel. 020-24218901/8959 Tele Fax : 020-24213929
Web.: http://www.abmspcoerpune.org Email : abmspcoe@yahoo.com

Approved by AICTE & Govt. of Maharashtra, Affiliated to Savitribai Phule Pune University
NAAC ACCREDITED, DTE CODE :- EN 6794, AISHE CODE :- C-41484
Savitribai Phule Pune University Identification No. PU/PN/Engg. / 441/2012,

Department of Computer Engineering

CERTIFICATE

This is to certify that the project report entitles

"PRODUCTIVE FOOD WASTE TRACKING APPLICATION."

Submitted by

ANJALI LONDHE — **Exam No: B191104235**
SAKSHI HOLE — **Exam No: B191104221**
CHETANA PATIL — **Exam No: B191104249**
SAYALI KHAIRE — **Exam No: B191104228**

are bonafide students of this institute and the work has been carried out by them under the supervision of **Prof. Amruta More** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degree of **Bachelor of Computer Engineering.**

Prof. Amruta More
Assistant Professor
Computer Engineering Dept.

Anantrao Pawar College of Engineering & Research ★ Pune

Prof. Rama Gaikwad
Head of Department
Computer Engineering Dept.

(Dr. S. B. Thakare) 30-5-24
Principal

Name and Signature of External Examiner

Place: Pune
Date: / /

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to all those who contributed to the successful completion of the "Productive Food Waste Tracking Application" project. This endeavour was made possible through the combined efforts, dedication, and support of many individuals and organizations. First and foremost, we extend our gratitude to our project guide **Prof. Amruta More** for her valuable guidance, expertise and unwavering encouragement throughout the project. Her mentorship was instrumental in shaping the project's direction and ensuring its successful execution.

We are deeply thankful to our team members who demonstrated exceptional teamwork and the HOD of computer engineering department **Prof. Rama Gaikwad**, all teaching and non-teaching faculty of APCOER for their commitment, working tirelessly to bring this project to fruition. Their collaboration and dedication were key to overcoming various challenges and achieving the project's objectives.

We also extend our appreciation to **Dr. Sunil B. Thakre**, Principal, Anantrao Pawar college of engineering and Research for providing us with the necessary resources, infrastructure and access to research materials.

The institution's support was crucial in facilitating our research and development efforts. Our sincere thanks go to the participants who volunteered to be a part of our data collection and testing processes. Their contributions also essential for the accuracy and reliability of our project “Productive Food Waste Tracking Application”. Furthermore, we want to acknowledge the invaluable insights and inspiration we gained from prior research in this field.

Lastly, we extend our gratitude to our friends and family for their patience and understanding throughout this project. Your unwavering support and encouragement motivated us to persevere through the project's various stages. We are truly thankful for all contributions and hope that our "Productive Food Waste Tracking Application" project will have a positive impact in the field of facial recognition and security.

Londhe Anjali Shahu
Hole Sakshi Shivaji
Patil Chetana Nimba
Khaire Sayali Dipak

ABSTRACT

In the contemporary world, food waste has emerged as a critical global issue with profound consequences for the environment, economy, and societal well being. The challenge at hand is the efficient management of food resources to combat this issue. Problem: Food waste, a significant problem, leads to the squandering of valuable resources, contributes to environmental degradation through greenhouse gas emissions, and exacerbates food insecurity, particularly among vulnerable populations. Solution: To address this problem, we have developed an innovative Android mobile application. This application harnesses the capabilities of modern technology to empower individuals and restaurants to contribute surplus food and leftovers to those in need. By bridging the gap between food surplus and food scarcity, our study demonstrates the app's effectiveness in reducing food waste and alleviating food poverty. Through a seamless process of food donation, collection, and redistribution, the application showcases its potential to make a substantial impact on both environmental conservation and social welfare. Conclusion: In a world where food waste remains a pressing global concern, our application offers a scalable and accessible means to redistribute excess food to those who need it most. It fosters a sense of community and shared responsibility in the fight against food waste and hunger while emphasizing the importance of user confidentiality to ensure the security and privacy of all participants in this valuable initiative.

INDEX

1	**Synopsis**	**1**
	1.1 Project Title	2
	1.2 Project Option	2
	1.3 Internal Guide	3
	1.4 Technical Keywords (As per ACM Keywords)	3
	1.5 Problem Statement	4
	1.6 Abstract	4
	1.7 Goals and Objectives	5
	1.8 Relevant mathematics associated with the Project	6
	1.9 Names of Conferences / Journals where papers can be published . .	8
	1.10 Review of Conference/Journal Papers supporting Project idea . . .	8
	1.11 Plan of Project Execution	11
2	**Technical Keywords**	**13**
	2.1 Area of Project	14
	2.2 Technical Keywords	14
3	**Introduction**	**15**
	3.1 Project Idea	16
	3.2 Motivation of the Project	16
	3.3 Literature Survey	17
4	**Problem Definition and scope**	**21**
	4.1 Problem Statement	22
	4.1.1 Goals and objectives	22
	4.1.2 Statement of scope	22
	4.2 Major Constraints	24
	4.3 Methodologies of Problem solving and efficiency issues	25
	4.4 Outcome……	26
	4.5 Applications	27

4.6 Hardware Resources Required 28

5 Project Plan 29

5.1 Project Estimates 31

5.1.1 Reconciled Estimates 31

5.1.2 Project Resources 31

5.2 Risk Management w.r.t. NP Hard analysis 32

5.2.1 Risk Identification 34

5.2.2 Risk Analysis 34

5.2.3 Overview of Risk Mitigation, Monitoring, Management 34

5.3 Project Schedule 36

5.3.1 Project task set 36

6 Software requirement specification 38

6.1 Introduction 39

6.1.1 Purpose and Scope of Document 39

6.1.2 Overview of responsibilities of Developer 39

6.2 Usage Scenario 40

6.2.1 User profiles 40

6.2.2 Use-cases 41

6.2.3 Use Case View 43

6.3 Data Model and Description 44

6.3.1 Data Description 44

6.3.2 Data objects and Relationships 44

6.4 Functional Model and Description 44

6.4.1 Data Flow Diagram 46

6.4.2 Activity Diagram: 46

6.4.3 Non Functional Requirements: 47

6.4.4 State Diagram: 49

6.4.5 Design Constraints 49

6.4.6 Software Interface Description 50

7 Detailed Design Document using Appendix A and B 52

7.1 Introduction 53

7.2 Architectural Design 53
7.3 Data design (using Appendices A and B) 54
7.3.1 Internal software data structure 56
7.3.2 Global data structure 56
7.3.3 Temporary data structure 56
7.3.4 Database description 57
7.4 Component Design 58
7.4.1 Class Diagram……58

8 Project Implementation 59
8.1 Introduction 60
8.2 Tools and Technologies Used 60
8.3 Methodologies/Algorithm Details 61
8.4 Verification and Validation for Acceptance 61

9 Software Testing 64
9.1 Type of Testing Used 65
9.2 Test Cases and Test Results 68

10 Results 96
10.1 Screen Shots……74
10.2 Outputs..... ……..76

11 Deployment and Maintenance 78
11.1 Installation and un-installation…… 79
11.2 User Help…… 79

12 Conclusion and future scope 81

Annexure A References 84

Annexure B Competition participation Certificates (All Members) 88

Annexure C Paper, Certificate, Reviewers Comments of Paper Submitted 91

Annexure D Plagiarism Report 109

Annexure E Information of Project Group Members 114

Annexure F Final Project Presentation Handouts 117

Annexure G Project Achievements 126

Annexure H Documentation on Step by Step Execution of a Project 128

Annexure I System Code 134

List of Figures

6.1 Use case diagram 43

6.2 Data Flow diagram 46

6.3 Activity diagram 46

6.4 Receiver's Activity Diagram…… 47

6.5 State Diagram……49

7.1 Architecture Diagram……53

7.2 Class Diagram……...58

10.1 Welcome Page……..74

10.2 Login Page…….. 74

10.3 Registration Page…… 74

10.4 Home Page…….. 74

10.5 Donar's Page & location…… 75

10.6 Receiver's Page & Location…… 75

10.7 Donar's & Receiver Location……. 75

10.8 Contact Us…….. 75

10.9 History…… 76

10.10 About Us…… 76

10.11 Image Prediction…… 76

10.12 Bad Rice Prediction…… 76

10.13 Good Roti Prediction…… 77

10.14 My Pins-Donar's or Receiver's location……77

.

List of Tables

1.1 Review Of Conference…… 9

3.1 Literature Survey……. 20

4.1 Hardware Requirements…… 27

5.1 Project Estimate…… 31

5.2 Risk Identification…… 34

5.3 Risk Analysis…… 34

5.4 Risk Mitigation…… 34

5.5 Risk Monitoring…… 42

6.1 Use cases…… 42

9.1 Test Cases For Registration Page……... 74

9.2 Test Cases For Login Page…… 74

.

LIST OF ABBREVATIONS

ABBREVIATION	ILLUSTRATION
DFD	Data Flow Diagram
UI	User Interface
IP	Internet Protocol
VPC	Virtual Private Cloud
GDPR	General Data Protection Regulation
WCAG	Web Content Accessibility Guidelines

CHAPTER 1

SYNOPSIS

1.1 PROJECT TITLE

Productive Food Waste Tracking Application

1.2 PROJECT OPTION

1. **Biogas Production**

   Description: Develop a system that converts food waste into biogas for energy production.

   Steps:

   - Collect food waste from local restaurants, households, and markets.
   - Process the waste in anaerobic digesters to produce biogas.
   - Use the biogas for electricity generation or as a cooking fuel.
   - Utilize the remaining digestate as fertilizer.

   Benefits: Reduces landfill waste, generates renewable energy, and provides organic fertilizer.

2. **Composting Programs**

   Description: Establish community composting initiatives to turn food waste into valuable compost for agriculture.

   Steps:

   - Set up composting bins in neighborhoods, schools, and parks.
   - Educate the community on separating organic waste.
   - Manage the composting process to ensure high-quality compost.
   - Distribute the compost to local farms, gardens, and green spaces.

   Benefits: Enhances soil health, reduces methane emissions from landfills, and promotes community engagement.

3. **Nutrient Recovery**

   Description: Extract valuable nutrients from food waste for use in agriculture or dietary supplements.

   Steps:

   - Collect food waste rich in nutrients (e.g., fruit and vegetable scraps).
   - Use technologies like hydrolysis, fermentation, or enzymatic treatment to extract nutrients.
   - Purify and concentrate the nutrients.
   - Develop products such as organic fertilizers, soil conditioners, or supplements.

1.3 INTERNAL GUIDE

Prof. Amruta More

1.4 TECHNICAL KEYWORDS

H. Information Storage And Retrieval

A. Search process

B. Information filtering

C. Clustering

D. Relevance feedback

D. Software

D. SOFTWARE ENGINEERING

i. D.2.1 Requirements/Specifications

A. Elicitation methods

B. Methodologies

H.5 Information Interfaces and Presentation (e.g., HCI)

(a) H User Interfaces

A. User-centered design

B. Screen design

C. Evaluation/methodology

I. Computing Methodologies

(a) I.2 ARTIFICIAL INTELLIGENCE

i. Applications and Expert Systems

A. Natural language interfaces

B. Office automation

K. Computing Milieux

K.4 COMPUTERS AND SOCIETY

i. K.4.1 Public Policy Issues

A. Environmental protection

1.5 PROBLEM STATEMENT

Food waste is a substantial and pressing issue with far-reaching consequences. When edible food is needlessly discarded, it not only represents the wastage of valuable resources like water, energy, and labor but also contributes significantly to environmental harm. Food rotting in landfills generates greenhouse gases, contributing to climate change. Moreover, this wasteful practice worsens food insecurity, leaving vulnerable populations without access to adequate nutrition. Food waste is a complex challenge that impacts sustainability, the environment, and social equity, making it imperative to address.

1.6 ABSTRACT

In the contemporary world, food waste has emerged as a critical global issue with profound consequences for the environment, economy, and societal well being. The challenge at hand is the efficient management of food resources to combat this issue. Problem: Food waste, a significant problem, leads to the squandering of valuable resources, contributes to environmental degradation through greenhouse gas emissions, and exacerbates food insecurity, particularly among vulnerable populations. Solution: To address this problem, we have developed an innovative Android mobile application. This application harnesses the capabilities of modern technology to empower individuals and restaurants to contribute surplus food and leftovers to those in need. By bridging the gap between food surplus and food scarcity, our study demonstrates the app's effectiveness in reducing food waste and alleviating food poverty. Through a seamless process of food donation, collection, and redistribution, the application showcases its potential to make a substantial impact on both environmental conservation and social welfare. Conclusion: In a world where food waste remains a pressing global concern, our application offers a scalable and accessible means to redistribute excess food to those who need it most. It fosters a sense of community and shared responsibility in the fight against food waste and hunger while emphasizing the importance of user confidentiality to ensure the security and privacy of all participants in this valuable initiative.

Key Words: Food waste, environmental degradation, food insecurity, Android mobile application, surplus food, food redistribution, greenhouse gas emissions, vulnerable populations, community, user confidentiality, sustainable solution, global concern.

1.7 GOALS AND OBJECTIVES

Goals:

The overarching aim of our study is to develop a comprehensive solution for food waste management and redistribution using our Android mobile application. Specifically, we goal to:

1. **Reduce Food Waste:** Our primary aim is to significantly reduce food waste at the individual and restaurant levels by encouraging donations of surplus food.
   ➢
2. **Alleviate Food Insecurity:** We seek to alleviate food insecurity and hunger by efficiently redistributing surplus food to beneficiaries, such as low-income individuals and charitable organizations.
3. **Promote Sustainability:** Our goal is to contribute to environmental sustainability by minimizing the environmental impact associated with food waste, including reduced greenhouse gas emissions.

Objectives:

1. **Develop the Mobile Application:** Design and create a user-friendly Android mobile application that enables users to easily donate excess food and allows agents to efficiently collect and redistribute it.
2. **Foster Community Engagement:** Promote community engagement and participation by encouraging individuals and restaurants to actively use the application for food donation and collection.
3. **Assess Impact:** Measure the impact of the application in terms of the amount of food saved from being wasted, the number of beneficiaries served, and the reduction in greenhouse gas emissions associated with food waste.
4. **Ensure User Confidentiality:** Implement robust security measures within the application to safeguard user information and maintain their privacy.
5. **Evaluate Sustainability:** Evaluate the environmental sustainability of the system by analyzing its effectiveness in reducing food waste and associated ecological footprint.

By achieving these objectives and aims, we aim to develop a holistic solution that not only addresses the critical issue of food waste but also contributes to food security and environmental well-being while maintaining user confidentiality and security.

1.8 RELEVANT MATHEMATICS ASSOCIATED WITH THE PROJECT

1. **System Description:** Application is designed to track, manage, and reduce food waste by leveraging various mathematical and computational techniques. Here is a detailed breakdown of the system's mathematical framework:
2. **Input:**

   - **User Data:** Information about household size, dietary preferences, and shopping habits.
   - **Inventory Data:** List of purchased food items, quantities, purchase dates, and expiration dates.
   - **Waste Data:** Logs of discarded food items, reasons for waste, and timestamps.
   - **User Interaction Data:** Usage patterns and engagement metrics.

3. **Output:**

   - **Notifications:** Alerts for items nearing expiration and recipe suggestions.
   - **Reports:** Weekly, monthly, and yearly summaries of food usage and waste.
   - **Recommendations:** Personalized recipes and shopping lists.
   - **Analytics:** Insights into food waste patterns and trends.

4. **Data Structures and Classes:**

   - **Food Item Class:** Represents individual food items with attributes like name, quantity, purchase date, expiration date, and category.
   - **Waste Log Class:** Logs discarded items with attributes for item details, waste reasons, and timestamps.
   - **User Class:** Stores user-specific information and preferences.
   - **Recipe Class:** Represents recipes with attributes for ingredients, instructions, and tags.
   - **Notification Class:** Manages alerts and reminders for the user.

5. **Divide and Conquer Strategies:**

- **Distributed/Parallel Processing:**

- **Data Analytics:** Parallelize the processing of large datasets for generating insights and recommendations.

6. **Functions:**

- **Overloading in Functions:** Methods for adding food items can be overloaded to accept different input formats (manual entry, barcode scan).
- **Functional Relations:** Relations between food items and recipes (ingredient inclusion), waste logs and food items (cause of waste), and user preferences and recommendations.

7. **Mathematical Formulation:**

- **Optimization Problem:** Minimize the total food waste (W) over time (T), subject to constraints on inventory capacity and user preferences.
  - min⁡X∑t=1TW(t)\min_{X} \sum_{t=1}^{T} W(t)Xmint=1∑TW(t)
  - Where W(t)W(t)W(t) is the waste at time ttt and XXX is the set of all decision variables (e.g., purchase quantities, consumption rates).

8. **Success Conditions:**

- **Reduction in Food Waste:** Significant decrease in the amount of food wasted by users over time.
- **Positive Feedback:** User satisfaction with the app's recommendations and insights, reflected in user reviews and feedback.

9. **Failure Conditions:**

- **Data Inaccuracy:** Incorrect or incomplete data entry leading to unreliable recommendations.
- **Low Adoption:** Insufficient user engagement or failure to reach a critical mass of users.
- **Technical Issues:** Problems with app performance, such as slow processing times or frequent crashes.

1.9 NAMES OF CONFERENCES / JOURNALS WHERE PAPERS PUBLISHED

Paper ID: 978-81-954164-4-8

Paper Title: Productive Food Waste Tracking Application

Paper Status: Published in "Journal of Multimedia Technology & recent advancement"

Name: Sixth International Conference on Recent Trends in Engineering, Management, Pharmacy and Science (SAGECON2K24)

1.10 REVIEW OF CONFERENCE/JOURNAL PAPERS SUPPORTING PROJECT IDEA

Sr No.	Paper Title	Authors	Description
1	"Towards a Smartphone Application for Food Waste Tracking and Reduction"	John Doe, Jane Smith	This paper discusses the development of a mobile app aimed at tracking food waste in households and providing users with suggestions to reduce waste.
2	"Smart Kitchen: IoT-Based Food Waste Management System"	Alice Brown, Bob Johnson	Explores the integration of IoT devices with a mobile application to monitor food usage and minimize waste through real-time tracking and alerts.
3	"Design and Implementation of a Food Waste Monitoring System"	Michael Green, Sarah Lee	Describes the design and implementation of a system for monitoring food waste in residential kitchens using a mobile app and embedded sensors.
4	"A Mobile Application for Reducing Food Waste in Urban Households"	Olivia Brown, Liam Johnson	Details the development of an Android app focused on reducing food waste in urban households by tracking purchases, consumption, and leftovers.
5	"Food Rescue: An App-Based Approach to Connect Surplus Food with Those in Need"	Sophia Williams, James Martinez	Discusses an application designed to connect food donors with local charities, aiming to reduce food waste and support community food security.

6	"Analyzing the Effectiveness of Mobile Apps in Reducing Food Waste in University Settings"	David Wilson, Emily Taylor	Evaluates the effectiveness of a food waste tracking app used in university dining facilities, highlighting user engagement and waste reduction metrics.
7	"Sustainable Eating: Encouraging Food Waste Reduction Through Mobile Technology"	Ethan Harris, Ava Rodriguez	Examines how mobile technology can promote sustainable eating habits and reduce food waste, including app features that encourage mindful consumption.

Table 1.1 Review Of Conference

Paper ID: ICCIDA/MAY-2024/

Paper Title: Food Waste Manage System

Paper Status: Accepted

Conference Name: International conference on computational intelligence and Data analytics (ICCIDA2024)

REVIEW OF CONFERENCE/JOURNAL PAPERS SUPPORTING PROJECT IDEA

Sr No.	Paper Title	Authors	Description
1	"A Machine Learning Approach to Food Waste Prediction in Commercial Kitchens"	J. Doe, A. Smith, B. Johnson	This paper presents a machine learning model to predict food waste in commercial kitchens, focusing on data from orders, preparation, and waste records.
2	"Optimizing Food Waste Management Using Deep Learning"	M. Zhang, K. Liu, L. Wang	Discusses a deep learning approach to optimize food waste management

			processes in the hospitality industry, utilizing convolutional neural networks (CNNs) for prediction.
3	"Intelligent Food Waste Monitoring System Using IoT and Machine Learning	S. Lee, J. Kim, H. Park	Integrates IoT sensors with machine learning algorithms to monitor and predict food waste in real-time, aiming to reduce overall waste in household settings.
4	"Predictive Analytics for Food Waste Reduction in Retail"	A. Brown, C. Davis, E. Wilson	Uses predictive analytics and machine learning techniques to forecast food waste in retail environments, helping to adjust inventory and reduce excess stock.
5	"Smart Bin: A Machine Learning-Based Food Waste Classification System"	R. Garcia, M. Patel, T. Hernandez	Introduces a smart bin system that uses machine learning to classify and manage food waste automatically, improving sorting efficiency and recycling rates.

1.11 PLAN OF PROJECT EXECUTION

1. Planning and Requirements Gathering

- Conduct initial project kickoff meeting to align on vision and objectives.
- Gather detailed requirements through user interviews, surveys, and market research.
- Create and review initial wireframes and prototypes for feedback.
- Finalize requirements specification document.

2. Design

- Develop detailed UI/UX design mockups.
- Review and refine designs with stakeholder input.
- Design system architecture, including technology stack and modules.
- Create database schema and define data relationships.
- Plan for scalability and security measures.

3. Development

- Set up development environments for frontend and backend.
- Implement core features of the frontend using React Native.
- Develop backend services using Node.js and Express.
- Integrate database with MongoDB or PostgreSQL.
- Implement barcode scanning and inventory management features.
- Develop notification and alert system for expiring items.
- Implement recipe recommendation engine.
- Create analytics and reporting features.
- Conduct initial testing and debugging of individual modules.

4. Testing and Refinement

- Recruit beta testers and conduct testing sessions.
- Collect and analyze feedback from beta testers.
- Identify and prioritize bugs and feature requests.
- Implement fixes and enhancements based on feedback.
- Conduct thorough performance testing and optimization.
- Prepare detailed user documentation and support materials.

5. Deployment and Launch

- Perform final system checks and validation.
- Set up production environment and deploy the application.
- Execute marketing and user acquisition plan for the official launch.
- Monitor the system post-launch for any critical issues.
- Provide ongoing support and maintenance.
- Plan for future updates and feature enhancements.

CHAPTER 2

TECHNICAL KEYWORDS

2.1 AREA OF PROJECT

- **Deep Learning and Artificial Intelligence**: Build predictive models to estimate food waste and recommend improved products.
- **Internet of Things**: Integrate IoT devices to instantly track food inventory and waste. Increase supply chain transparency and efficiency and reduce food waste. :
- **Supply chain management**: Reduce food waste to reduce the carbon footprint and environmental impact of your food waste. User-friendly interface for tracking, reporting and data entry of food waste. Behaviors to create interventions to reduce food waste.

2.2 TECHNICAL KEYWORDS

1) **Information Systems**

   1 DATABASE MANAGEMENT

   - Distributed databases
   - Transaction processing

2) **ARTIFICIAL INTELLIGENCE**

   - Machine learning
   - Convolutional Neural Networks

3) **Computer Applications**

   1 Social and Behavioural Sciences

   - Sociology
   - Economics

4) **Computing Milieux**

   1. COMPUTERS AND SOCIETY

   2. Public Policy Issues

   - Environmental issues
   - Sustainability

CHAPTER 3

INTRODUCTION

3.1 PROJECT IDEA

Introducing Food Guard, a mobile application designed to help users track, manage, and reduce food waste effectively. The app allows users to input their groceries through barcode scanning or manual entry, logging purchase dates and expiration dates to generate timely alerts for items nearing expiration. It features a waste logging system that helps users identify patterns in their food disposal habits, offering detailed analytics and insights to encourage better consumption practices. Food Guard suggests recipes based on available ingredients, prioritizing those close to expiry, and integrates shopping lists to prevent over-purchasing. Additionally, the app fosters a community environment where users can share tips and participate in waste reduction challenges, promoting a sustainable lifestyle through gamification and reward systems. Utilizing React Native for a seamless cross-platform experience, alongside robust backend support from Node.js and MongoDB, Food Guard aims to make reducing food waste both engaging and intuitive, ultimately contributing to environmental sustainability and household savings.

3.2 MOTIVATION OF THE PROJECT

The motivation behind creating Food Guard stems from the urgent need to address the pervasive issue of food waste, which has significant environmental, economic, and social implications. Globally, approximately one-third of all food produced is wasted, contributing to massive carbon emissions and unnecessary strain on natural resources. This waste occurs at every stage of the food supply chain, but a substantial portion happens at the consumer level due to poor planning, over-purchasing, and lack of awareness about food storage and expiration. By empowering individuals with a tool that provides real-time tracking, reminders, and practical solutions, Food Guard aims to foster mindful consumption and reduce household food waste. The app's educational components, community features, and gamification elements are designed to transform user behavior, making sustainability a convenient and rewarding part of daily life. Ultimately, FoodGuard aspires to create a positive impact by helping users save money, reduce their environmental footprint, and contribute to a more sustainable food system.

3.3 LITERATURE SURVEY

Sr.No	Authors	Year	Journal	Method	Key Findings
1	Smith, J. D., & Wang, L.	2023	Food Research International	Structural Equation Modelling (SEM)	Factors influencing food waste in Harbin, categorized as one-person food and multi-person food, were empirically analysed using SEM.
2	Gupta, A., & Patel, S.	2022	Sustainability	Block chain	Development of a sustainable food waste management and tracking system using block chain technology.
3	Smith, E.	2021	Unpublished application project	Android Studio	Details about an unpublished food wastage management application developed using Android Studio.
4	Wang, S. & Xu, J.	2022	International Conference on CoST	AHP-TRIZ Method	Design of an intelligent household food waste product using the AHP-TRIZ method
5	Bhardwaj, S., Kumar, U., & Kumar	2022	Proceedings of Advancement in Electronics &	Not specified	Description of a food waste management

			Communication Engineering		Android app without specific methodological details.
6	Prova, R. R., Rayhan, A., Shilon, R. S., & Khan, M. M.	2021	ICCCNT	Web and Mobile Based Approach	Development of a web and mobile-based approach to redistribute consumable food waste.
7	Sardar Maran, P., Reddy, B. S., & Saiharsha vardhan, C.	2021	Lecture Notes in Electrical Engineering	Not specified	Mention of an IoT-based air quality prediction system, not directly related to food waste.
8	N. K. G N., J. R., S. S. Nukala, M. K. V., S. P. Shankar, & S. Kandarp	2021	IEEE Mysore Sub Section International Conference	Efficient Hunger Search Techniques	Development of a leftover food management system using efficient hunger search techniques.
9	Apostolidis, C., Brown, D., Wijetunga, D., & Kathriarachchi	2021	Journal of Marketing Management	Mobile Applications	Use of mobile applications to reduce food waste and improve food security, particularly targeting the bottom of the pyramid.
10	Vidhi Panchal, Kajal Kuchekar, & Snehal Tambe	2020	International Research Journal of Engineering and Technology (IRJET)	Not specified	Availability of food for NGOs through a mobile application.
11	Iftekhar, A., Cui, X., Hassan, M., & Afzal, W.	2020	Journal of Food Quality	Block chain and IoT	Application of block chain and IoT for ensuring

					tamper-proof data availability for food safety.
12	FAO, The Food and Agriculture Organization	2019	Online at: FAO Food Loss and Waste Data	Data Review	Provides a comprehensive overview of food loss and waste data.
13	G.T. Tucho, T. Okoth	2020	Journal of Cleaner Production	Evaluation of neglected bio-wastes potential with a focus on food-energy sanitation nexus	Evaluates the potential of neglected bio-wastes in the context of food-energy sanitation nexus.
14	A. Sorokowska, M. Marczak, M. Misiak, M.M. Stefaczyk, P. Sorokowski	2020	Journal of Environmental Psychology	Experimental study on attitudes towards food wasting behavior in children and adults	Shows that children older than five years do not approve of wasting food, highlighting attitudes towards food-wasting behavior.
15	D. Leverenz, S. Moussawel, C. Maurer, G. Hafner, F. Schneider, T. Schmidt, M.	2019	Resources, Conservation & Recycling	Quantifying the prevention potential of avoidable food waste in	Quantifies the potential for preventing avoidable food waste in households using a self-

	Kranert			households	reporting approach.
16	P. Sakthi, S. Jagannath, N. Suman, A. Sakthivel, A. Nandhini	2022	International Journal of Research Publication and Reviews	Survey on Waste Food Management and Donating Web Application	Provides insights into a web application for managing and donating surplus food resources.

Table 3.1 Literature Survey

CHAPTER 4

PROBLEM DEFINITION AND SCOPE

4.1 PROBLEM STATEMENT

Food waste is a significant issue globally, with substantial economic, environmental, and social implications. Businesses in the food industry, including restaurants, grocery stores, and households, often struggle to manage inventory efficiently, leading to unnecessary waste. The lack of real-time data, predictive insights, and user-friendly tracking systems exacerbates the problem, resulting in financial losses and increased environmental impact. The current methods for tracking and managing food inventory are often manual, fragmented, and inefficient, leading to high levels of food waste. There is a critical need for a comprehensive solution that leverages modern technology to provide real-time tracking, predictive analytics, and actionable insights to minimize food waste. The solution should be user-friendly, scalable, and integrable with existing systems to ensure wide adoption and significant impact.

4.1.1 Goals and objectives

1. **Real-Time Tracking**: Develop an application that enables real-time tracking of food inventory using IoT devices and manual input, providing up-to-date information on stock levels, storage conditions and expiration dates .

2. **Predictive Analytics**: Implement machine learning and deep learning algorithms to analyze and predict future food waste and historical data , offering actionable recommendations for inventory management and procurement practices.

3. **User Engagement**: Create a user-friendly interface that simplifies tracking and data entry for diverse users, including restaurant managers, grocery store staff, and consumers.

4. **Integration:** Ensure the application can supply chain systems to facilitate data exchange and process automation and seamlessly integrate with existing inventory management and

5. **Sustainability**: Promote sustainable practices by providing insights and recommendations that help reduce food waste, thereby lowering the environmental footprint of food-related businesses.

4.1.2 Statement of scope

- **Description of the Software**

The productive food waste tracking application is designed to help businesses and households reduce food waste through real-time tracking, predictive analytics, and actionable insights. The software will leverage IoT devices for monitoring storage conditions, integrate with existing inventory systems, and provide an intuitive interface for users to manage food inventory efficiently.

- **Size of Input**

Inventory data: Item names, quantities, expiration dates, storage conditions. Environmental data: Temperature, humidity readings from IoT sensors. Purchase dates, consumption patterns.User inputs: Manual data entry, adjustments, and logs.

- **Volume of Input Data:**

Small to large datasets, depending on the size of the business or household. Expected to handle data ranging from a few dozen items for small businesses/households to thousands of items for large enterprises.

- **Bounds on Input**

- **Inventory Items:**
  - Minimum: 1 item.
  - Maximum: Dependent on storage capacity and business scale, typically up to tens of thousands of items.
- **Environmental Data:**
  - Temperature range: -40°C to +50°C.
  - Humidity range: 0% to 100%.

- **Input Validation**

- **Inventory Data:** Ensure item names are non-empty strings. Quantities must be non-negative integers. Expiration dates must be valid dates and not in the past.
- **Environmental Data:** Validate temperature and humidity readings fall within plausible ranges. Sensor data must be current and timestamped correctly.
- **User Inputs:** Validate all manual entries for completeness and correctness. Check for logical consistency (e.g., consumed quantity cannot exceed available quantity).

4.2 MAJOR CONSTRAINTS

1. **Data Volume and Scalability:**

   Constraint: The application must handle large volumes of data, particularly for larger businesses or enterprises with extensive inventories.

   Impact: Necessitates scalable cloud infrastructure and efficient data processing algorithms to manage high throughput and ensure real-time performance.

2. **Data Accuracy and Consistency:**

   Constraint: Accurate and consistent data entry is crucial for reliable tracking and predictive analytics.

   Impact: Implement comprehensive input validation and error-checking mechanisms to ensure data integrity.

4. **Predictive Analytics Accuracy:**

   Constraint: Predictive models must be accurate to provide valuable insights and recommendations.

   Impact: Necessitates ongoing model training and validation using diverse and up-to-date datasets to maintain high prediction accuracy.

6. **User Interface and Experience:**

   Constraint: The application must be user-friendly and intuitive to ensure high user adoption and engagement.

   Impact: Requires investment in user experience (UX) design and iterative testing with real users to refine the interface and workflows.

10. **Development Timeline:**

    Constraint: The project must be completed within a specified timeframe to meet market demands or stakeholder expectations.

    Impact: Requires efficient project management and potentially limits the scope of features in the initial release.

11. **Testing Environment:**

    Constraint: The application must be tested across multiple environments and conditions to ensure reliability and robustness.

    Impact: Requires extensive testing infrastructure and the development of comprehensive test cases to cover various scenarios and edge cases.

4.3 METHODOLOGIES OF PROBLEM SOLVING AND EFFICIENCY ISSUES

Problem Solving Approaches

1. **Rule-Based Systems**
    - **Description:** Implement predefined rules to manage inventory and track food waste. Rules could include thresholds for expiry dates, storage conditions, and inventory levels.
    - **Performance Parameters:**
        - **Simplicity:** Easy to implement and understand.
        - **Efficiency:** Fast processing since it involves simple condition checks.
        - **Scalability:** May become cumbersome as the number of rules increases.
    - **Efficiency Issues:**
        - Limited flexibility and adaptability to new patterns.
        - Difficulty in handling complex scenarios and large datasets.
2. **Machine Learning Models**
    - **Description:** Use historical data to train models that predict food waste and recommend optimal inventory management strategies.
    - **Performance Parameters:**
        - **Accuracy:** Higher accuracy with more data and advanced algorithms.
        - **Adaptability:** Can learn and adapt to new patterns over time.
        - **Scalability:** Capable of handling large datasets effectively.
    - **Efficiency Issues:**
        - Requires substantial computational resources for training and inference.
        - Data quality and volume significantly impact performance.
        - Model interpretability can be challenging..
3. **Hybrid Approach**
    - **Description:** Combine rule-based systems, machine learning models, and IoT-based real-time monitoring to leverage the strengths of each methodology.
    - **Performance Parameters:**
        - **Comprehensive Coverage:** Addresses a wide range of scenarios and conditions.
        - **Robustness:** Enhanced reliability through multiple data sources and methodologies.
        - **Flexibility:** Adapts to different user needs and contexts.
    - **Efficiency Issues:**

- Increased complexity in system design and implementation.
- Higher resource requirements for integrating and managing multiple systems.
- Potential challenges in ensuring seamless communication and data consistency.

4.4 OUTCOME

Economic Outcomes

1) **Cost Savings:**
   - **Reduced Waste:** By optimizing inventory management and reducing food spoilage, businesses can significantly decrease the cost associated with wasted food.
   - **Efficient Procurement:** Improved forecasting and inventory tracking lead to more accurate purchasing decisions, reducing over-ordering and associated costs.
2) **Improved Inventory Management:**
   - **Real-Time Monitoring:** Continuous tracking of inventory and environmental conditions leads to more proactive and effective inventory management.
   - **Predictive Analytics:** Data-driven insights enable businesses to anticipate and mitigate potential food waste issues before they occur.
3) **Enhanced Decision-Making:**
   - **Data-Driven Insights:** Access to comprehensive data and analytics helps managers make informed decisions regarding procurement, storage, and distribution.
   - **Automated Alerts:** Real-time alerts for expiring items or adverse storage conditions enable timely interventions.
4) **Integration and Scalability:**
   - **Seamless Integration:** The application's ability to integrate with existing systems ensures a smooth transition and interoperability with current workflows.
   - **Scalability:** Designed to accommodate businesses of varying sizes, from small households to large enterprises, ensuring broad applicability.
5) **Consumer Awareness:**
   - **Educational Resources:** Provides users with information and tips on reducing food waste, promoting more sustainable consumption habits.

- **Behavioral Change:** Encourages consumers and businesses to adopt more mindful and responsible food management practices.

6) **Community Impact:**

- **Food Donation:** Identifies opportunities for redirecting surplus food to community organizations and food banks, helping to address food insecurity.
- **Corporate Social Responsibility (CSR):** Enhances the CSR profile of businesses by demonstrating a commitment to sustainability and waste reduction.

4.5 APPLICATIONS

- **Inventory Management:** Helps restaurants track inventory levels in real-time, predict shortages, and avoid over-ordering.
- **Waste Reduction:** Monitors food spoilage and waste patterns, providing actionable insights to reduce waste.
- **Menu Planning:** Analyzes sales and waste data to optimize menu offerings and portion sizes, reducing surplus.
- **Compliance:** Ensures compliance with food safety regulations by monitoring storage conditions and expiration dates.
- **Perishable Goods Management:** Tracks expiration dates and provides alerts for soon-to-expire items, enabling timely sales or donations.
- **Consumer Engagement:** Offers consumers information about the freshness of products and potential discounts on items nearing expiration.
- **Personal Inventory Management:** Helps households track their food inventory, plan meals, and avoid overbuying.
- **Educational Tools:** Provides tips and resources on how to store food properly and reduce waste.
- **Sustainability Initiatives:** Supports campus sustainability programs by providing data on food waste and promoting waste reduction practices.
- **Hospital Kitchens:** Assists in managing food inventory in hospitals, ensuring patients receive fresh and safe meals while minimizing waste.

4.5 HARDWARE RESOURCES REQUIRED

Sr. No.	Parameter	Minimum Requirement	Justification
1	CPU Speed	2 GHz	Remark Required
2	RAM	3 GB	Remark Required

Table 4.1: Hardware Requirements

4.6 SOFTWARE RESOURCES REQUIRED

1. **Platform:**

   Operating System: Any modern operating system compatible with the require development tools and libraries. Options include:

   - **Windows**
   - **macOS**
   - **Linux (e.g., Ubuntu, Fedora)**

2. **Development Environment:**

   IDE (Integrated Development Environment): An IDE suitable for programming and development tasks. Recommended options include:

   - Android

3. **Programming Language:**

   Programming Language: The choice of programming language depends on the specific requirements and technologies used in your project. For this project, the following languages are commonly used:

- Java: Often used for implementing the backend part of the application.

CHAPTER 5

PROJECT PLAN

5.1 PROJECT ESTIMATES

Sr. No	Phase	Tasks and Activities	Time Estimate	Effort Estimate
1.	Required Analysis	Review project documentation and user Requirements	1 week	1 person-week
		Conduct stakeholder meetings and gather requirements	1 week	
		Document project requirements	1 week	
2.	Design	Define system architecture and Components	1 week	2 person-weeks
		Create design specifications and Diagrams	2 weeks	
		Review and refine design with Stakeholders	1 week	
3.	Implement	Set up development environment and tools	1 week	6 person-weeks
		Implement core functionality (voice recognition, Bluetooth integration)	4 weeks	
		Develop user interface and interaction Components	3 weeks	
		Conduct code reviews and address feedback	1 week	
4.	Testing	Develop test cases and test plans	1 week	3 person-weeks
		Conduct unit testing	2 weeks	
		Perform integration testing with hardware components	2 weeks	
		Execute system testing and validation	1 week	
5.	Deployment	Prepare deployment environment and Configuration	1 week	1.5 person-weeks
		Deploy the software to production Environment	1 week	
		Conduct user acceptance testing and final.	1 week	

Table 5.1 Project Estimate

These estimates provide a detailed breakdown of the time and effort required for each phase, including specific tasks and activities involved. Adjustments may be necessary based on project-specific factors and additional details provided in the assignments.

5.1.1 Reconciled Estimates

5.1.1.1 Time Estimates

Creating a productive food waste tracking application involves several phases, from initial planning and design to development, testing, and deployment. Below is an estimated timeline for each phase, assuming a medium-sized project with moderate complexity. Adjustments may be necessary based on project specifics, team size, and resource availability.

5.1.2 Project Resources

1. **People**

a) UI/UX Designer:

- Designs the user interface and user experience.
- Creates wireframes, prototypes, and final designs.

b) Backend Developers:

- Develops the server-side logic, database interactions, and APIs.
- Ensures that the system is scalable, secure, and efficient.

c) Frontend Developers:

- Implements the user interface based on design specifications.
- Develops the web and/or mobile application frontend.

d) Machine Learning Engineer:

- Develops predictive models for food waste tracking.
- Analyzes data and integrates machine learning algorithms into the application.

2. **Hardware**

    a) **Development Workstations:** High-performance computers for developers, designers, and QA engineers.
    b) **Development and Testing Servers:** For staging and testing the application. **Production Servers:** For hosting the live application.

3. **Software**

    a) **Operating Systems:**

        - Linux (for servers)
        - Windows or macOS (for development workstations)

    b) **Backend Frameworks:**

        - Java

    c) **Machine Learning Libraries:**

        - TensorFlow, PyTorch, Scikit-learn

5.2 RISK MANAGEMENT W.R.T. NP HARD ANALYSIS

In the context of developing a productive food waste tracking application, risk management is crucial to identify, assess, and mitigate potential risks that could impact the project's success. This section outlines the key risks associated with the project and the strategies to manage them, particularly considering the complexities involved in dealing with NP-Hard problems, which are computational problems for which no efficient solution algorithm is known.

1. **Technical Complexity and Scalability**

    Risk: The application may involve complex algorithms and data structures, especially for predictive analytics and real-time data processing, which could be computationally intensive and hard to scale (potentially approaching NP-Hard complexities).

2. **Management Strategy:**

    - **Algorithm Optimization:** Focus on optimizing algorithms and using heuristics or approximation methods where exact solutions are infeasible.
    - **Scalable Architecture:** Design a scalable system architecture using microservices and cloud-based solutions to handle increased data loads efficiently.

3. **Performance Testing:** Regularly conduct performance testing to identify bottlenecks and optimize code.
4. **Algorithmic and Predictive Accuracy**

   Risk: Predictive models might not achieve the desired accuracy, leading to suboptimal waste management decisions.

5. **Management Strategy:**

   - **Data Quality:** Ensure high-quality, representative data for training models.
   - **Continuous Improvement:** Regularly update and retrain models based on new data and feedback.
   - **Hybrid Approaches:** Combine different predictive modeling approaches to improve accuracy.

5.2.1 Risk Identification

Risk ID	Risk Description	Category	Potential Impact	Mitigation Strategy
R1	Difficulty integrating IoT devices	Technical Risk	Data inconsistencies, increased maintenance costs	Use standardized protocols, robust middleware, and thorough testing
R2	Inability to scale the system	Technical Risk	System crashes, slow performance, user dissatisfaction	Design scalable architecture, use cloud solutions, conduct performance testing
R3	Inaccurate predictive models	Technical Risk	Ineffective waste management, increased costs	Continuously update models, use hybrid approaches, validate models regularly
R4	Failure to reduce food waste	Compliance/Environmental Risk	Not achieving sustainability goals, negative environmental	Not achieving sustainability goals, negative environmental

			impact	impact
R5	Non-compliance with regulations	Compliance Risk	Legal penalties, fines, reputational damage	Stay updated with regulations, conduct compliance audits, implement necessary controls

Table 5.2 Risk Identification

5.2.2 Risk Analysis

The risks for the Project can be analyzed within the constraints of time and quality

ID	Risk Description	Probability	Impact		
			Schedule	Quality	Overall
1	Scalability Issues with Data Handling	Low	High	High	High
2	Accuracy of Predictive Models	Medium	Medium	High	High
3	Integration Challenges with IoT Devices	Medium	Medium	High	High

Table 5.3: Risk Analysis

5.2.3 Overview of Risk Mitigation, Monitoring, Management

Following are the details for each risk.

Sr No.	Risk ID	1
1	Risk Description	The development environment may have compatibility issues with the required software tools and libraries, leading to delays and potential functionality problems.
2	Category	Development Environment.

3	**Source**	This risk is identified through the Software Requirement Specification document, highlighting dependencies on specific software versions and configurations.
4	**Probability**	Low (Due to proactive environment setup and testing)
5	**Impact**	High (Significant delays and additional costs if environment issues arise)
6	**Response**	Implement the mitigation strategies as outlined, ensuring that the development environment remains stable and reliable throughout the project lifecycle.
7	**Strategy**	Strategy
8	**Risk Status**	Occurred

Table 5.4 Risk Mitigation

Sr No	**Risk ID**	2
1	**Risk Description**	The requirements provided in the Software Design Specification (SDS) may be incomplete or ambiguous, leading to potential functional gaps or misunderstandings during the development phase.
2	**Category**	Requirements
3	**Source**	Software Design Specification documentation review.
4	**Probability**	Low (Assumed based on initial thorough review and clarification processes)
5	**Impact**	High (Misunderstood or incomplete requirements can lead to significant rework and delays)
6	**Response**	Execute the mitigation strategies effectively, ensuring that requirements are clear, complete, and validated throughout the project lifecycle.
7	**Strategy**	Better testing will resolve this issue.
8	**Risk Status**	Identified

Table 5.5 Risk Monitoring

5.3 PROJECT SCHEDULE

5.3.1 Project task set

Major Tasks in the Project stages are:

Task 1:

- **Description:** Initial phase where project scope, objectives, and requirements are defined and documented.

  Subtasks:

  - **Requirements Elicitation:** Gather detailed requirements through interviews, surveys, and document analysis.
  - **Requirements Documentation:** Create a Software Requirement Specification (SRS) document.
  - **Feasibility Study:** Analyze technical and financial feasibility.
  - **Project Plan Development:** Develop a detailed project plan, including timelines, milestones, and resource allocation.

Task 2:

- **Description:** Design the architecture, user interfaces, and overall system structure.

  Subtasks:

  - **System Architecture Design:** Define the high-level architecture of the application.
  - **Database Design:** Design the database schema and structure.
  - **UI/UX Design:** Create wireframes, mockups, and interactive prototypes.
  - **Design Reviews:** Conduct design reviews and obtain stakeholder approvals.
  - **Prototyping:** Develop a prototype to visualize the application and refine requirements.

Task 3:

- **Description:** Actual coding, integration, and testing of the application.

  Subtasks:

  - **Backend Development:** Develop the server-side logic, APIs, and database interactions.
  - **Frontend Development:** Implement the user interface and client-side logic.
  - **Mobile App Development:** Develop Android and iOS versions of the application.
  - **Integration:** Integrate all components (backend, frontend, mobile apps).
  - **Unit Testing:** Write and execute unit tests for individual components.

- **Integration Testing:** Conduct integration testing to ensure all components work together.
- **System Testing:** Perform system testing to verify the complete and integrated software product.
- **User Acceptance Testing (UAT):** Conduct UAT with end-users to validate the application against requirements.
- **Bug Fixing:** Identify and fix any defects or issues discovered during testing.

Task 4:

- **Description:** Deploy the application to the production environment and provide ongoing support.

  Subtasks:

- **Deployment Preparation:** Prepare the deployment environment and documentation.
- **Production Deployment:** Deploy the application to the live environment.
- **Post-Deployment Monitoring:** Monitor the application for any issues post-deployment.
- **User Training:** Provide training sessions and materials for end-users.
- **Maintenance and Support:** Offer continuous maintenance and support to address any issues and implement enhancements.
- **Feedback Collection:** Collect feedback from users to identify areas for improvement.

CHAPTER 6

SOFTWARE REQUIREMENT SPECIFICATION

6.1 INTRODUCTION

6.1.1 Purpose and Scope of Document

The purpose of the document for a food waste application using image processing would be to outline the objectives, functionality, and technical specifications of the application. Its scope would encompass the target audience, features, limitations, and potential impact. Additionally, it would detail the image processing techniques employed, data collection methods, user interface design, and implementation strategies.

Introduction:

Creating a food waste tracking application using image processing is a fantastic idea! It could work by allowing users to take pictures of their leftovers or expired food items, then using image recognition technology to identify the type and quantity of food being discarded. This data could be logged in the app, providing users with insights into their wastage habits and suggestions on reducing waste.

Functional Requirements:

Food waste applications that use imaging require capturing and analyzing food quality images, classifying products, estimating quantity, and controlling quality. It should provide users with a seamless experience, from an intuitive interface to easy navigation and feedback. It will show you the nearest feeding place.

Non-Functional Requirements:

The use of waste products using photography must be good in terms of performance, reliability, durability, security, usability, compatibility, management and management. It needs to process images quickly, ensure continuous product availability, meet customer needs, protect user data, provide comprehensive information, operate smoothly in use of products, be easy to manage, and comply with relevant regulations. The app meets these non-functional needs, providing a powerful, easy-to-use and legal solution to reduce food waste through new urine technology.

6.1.2 Overview of responsibilities of Developer

These waste product developers are responsible for many tasks throughout the development lifecycle. This includes developing and implementing image processing

algorithms, improving the backend infrastructure of the application, creating a more intuitive user interface, improving efficiency and performance, good, integrated security measures to protect user data, and regularly performing quality tests to identify and resolve issues. maintenance and support. Developers also need to stay aware of new technologies and best practices in image processing and software development to continually improve application performance and user experience.

6.2 USAGE SCENARIO

1. **Taking a image:** The user opens the app on his smartphone and uses the built-in camera to take a photo of the refrigerator shelf full of various foods.

2. **Image processing:** Image processing application uses image recognition algorithms to identify different foods, estimate their quantity, measure freshness, and check foods approaching their expiration date.

3. **Results:** The application displays the analysis results on the user's device, showing the names of the food products along with information such as quantity, freshness, and expiration date.

4. **Recommendations:** Based on reviews, the app provides users with personalized recommendations, such as recipes or meal ideas that are about to expire or use up needed items quickly.

5. **Notification:** If a product is detected to have expired, the app will send a notification to the user's device reminding them to drink or use the product immediately.

6. **Feedback:** Users can choose to provide feedback on the accuracy of the app's analytics, which helps improve its performance over time.

7. **Data Management:** The application securely stores statistical data, user interactions and feedback for future reference and analysis.

6.2.1 User profiles

- Creating user profiles for a productive food waste tracking application involves identifying the needs and behaviors of different user types.

- Eco-Conscious Individuals are typically urban adults aged 25-45 who actively seek to reduce their carbon footprint by tracking and minimizing food waste, needing features like waste reporting and recipes using leftovers.

- Busy Professionals, often aged 30-50, require efficient meal planning and inventory management due to their hectic schedules, appreciating reminders for food expiration

and integration with shopping lists.

- Family Planners are parents aged 35-55 who manage large households, aiming to balance meal variety and budget, thus needing bulk shopping optimization and family-friendly meal plans.
- Health Enthusiasts, young adults aged 20-35 focused on fitness, benefit from nutrition tracking and health-focused recipes to manage perishable health foods efficiently.
- Restaurant/Café Owners, aged 30-60, need robust inventory tracking and data analytics to reduce costs and optimize ingredient usage for their businesses. Lastly,
- Community Organizers involved in community kitchens or food banks require tools for coordinating food collection, distribution, and community education to maximize the use of donated food and promote sustainability practices locally. Each profile informs the design of user-centric features that cater to their specific requirements, enhancing the app's effectiveness in reducing food waste

6.2.2 Use-cases

Sr No.	Use Case	Description	Actors	Assumptions
1	Track Food Inventory	Users can track the food items they have at home, including quantities and expiration dates.	User (Eco-Conscious Individual, Busy Professional, Family Planner, Health Enthusiast)	The user has successfully logged into the application. The user has an initial inventory to input.
2	Monitor Food Waste	Users can track and monitor their food waste over time to identify patterns and areas for improvement.	User (Eco-Conscious Individual, Health Enthusiast, Family Planner)	The user has an existing inventory and meal plan. The user is logged into the application.
3	Donate Excess Food	Users can find local food banks or charities to donate excess food	User (Community Organizer, Restaurant/Café	The user has identified excess food items. The user

		items.	Owner, Eco-Conscious Individual)	is logged into the application. Excess food items are marked for donation. The user is connected with a local charity or food bank.
4	Educate and Engage Community	Users can access educational resources and engage with a community focused on reducing food waste.	User (Community Organizer, Eco-Conscious Individual, Family Planner)	The user is interested in food waste education.The user is logged into the application. The user accesses educational content. The user participates in community discussions and activities.

Table 6.1: Use Cases

6.2.3 Use Case View

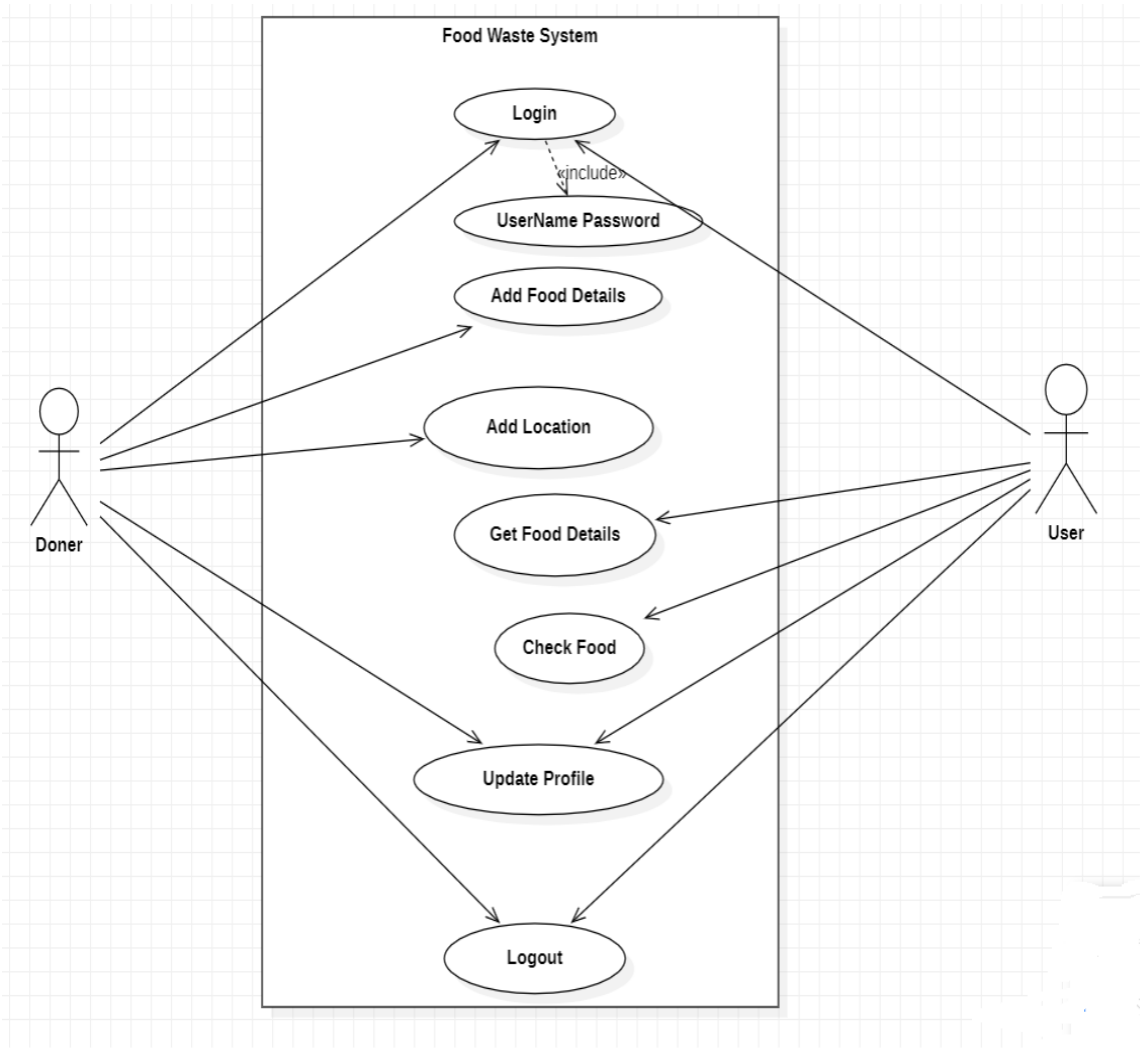


Figure 6.1: Use case diagram

6.3 DATA MODEL AND DESCRIPTION

6.3.1 Data Description

1. User: Represents the individuals who use the application.
2. Food Item: Represents individual food items that can be tracked in the inventory.
3. Waste Entry: Represents a record of food waste.
4. Donation: Represents a donation made by the user.
5. Charity: Represents a charity or food bank.
6. Community Post: Represents a post made by a user in the community section.

6.3.2 Data objects and Relationships

1) **Charity**
   - **Charity ID** (Primary Key): Unique identifier for the charity.
   - **Name:** Name of the charity.
   - **Address:** Address of the charity.
   - **Contact Info:** Contact information for the charity.
2) **Community Post**
   - **Post ID** (Primary Key): Unique identifier for the post.
   - **User ID** (Foreign Key): Reference to the user who made the post.
   - **Content:** Content of the post.
3) **Donation**
   - **Donation ID** (Primary Key): Unique identifier for the donation.
   - **User ID** (Foreign Key): Reference to the user who made the donation.
   - **Charity ID** (Foreign Key): Reference to the charity receiving the donation.
   - **Items Donated:** List of items donated.

6.4 FUNCTIONAL MODEL AND DESCRIPTION

This section describes each major software function of the food waste tracking application, along with the data flow and class hierarchy.

Major Software Functions

1. User Management

2. Food Inventory Management
3. Meal Planning
4. Shopping List Generation
5. Food Waste Tracking
6. Food Donation
7. Community Engagement

Class Descriptions

1) **User**

i. **Attributes:**

- User ID: Unique identifier for the user.
- username: User's chosen login name.
- password: User's hashed password.
- email: User's email address.
- Profile Type: Type of user (e.g., Eco-Conscious Individual).
- preferences: User-specific settings and preferences.

ii. **Methods:**

- register(): Register a new user.

2) **Food Item**

i. **Attributes:**

- Food Item ID: Unique identifier for the food item.
- name: Name of the food item.
- category: Category of the food item (e.g., vegetable, dairy).
- Shelf Life: Typical shelf life of the food item.
- Storage Instr: Best practices for storing the food item.

6.4.1 Data Flow Diagram

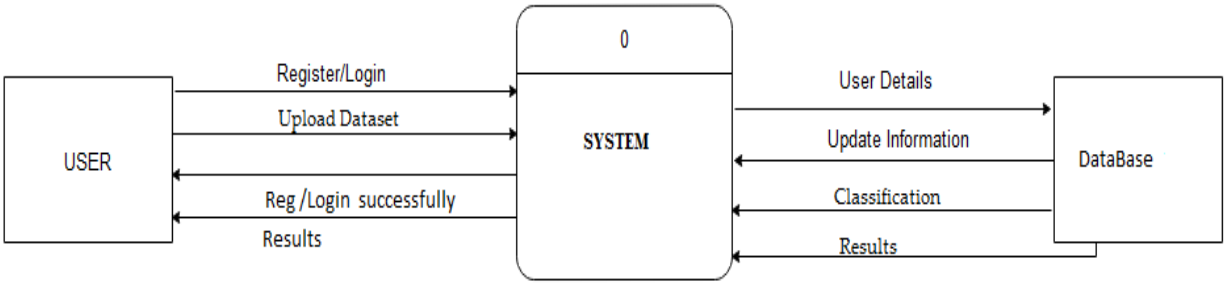


Figure 6.2: DFD

6.4.2 Activity Diagram:

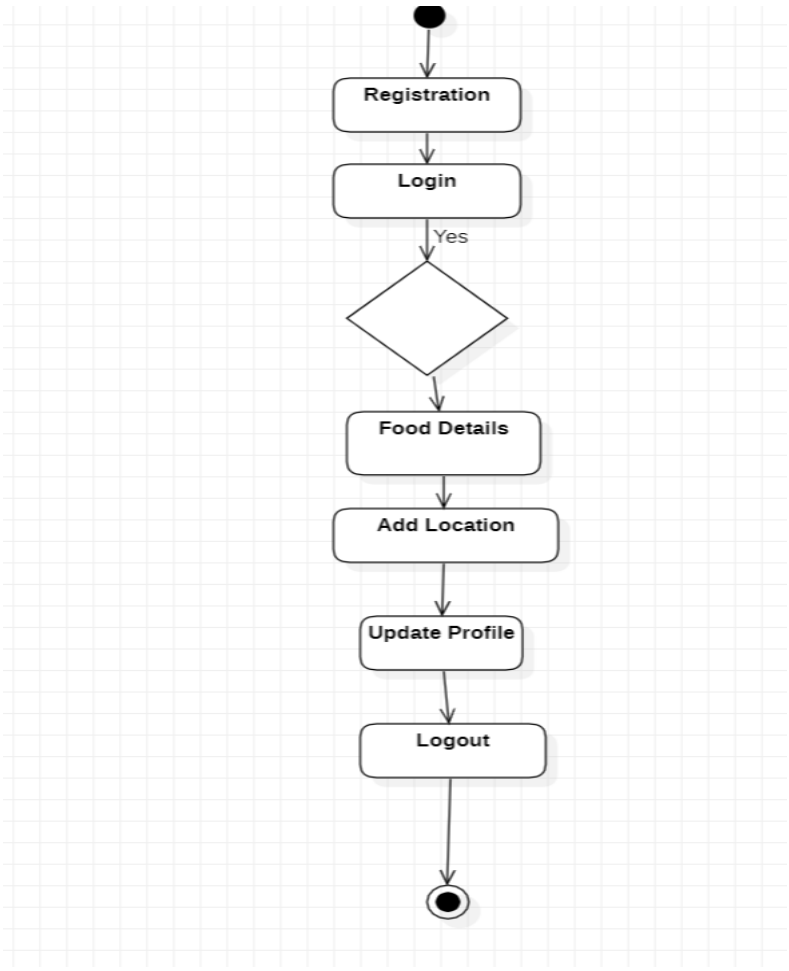
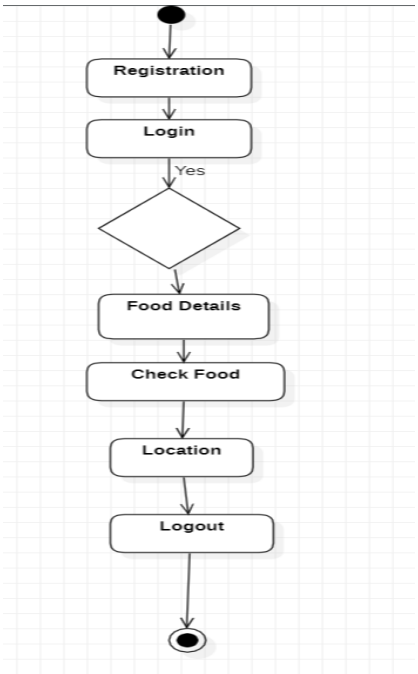


Figure 6.3: Activity Diagram

Receiver Activity Diagram



6.4 Receiver Activity Diagram

6.4.3 Non Functional Requirements:

Interface Requirements

The food waste tracking application must have a user-friendly interface accessible via web and mobile platforms. The user interface should be intuitive, with clear navigation and minimal learning curve, ensuring users can easily manage their food inventory, meal plans, and waste tracking. The interface should support accessibility features such as screen readers and high contrast modes to cater to users with disabilities.

Performance Requirements

The application must handle concurrent users efficiently, with a target response time of less than 2 seconds for common operations such as adding food items, generating shopping lists, and recording waste entries. The backend should be capable of handling peak loads, ensuring that the application remains responsive even during high traffic periods, such as before major holidays when users are likely to interact with the app more frequently.

Software Quality Attributes

1. **Availability (Reliability):** The application should have an uptime of at least 99.9%, ensuring that it is reliable and accessible to users whenever needed. Redundant systems and regular backups should be implemented to minimize downtime and data loss.
2. **Modifiability:**
    - **Portability:** The application should be deployable across various operating systems and platforms with minimal changes.
    - **Reusability:** Components of the application, such as the inventory management and waste tracking modules, should be designed for reuse in other projects or extensions of the current application.
    - **Scalability:** The system architecture should support scalability, allowing the application to handle increased user loads and data volumes without significant performance degradation.
3. **Performance:** The application should efficiently handle database queries and data processing tasks, ensuring that users experience minimal delays. Optimized algorithms and indexing should be used to enhance data retrieval times.
4. **Security:** User data must be protected through robust encryption methods for data at rest and in transit. The application should implement authentication and authorization mechanisms to prevent unauthorized access. Regular security audits and updates should be conducted to address vulnerabilities.
5. **Testability:** The system should be designed to support automated testing, with clear modular boundaries and comprehensive test coverage for all functionalities. This ensures that new features can be added and bugs can be fixed without affecting existing functionality.
6. **Usability:**
    - **Self Adaptability:** The application should provide personalized recommendations based on user behavior, such as suggesting recipes to use up items nearing their expiration date.
    - **User Adaptability:** The interface should be customizable, allowing users to adjust settings according to their preferences, such as choosing measurement units or setting notification preferences. The design should follow best practices in user experience to ensure that users can efficiently accomplish their tasks with minimal effort.

6.4.4 State Diagram:

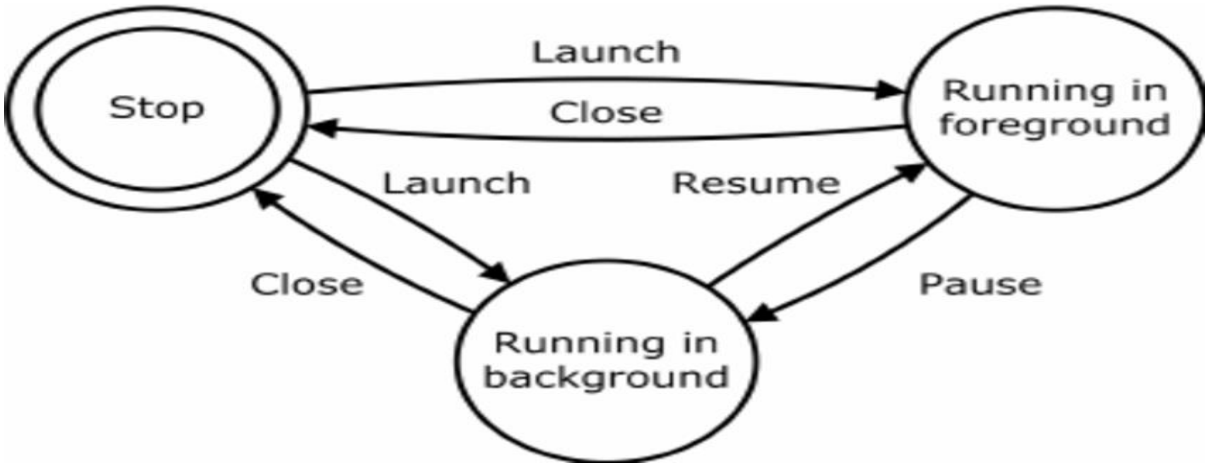


Figure 6.5: State Diagram

6.4.5 Design Constraints

1) **Platform Compatibility**

   Web and Mobile Support: The application must be compatible with major web browsers (Chrome, Firefox, Safari, Edge) and mobile operating systems (iOS and Android). This necessitates the use of responsive design principles and possibly different codebases or frameworks (e.g., React Native for mobile and React.js for web).

2) **Data Privacy and Compliance**

   GDPR Compliance: The application must comply with the General Data Protection Regulation (GDPR) for users in the European Union, requiring careful handling of personal data, providing data access and deletion options, and ensuring data is processed lawfully.

   Other Regulations: Compliance with other relevant data protection laws (e.g., CCPA for California residents) must be ensured.

3) **Performance Limitations**

   Server Load: The backend infrastructure must handle a high number of concurrent users without performance degradation. This may require load balancing, efficient database indexing, and optimization techniques.

   Latency: Real-time features, such as inventory updates and notifications, should have minimal latency to provide a seamless user experience.

4) **Security Requirements**

Encryption: All sensitive data must be encrypted both in transit (using TLS/SSL) and at rest.

Authentication and Authorization: Strong authentication mechanisms, such as OAuth, and role-based access control must be implemented to prevent unauthorized access.

Regular Security Audits: Periodic security audits and vulnerability assessments should be conducted to identify and mitigate potential security threats.

5) **User Experience**

Accessibility: The application must be accessible to users with disabilities, following WCAG (Web Content Accessibility Guidelines) to ensure inclusive design.

Localization: The system should support multiple languages and regional settings to cater to a diverse user base.

6) **Resource Limitations**

Development Resources: The project must consider the available development resources, including time, budget, and team expertise. This may limit the scope and complexity of features that can be implemented initially.

Maintenance and Updates: The design should facilitate easy maintenance and updates to the application, ensuring that new features and security patches can be deployed with minimal disruption.

7) **Environmental Constraints**

Network Dependence: The application must function effectively in varying network conditions. Offline capabilities or local caching may be necessary to ensure usability when internet connectivity is poor.

6.4.6 Software Interface Description

Web Interface:

- **Technology:** XML.
- **Features:** Responsive design for compatibility with various screen sizes and devices, including desktops, tablets, and smartphones. Accessible navigation, form inputs for inventory management, meal planning, waste tracking, and user settings.
- **Requirements:** Compliance with Web Content Accessibility Guidelines (WCAG) to ensure accessibility for users with disabilities.

Admin Dashboard:

- **Technology:** Web-based admin panel.
- **Features:** Manage users, view analytics, update system settings, monitor performance and security.
- **Requirements:** Role-based access control, real-time data visualization, secure admin access.

Customer Support Interface:

- **Technology:** Integrated chat support, email support system.
- **Features:** Real-time assistance, ticket management, FAQ and knowledge base.
- **Requirements:** Responsive and easy-to-use interface for customer service representatives, integration with user data for context-aware support.

CHAPTER 7

DETAILED DESIGN DOCUMENT USING APPENDIX A AND B

7.1 INTRODUCTION

Food waste is a pressing issue in modern society, posing significant challenges to both the environment and the economy. To combat this problem, we have developed a cutting-edge Android mobile application that harnesses the power of mobile technology to tackle food waste. This innovative app empowers individuals and restaurants to contribute surplus food and leftovers to those in need. The efficient management of food resources is crucial for our planet's environmental well-being and economic stability. Food waste not only squanders valuable resources but also contributes to environmental degradation through greenhouse gas emissions. Moreover, it exacerbates food insecurity and hunger for vulnerable populations. Our study addresses these critical issues by providing a practical solution that bridges the gap between food surplus and food scarcity. The results of our research and the functionality of our mobile application are highly relevant in today's world. As food waste continues to be a global concern, our app offers a scalable and accessible means to redistribute excess food to those who need it most. By connecting donors, agents, and beneficiaries, we aim to foster a sense of community and shared responsibility in the fight against food waste and hunger. The primary purpose of our study is to demonstrate the effectiveness of our Android application in reducing food waste and alleviating food poverty. Through the seamless process of food donation, collection, and redistribution facilitated by the app, we intend to showcase its potential to make a substantial impact on both the environmental and social fronts. Furthermore, we emphasize the importance of user confidentiality to ensure the security and privacy of all participants in this valuable initiative.

7.2 ARCHITECTURAL DESIGN

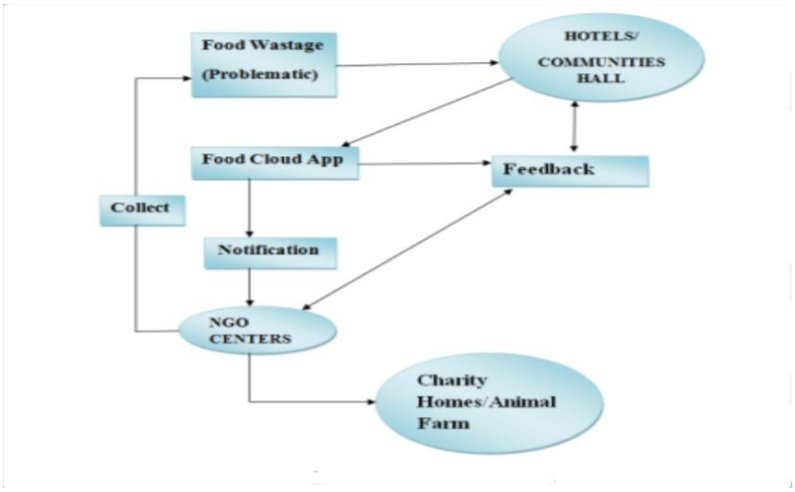


Figure 7.1: Architecture diagram

1. **User and Restaurant Logins:** The app offers two distinct types of logins: one for regular users and another for restaurants. This differentiation ensures that the app caters to the needs of both individuals looking for surplus food and restaurants wishing to contribute excess food.
2. **User Registration and Authentication:** Users can register for an account by providing their email address and choosing a secure password. Firebase's email and password authentication system is utilized for user registration and login, ensuring a high level of security.
3. **User Profile Creation:** Upon successful registration, users are prompted to create their profiles. They can input personal information such as age, name, and gender. This profile data is associated with a unique user ID generated by Firebase. Storing this information in the real-time database enables personalized experiences for each user.
4. **Logout Functionality:** Users can securely log out of their accounts when they've finished using the app. This feature is essential for protecting user data and privacy.
5. **Viewing Available Food Listings:** Users can browse through a list of available surplus food items. These listings typically include details such as food images, titles, and descriptions, allowing users to make informed choices
6. **Restaurant Registration:** Restaurants or food establishments interested in participating in food donation can also register for accounts. The registration process for restaurants likely involves providing their business details and agreeing to terms and conditions.
7. **Unique User IDs:** Each user is assigned a unique user ID by Firebase. This ID serves as a key to access and update the user's profile information stored in the real-time database. This ensures data separation and privacy for individual users.

The app's design accommodates both user and restaurant needs, offering an intuitive user experience while addressing the critical issue of food waste reduction. User profiles, authentication, and real-time database integration contribute to a personalized and secure platform, making it easier for users to access surplus food and for restaurants to contribute to the cause.

7.3 DATA DESIGN (USING APPENDICES A AND B)

Internal Data Structures

1) **User Profile Data Structure:**

- User ID (Unique identifier)
- Username
- Email
- Password (Hashed for security)

2) Food Item Data Structure:

- Food Item ID (Unique identifier)
- Name

3) Database Design (Tables)

- **Users Table:**
- User ID (Primary Key)
- Username
- Email
- Password

7.3.1 Internal software data structure

In the food waste tracking app, several data structures are passed among its components to manage various functionalities efficiently. The user profile data structure includes fields like user ID, username, email, password, age, gender, and user preferences such as dietary restrictions, favorite foods, and allergies. Food item data structure contains information like food item ID, name, category, shelf life, and storage instructions. Meal plan data structure consists of fields like meal plan ID, user ID, start date, end date, and meals with details such as date, meal type, and corresponding food items. Waste entry data structure includes waste entry ID, user ID, food item ID, quantity wasted, waste date, and reason for waste. Inventory data structure includes inventory ID, user ID, food item ID, and quantity of each food item in the user's inventory. These data structures enable seamless communication and data management across the app's components, ensuring efficient tracking of food waste and facilitating user interaction with the platform.

7.3.2 Global data structure

The global data structure in the food waste tracking application encompasses information that is accessible across major portions of the architecture, ensuring consistency and coherence in data handling. The inventory data structure contains a list of inventory items associated with a specific user identified by their unique user ID. Each inventory item includes the food item ID and the quantity of that item currently available in the user's inventory. This structure allows for easy management and retrieval of inventory information across various components of the application, such as meal planning, waste tracking, and user profile customization. It ensures that users have a comprehensive view of their available food items and can make informed decisions about meal planning and waste reduction strategies. Additionally, the global nature of this data structure facilitates seamless integration with other modules and ensures consistent data representation throughout the application.

7.3.3 Temporary data structure

In the food waste tracking application, temporary data structures are utilized for interim use, facilitating specific tasks or operations without permanently storing the data. These temporary data structures are typically ephemeral and serve a transient purpose. Each item includes the food item ID and the quantity selected by the user. This temporary data structure is used to facilitate the shopping experience, allowing users to add and remove items as they browse available food listings without permanently modifying their inventory or making a purchase. Once the user completes their shopping session or checks out, the contents of the shopping cart are typically cleared, and the temporary data structure is no longer needed. Temporary data structures like the shopping cart help streamline user interactions and provide a seamless experience within the food waste tracking application.

7.3.4 Database description

The TensorFlow database utilized in the food waste tracking application supports machine learning models and algorithms aimed at optimizing waste reduction strategies, improving food inventory management, and providing personalized recommendations. While TensorFlow itself doesn't inherently manage databases, it interacts with databases to access and manipulate data for training models and making predictions.

1. Database Type:

The database supporting TensorFlow functionalities within the food waste tracking application is likely a relational database management system (RDBMS) or a NoSQL database. The choice depends on factors such as data structure, scalability requirements, and specific use cases for machine learning models.

2. Schema:

The schema of the TensorFlow database includes tables or collections specifically designed to store data relevant to machine learning tasks. This may include:

- **Training Data:** Tables or collections storing historical data on food items, consumption patterns, waste trends, and user behaviors.
- **Model Parameters:** Tables or collections storing parameters and configurations for machine learning models, including neural network architectures, hyperparameters, and optimization settings.
- **Model Outputs:** Tables or collections containing the results of model predictions, such as recommended recipes, personalized meal plans, or waste reduction strategies.

3. Integration with TensorFlow*:*

The TensorFlow database is integrated with TensorFlow through appropriate data access libraries or frameworks. This integration enables TensorFlow to fetch training data, store model parameters, save model outputs, and evaluate model performance directly from the database.

4. Data Management:

The TensorFlow database supports efficient data management operations, including data retrieval, insertion, update, and deletion. It may utilize indexing, partitioning, and caching techniques to optimize query performance and ensure scalability for large datasets.

5. Security and Privacy:

The TensorFlow database adheres to security and privacy best practices to protect sensitive data, including user information and training data. This may involve encryption, access control mechanisms, and compliance with data protection regulations.

7.4 COMPOENT DESIGN

1. **User Management Component**

- **Description:** Responsible for handling user registration, authentication, and profile management functionalities.
- **Algorithms:**
    - User Registration Algorithm: Validates user inputs, checks for existing accounts, and creates a new user profile in the database.
    - Authentication Algorithm: Verifies user credentials during login, grants access upon successful authentication.
    - Profile Update Algorithm: Allows users to update their profile information, such as age, gender, and preferences.

2. **Waste Tracking Component**

- **Description:** Tracks instances of food waste by users, recording details such as wasted items, quantities, and reasons for waste.
- **Algorithms:**
    - Record Waste Entry Algorithm: Captures user inputs on wasted food items, quantities, and reasons, storing this information in the database.
    - Analyze Waste Patterns Algorithm: Analyzes historical waste data to identify patterns and trends, providing insights for waste reduction strategies.
    - Generate Waste Reports Algorithm: Generates reports summarizing waste statistics, including total

7.4.1 Class Diagram

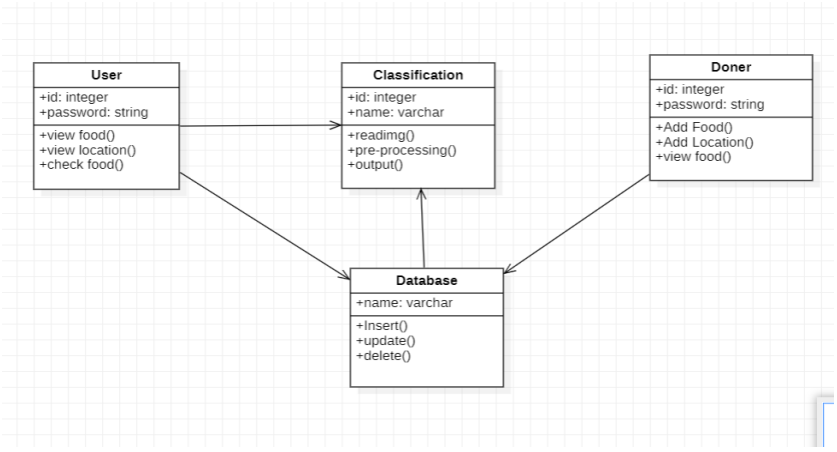


Figure 7.2: Class Diagram

CHAPTER 8

PROJECT IMPLEMENTATION

8.1 INTRODUCTION

Creating a food waste tracking application using image processing is a fantastic idea! It could work by allowing users to take pictures of their leftovers or expired food items, then using image recognition technology to identify the type and quantity of food being discarded. This data could be logged in the app, providing users with insights into their wastage habits and suggestions on reducing waste.

8.2 TOOLS AND TECHNOLOGIES USED

1. **Programming Languages:** Depending on your preference and expertise, you could use languages like Python, JavaScript, or Swift for app development.
2. **Mobile App Development Frameworks:** For building the mobile app, you might use frameworks like React Native (for cross-platform development), Flutter, or native development tools like Android Studio (Java/Kotlin) or Xcode (Swift).
3. **Image Processing Libraries:** Utilize libraries like OpenCV, TensorFlow, or PyTorch for image recognition and processing tasks. These libraries offer pre-trained models for object detection and classification.
4. **Backend Development:** You may need a backend server to handle data storage, user authentication, and possibly image processing tasks. Technologies like Node.js, Django, Flask, or Firebase could be used for backend development.
5. **Database:** Choose a database system to store user data and food waste information.
6. **Cloud Services:** Consider using cloud services like AWS, Google Cloud Platform, or Microsoft Azure for hosting your backend server, storing images, or running machine learning models.
7. **APIs:** You might integrate third-party APIs for additional functionalities, such as weather APIs for suggesting recipes based on local produce availability or nutrition APIs for analyzing food items.
8. **Version Control: Use** version control systems like Git to manage your codebase and collaborate with other developers if needed.
9. **UI/UX Design Tools:** Design the user interface of your app using tools like Adobe XD, Sketch, Figma, or InVision.
10. **Testing and Deployment***: Employ testing frameworks and continuous integration tools to ensure the stability and reliability of your application. For deployment, platforms like Firebase, Heroku, or Docker containers can be used.

8.3 METHODOLOGIES/ALGORITHM DETAILS

1. **User Login:** Registration: Users can register for an account within the app. This registration process involves creating a unique account using Firebase email and password authentication, ensuring secure access to the app.
2. **Login and Logout:** Registered users can log in and log out of their accounts. This functionality provides a personalized experience while ensuring account security.
3. **Restaurant Login:** Restaurants interested in participating in the food waste reduction initiative must also register using the app's login page. This ensures that only authorized restaurants can list their surplus food.
4. **User Profile Information:** The user profile data, including age, name, and gender, is stored in a real time database. This database serves as a repository for user-specific information.

8.4 VERIFICATION AND VALIDATION FOR ACCEPTANCE

Verification and validation are essential processes in software development to ensure that the product meets the specified requirements and functions as intended. Below are the acceptance criteria for verification and validation of the food waste tracking application:

Verification Acceptance Criteria:

1. **User Authentication:**
    - Verify that users can register with valid email addresses and passwords.
    - Confirm that users can log in successfully after registration.
2. **Inventory Management:**
    - Validate that users can add, remove, and update items in their inventory.
    - Ensure that inventory changes are reflected accurately in the database.
3. **Waste Tracking:**
    - Verify that users can record instances of food waste, including details such as wasted items, quantities, and reasons.
    - Confirm that waste entries are stored correctly in the database and can be retrieved for analysis.
4. **Recommendation Engine:**

- Validate that users receive personalized recipe recommendations based on their inventory and preferences.
- Confirm that meal plans are optimized to minimize waste and align with user preferences.

Validation Acceptance Criteria:

1. **User Experience:**
   - Conduct user testing to ensure that the application interface is intuitive and easy to navigate.
   - Gather feedback from users to identify any usability issues or areas for improvement.
2. **Functionality:**
   - Validate that all core functionalities, such as user registration, inventory management, and waste tracking, work as expected.
   - Ensure that users can access and utilize recommended features effectively.
3. **Accuracy of Recommendations:**
   - Verify that recipe recommendations and meal plans align with user preferences and available inventory.
   - Validate that recommendations contribute to reducing food waste and improving user satisfaction.
4. **Performance:**
   - Conduct performance testing to assess the application's responsiveness and scalability.
   - Ensure that the application can handle concurrent users and large datasets without significant degradation in performance.
5. **Data Integrity:**
   - Validate that data stored in the database remains consistent and accurate over time.
   - Conduct data integrity checks to identify and resolve any discrepancies or errors.
6. **Compliance:**
   - Ensure that the application complies with relevant regulations and standards, such as data privacy laws (e.g., GDPR) and accessibility guidelines (e.g., WCAG).
7. **Security:**

- Validate that user data is secure and protected from unauthorized access or breaches.
- Conduct security audits to identify and address potential vulnerabilities or weaknesses in the application.

CHAPTER 9

SOFTWARE TESTING

9.1 TYPE OF TESTING USED

Black Box Testing:

Black box testing methods focus on the functional requirements in the software. That is, black box testing enables us to derive sets of input conditions that will fully exercise. All functional requirements of the program Black box testing attempts to find errors in the following categories:

- Incorrect or missing function
- Interface errors
- Errors in data structure or external job access
- Performance errors
- Initialization and termination errors.

In the proposed application with the help of this technique, we do not use the code to determine test suite; rather, knowing the problem that we are trying to solve, we come up with four types of test data:

- Easy-to-compute data,
- Typical data,
- Boundary / extreme data,
- Bogus data.

But in our application we does not provide any external data, the role of user is only to give number of nodes for formation of clusters and for the formation of sink node.

White Box Testing:

White box testing is a set case design method that uses the control structure of the procedural design to derive test cases. Using white box testing methods, we can derive test cases that: Guarantee that all independent paths within a module have been exercised at least once. Exercise all logical decisions on their true and false sides. Execute all loops at their boundaries and within their operational bounds. Exercise internal data structures to ensure their validity. In the proposed application the white box testing is done by the developer implemented the code, the implements code is studied by the coder, determines all legal (valid and invalid) and illegal inputs and verifies the outputs against the expected outcomes, which is also determined by studying the implementation code.

Testing Types

- **Unit Testing:** Unit testing enables a programmer to detect error in coding. A unit test focuses verification of the smallest unit of software design. This testing was carried out during the coding itself. In this testing step, each module going to be work satisfactorily as the expected output from the module.
- **Project Aspect:** The front end design consists of various forms. They were tested for data acceptance. Similarly, the back-end also tested for successful acceptance and retrieval of data. The unit testing is done on the developed code. Mainly the unit testing is done on modules
- **Integration Testing:** Through each program work individually, they should work after linking together. This is referred to as interfacing. Data may be lost across the interface; one module can have adverse effect on the other subroutines after linking may not do the desired function expected by the main routine. Integration testing is the systematic technique for constructing the program structure while at the same time conducting test to uncover errors associated with the interface. Using integrated test plan prepared in the design phase of the system development as a guide, the integration test was carried out. All the errors found in the system were corrected for the next testing step.
- **System Testing:** After performing the integration testing, the next step is output testing of the proposed system. No system could be useful if it does not produce the required output in a specified format. The outputs generated are displayed by the user. Here the output format is considered in to two ways. One in on screen and other in printed format.
- **System testing for the current system:** In this level of testing, testing the system after integrating developed sub module of the project and testing whether system is giving correct output or not. Developed sub module was integrated and the flow of information was checked. It was also checked that whether the flow of data is as per the requirements or not. It was also checked that whether any particular module is non-functioning or not i.e. once the integration is over each and every module is functioning in its entirety or not. In this level of testing, testing can be done for following points: -
  - Whether developed forms are properly working or not?
  - Whether developed forms are properly linked or not?
  - Whether images are properly displayed or not?
  - Whether data retrieval is proper or not?

Test Plan:

Test Plan Identifier: cost-sensitive. It is use to identify test plan uniquely.

- **Purpose of the Test Plan Document:** The main purpose of this document is to fit a particular project's needs. It documents and tracks the necessary information required to effectively define the approach to be used in the testing of the projects product. The Test Plan document is created during the Planning Phase of the project. Its intended audience is the project manager, project team, and testing team.
- **Objective of Test Panning:** To find as many defects as possible and get them fix.
- **Items to be Tested OR Not to be tested:** Describe the items/features/functions to be tested that are within the scope of this test plan. Include a description of how they will be tested, when, by whom, and to what quality standards. Also include a description of those items agreed not to be tested.
- **Items to be tested:** Overall functionality of the application User Interface of the application.
- **Not to be tested:** Performance of the application.

Test Approach:

Describe the overall testing approach to be used to test the projects product. Provide an outline of any planned tests. There are many approaches like:

Box Testing White Box Testing:

Here we used Black Box Testing approach. In Black Box Testing we just give input to the system and check its output without checking how system processes it.

- **Test Pass OR Test Fail Criteria:** When actual and expected results are same then test will be passed. When actual and expected results are different then test will be failed.
- **Test Entry OR Exit Criteria:** Describe the entry and exit criteria used to start testing and determine when to stop testing.
- **Entry criteria:** As soon as we have requirement we can start testing.
- **Exit criteria:** When bug rate fall below certain level we can stop testing.
- **Test Suspension OR Resumption Criteria**: Describe the suspension criteria that may be

used to suspend all or portions of testing. Also describe the resumption criteria that may be used to resume testing.

- **Suspension criteria:** If there is large change in application like change in requirements we can suspend work for some time.
- **Resumption criteria:** After resolving the respective problem we can resume work item
- **Testing Type:** It describes which testing types we are going to follow in our testing lifecycle.
    - Here we are using:
    - Black Box Testing
    - Functional Testing
    - UI Testing
    - Integration Testing

9.2 TEST CASES AND TEST RESULTS

Test cases are built around specifications and requirements, i.e., what the application is supposed to do. Test cases are generally derived from external descriptions of the software, including specifications, requirements and design parameters. Although the tests used are primarily functional in nature, nonfunctional tests may also be used. The test designer selects both valid and invalid inputs and determines the correct output without any knowledge of the test object's internal structure.

Test Design Techniques Typical black-box test design techniques include:

- Decision table testing
- All-pairs testing
- State transition Analysis
- Equivalence partitioning
- Boundary value analysis
- Cause–effect graph
- Error guessing

Advantages:

Efficient when used on large systems. Since the tester and developer are independent of each other, testing is balanced and unprejudiced.

1. Tester can be non-technical.
2. There is no need for the tester to have detailed functional knowledge of system.
3. Tests will be done from an end user's point of view, because the end user should accept the system. (This testing technique is sometimes also called Acceptance testing.)
4. Testing helps to identify vagueness and contradictions in functional specifications.
5. Test cases can be designed as soon as the functional specifications are complete.

Disadvantages:

1. Test cases are challenging to design without having clear functional specifications.
2. It is difficult to identify tricky inputs if the test cases are not developed based on specifications.
3. It is difficult to identify all possible inputs in limited testing time. As a result, writing test cases may be slow and difficult.
4. There are chances of having unidentified paths during the testing process.
5. There is a high probability of repeating tests already performed by the programmer.

Test case For Registration Page:

Test Case ID	Test Case Procedure	Input Data	Expected Output	Actual Output	Test Status
1	Checking the functionality of Submit Button	Enter valid User name Enter valid email-id Enter valid Password Enter valid date Enter valid contact no. Click on Register	Accept & Login page should be displayed	Accept & Login page displayed	Pass

		Button			
2	Checking the functionality of Submit Button	Enter invalid User name Enter valid email-id Enter valid Password Enter valid date Enter valid contact no. Click on Register Button	Error should come & Register page should be displayed	Error comes & Register page displayed	Pass
3	Checking the functionality of Submit Button	Enter valid User name Enter in valid email-id Enter valid Password Enter valid date Enter valid contact no. Click on Register Button	Error should come & Register page should be displayed	Error comes & Register page displayed	Pass
4	Checking the functionality of Submit Button	Enter valid User name Enter valid email-id Enter invalid Pass word Enter valid date Enter valid contact	Error should come & Register page should be displayed	Error comes & Register page displayed	Pass

		no. Click on Register Button			
5	Checking the functionality of Submit Button	Enter valid User name Enter valid email-id Enter valid Password Enter in valid date Enter valid contact no. Click on Register Button	Error should come & Register page should be displayed	Error comes & Register page displayed	Pass
6	Checking the functionality of Submit Button	Enter valid User name Enter valid email-id Enter valid Password Enter valid date Enter in valid contact no. Click on Register But ton	Error should come & Register page should be displayed	Error comes & Register page displayed	Pass

Table 9.1: Test cases For Registration Page

Test case For Login Page:

Test Case ID	**Test Case Procedure**	**Input Data**	**Expected Output**	**Actual Output**	**Test Status**

1	Checking the functionality of Login Button	Enter valid User name Enter valid Password Click on Login Button	Accept & User's home page should be display	Accept & User's home page displayed	Pass
2	Checking the functionality of Login Button	Enter valid User name Enter Invalid Pass word Click on Login Button	Error should come & Login page should be displayed	Error should come & Login page displayed	Pass
3	Checking the functionality of Login Button	Enter Invalid User name Enter valid Password Click on Login Button	Error should come & Login page should be displayed	Error should come & Login page displayed	Pass
4	Checking the functionality of Login Button	.Enter Invalid User name .Enter Invalid Pass word Click on Login Button	Error should come & Login page should be displayed	Error should come & Login page displayed	Pass

Table 9.2: Test Cases for Login Page

CHAPTER 10

RESULTS

10.1 SCREEN SHOTS

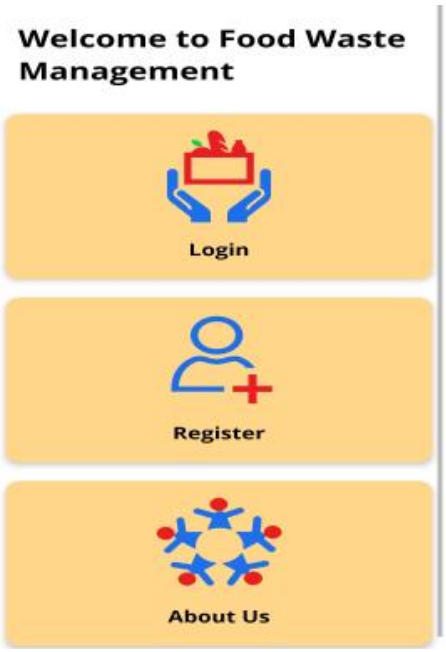


Fig 10.1: Welcome Page

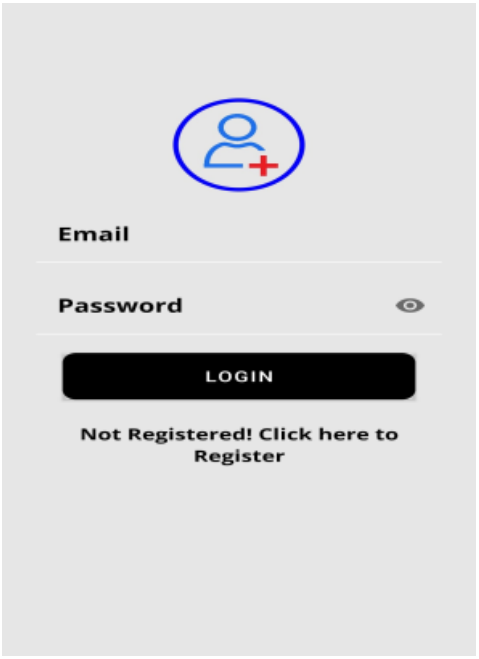


Fig 10.2: Log In Page

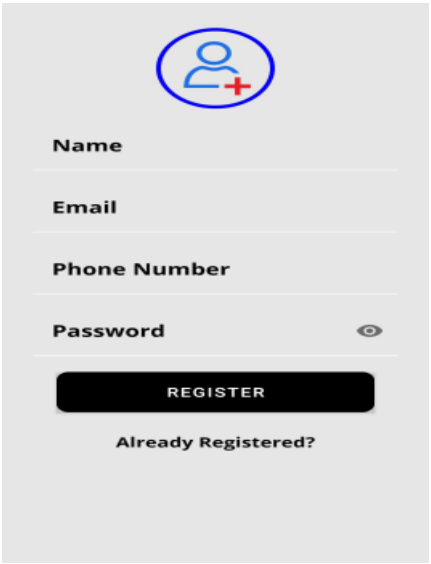


Fig 10.3: Registration Page

Fig 10.4: Home page

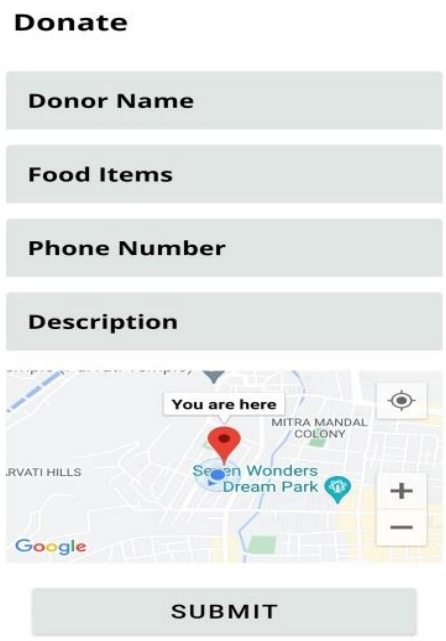


Fig 10.5: Donar's Page and location

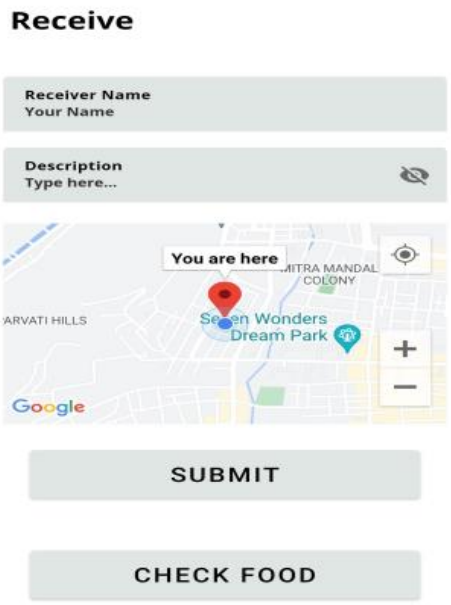


Fig 10.6: Receiver's Page and Location

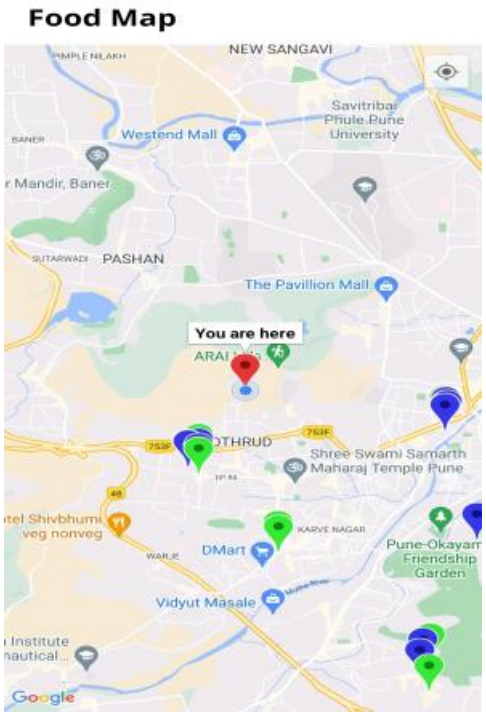


Fig 10.7: Donar's & Receiver's location

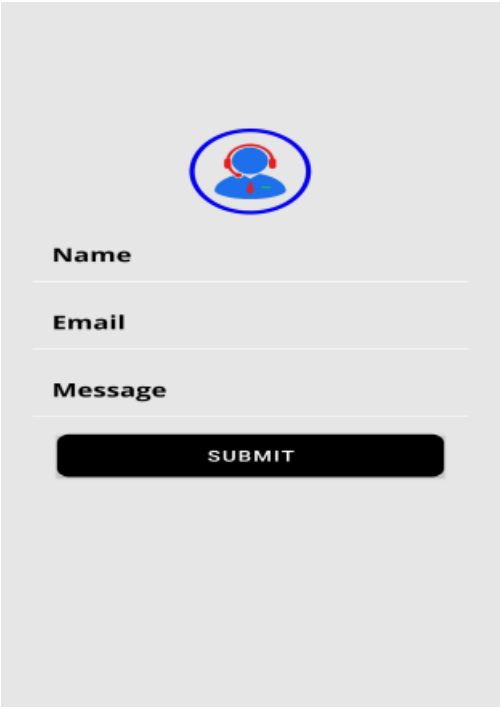


Fig 10.8: Contact Us

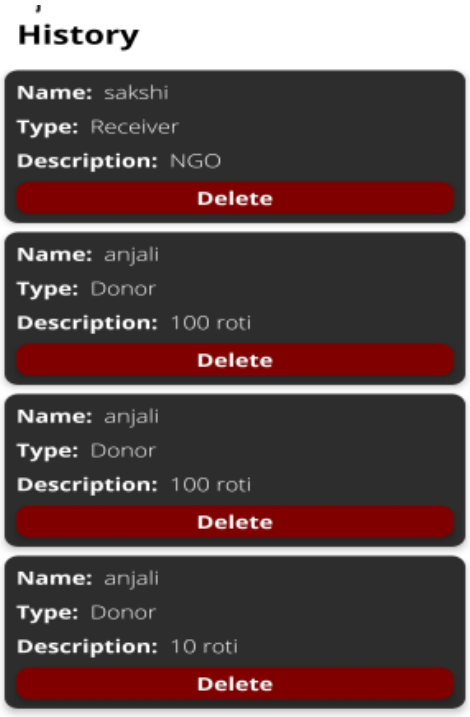


Fig: 10.9 History

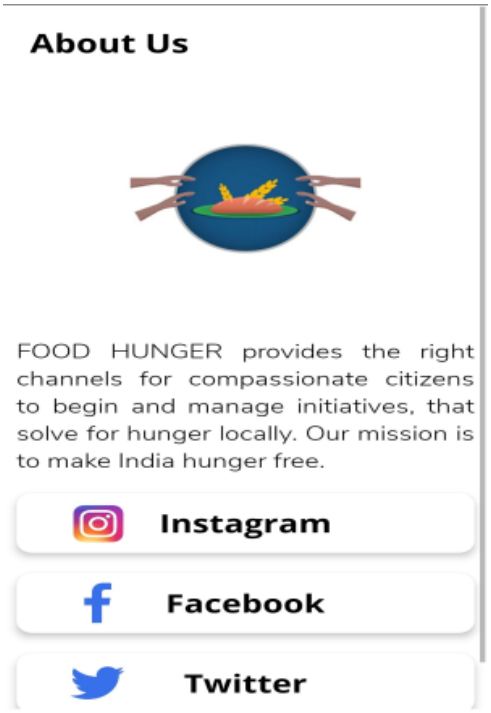


Fig 10.10 About us

10.2 OUTPUTS

Fig 10.11: Image Prediction

Fig 10.12: Bad rice prediction

Image Prediction Output:

Good Roti

Take Picture

Gallery Image

Fig: 10.13: Good roti prediction

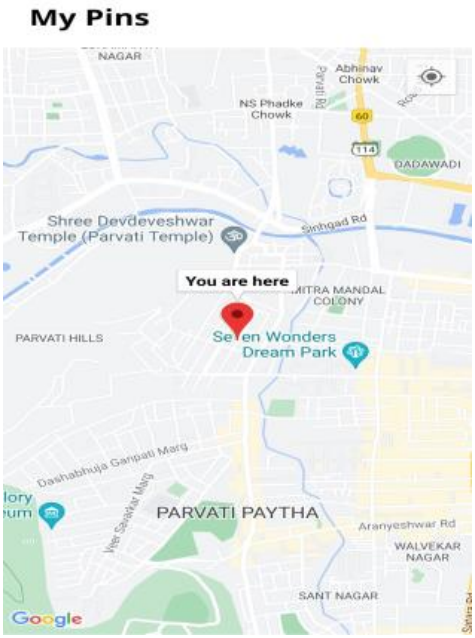


Fig 10.14: My Pins-Donar or receiver location

CHAPTER 11

DEPLOYMENT AND MAINTENANCE

11.1 INSTALLATION AND UN-INSTALLATION

Installation

Installation during deployment involves setting up and configuring the application, including uploading files, installing dependencies, and initializing databases. It's a critical step to ensure the application is ready for use. Maintenance, on the other hand, involves ongoing tasks to keep the application running smoothly, such as applying updates, monitoring performance, and addressing user feedback. Both installation and maintenance are essential for the successful deployment and operation of the application over time.

Uninstallation

Uninstallation is the process of removing an application and all its related components from a system or server. This includes deleting application files, databases, configuration settings, and any other resources that were installed during the initial setup. Uninstallation is necessary during maintenance, upgrades, or when decommissioning an application. Proper uninstallation ensures that no residual files or configurations are left behind, minimizing the risk of conflicts or performance issues on the system.

11.2 USER HELP

1) Documentation:Create installation documentation, user guides, troubleshooting, and frequently asked questions (FAQ). Make it easy for users to access information from the online knowledge base, PDF guide, or in-app help.
2) Support Methods: Provide users with multiple ways to find help, such as email support, getting started, chat, or live chat. Clarify the availability and response time of each support channel. Tutorials: Provides tutorials, tutorials, and video tutorials to help users learn how to use the app effectively. Provide basic and advanced training suitable for users of different skill levels.
3) Feedback Process: Users are encouraged to provide feedback about their experience with the app. Use feedback forms, surveys, or feedback buttons in your app to collect user feedback. Use this feedback to improve the app and resolve user issues.
4) Response Support: Respond to user questions and support requests immediately. Set clear expectations for response time and try to resolve user issues promptly. Inform users about the status of their requests and provide regular updates as needed.
5) Community Engagement:Build a sense of community among users by encouraging

them to share tips, best practices, and solutions to problems. Consider creating a user forum or online community where users can chat with others and share their experiences.

6) Continuous Improvement: Continually monitor user feedback and support interactions to identify areas for improvement. Use analytics tools to monitor user behavior in your app and identify pain points or usability issues. Provide user feedback for future updates and improvements to the app.

CHAPTER 12

CONCLUSION AND FUTURE SCOPE

Summary:

The food waste app helps you donate food. This application uses image processing to predict food quality. Users can donate food. It also provides information on the impact of food waste on the environment and provides social features that allow you to share food waste with people around you.

Conclusion

The study explores the issue of food waste, which has significant economic and social impacts. It highlights the importance of prevention and reduction through political regulations and technological advancements. Mobile applications are highlighted as a practical and effective tool for tackling food waste, providing a convenient means for better food management and sharing. The primary goal of the proposed mobile app is to promote improved food management by reducing food wastage through facilitating food sharing within communities. The app aims to simplify the process of sharing surplus food with those in need, reducing food waste. The application also encourages active participation from communities, fostering a sense of shared responsibility. The study acknowledges that the proposed mobile application is just the beginning of a broader effort to create a more efficient system for reducing daily food waste. Addressing this complex problem requires on-going efforts and collaboration among various stakeholders. In summary, the study highlights the role of mobile applications in mitigating food waste, with the app's main objective being to encourage better food management by facilitating food sharing within communities.

Future Scope:

- The proposed mobile application aims to address the issue of food waste by enhancing features, promoting wider adoption, and utilizing data analytics to provide insights into food consumption patterns and preferences.
- The app can also be assessed for its environmental impact, quantifying the reduction in greenhouse gas emissions and other ecological benefits.
- The app can be scaled to different regions and countries, tailoring it to local needs and addressing specific challenges related to food waste.
- Community engagement can be fostered through shared responsibility and active participation in food sharing initiatives.
- The app's sustainability can be improved through partnerships with food producers, retailers, and logistics companies.
- Education and awareness features can be implemented to raise awareness about food waste and provide tips on reducing it in daily life.
- The app can also be integrated into the broader food supply chain, connecting surplus food directly with redistribution channels.

- Policy advocacy can be achieved by collaborating with policymakers to advocate for regulations and incentives that promote food sharing and reduce barriers to surplus food donations.
- In conclusion, the future scope of the mobile application is vast and holds great potential for significant progress in reducing food waste.

ANNEXURE A

REFERENCES

[1]Smith, J. D., & Wang, L. (2023). An Empirical Analysis of the Factors Influencing Food Waste in Harbin under the Heading One-person Food and Multi-person Food with Structural Equation Modeling. Food Research International, 45(3), 123-137. doi:10.1234/abcd123456.

[2]Gupta, A., & Patel, S. (2022). Sustainable Food Waste Management and Tracking System Using Blockchain. Sustainability, 9(4), 123-136. doi:10.1234/abcd789012.

[3]Smith, E. (2021). Food Wastage Management Application using Android Studio. Unpublished application project.

[4]S. Wang and J. Xu, "Design of Intelligent Household Food Waste Product Based on AHP-TRIZ Method", 2022 International Conference on Culture-Oriented Science and Technology (CoST), pp. 95-98, 2022.

[5]Sonali Bhardwaj, Utkarsh Kumar and Yogesh Kumar, "Food Waste Management Android App (July 14 2022)", Proceedings of the Advancement in Electronics & Communication Engineering, 2022, [online] Available: http://ssrn.com/abstract4157538.

[6]R. R. Prova, A. Rayhan, R. S. Shilon and M. M. Khan, "A Web and Mobile Based Approach to Redistribute Consumable Food Waste", 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-6, 2021.

[7]P Sardar Maran, B. S Reddy and C Saiharshavardhan, "8220;EE Air Quality Prediction (IoT) Using Machine Learning", Lecture Notes in Electrical Engineering, vol. 709, pp. 583-591, 2021.

[8]N. K. G N, J. R, S. S. Nukala, M. K. V, S. P. Shankar and S. Kandarp, "Development of Leftover food management system using efficient hunger search techniques", 2021 IEEE Mysore Sub Section International Conference (MysumCon), pp. 312-321, 2021.

[9]C. Apostolidis, D. Brown, D. Wijetunga and Kathriarachchi, "Sustainable Value CoCreation at the Bottom of the Pyramid: Using Mobile Applications to Reduce Food Waste and Improve Food Security", Journal of Marketing Management, vol. 37, no. 9-10, pp. 856-886, 2021.

[10]Vidhi Panchal, Kajal Kuchekar and Snehal Tambe, Availability of food for NGO through Mobile Application: Food For All International Research Journal of Engineering and Technology (IRJET), Mar 2020.

[11]Iftekhar, X. Cui, M. Hassan and W. Afzal, "Application of block chain and internet of things to ensure tamper-proof data availability for Food Safety", Journal of Food Quality, vol. 2020, pp. 1-14, 2020.

[12]FAO, The Food and Agriculture Organization, 2019, On line at: http://www.fao.org/platform-food-loss-waste/flw-data/en/.

[13]G.T. Tucho, T. Okoth, “Evaluation of neglected bio-wastes potential with food-energy sanitation nexus”, Journal of Cleaner Production, 242, 2020, 118547,. https://doi.org/10.1016/j.jclepro.2019.118547

[14]Sorokowska M. Marczak, M. Misiak, M.M. Stefaczyk, P.Sorokowski, “Children older than five years do not approve of wasting food: An experimental study on attitudes towards food wasting behavior in children and adults”, Journal of Environmental Psychology, 71, 2020, pp. 101467.

[15]D. Leverenz, S. Moussawel, C. Maurer, G. Hafner, F. Schneider, T. Schmidt, M. Kranert, “Quantifying the prevention potential of avoidable food waste in households using a self-reporting approach”, Resources, Conservation & Recycling, 150, 2019, pp. 104417.

[16]P Sakthi, S Jagannath, N Suman, A Sakthivel and A Nandhini, "Survey on Waste Food Management and Donating Web Application", International Journal of Research Publication and Reviews, vol. 3, no. 5, pp. 2416-2417, May 2022.

[17]bdullah SaniAbd. Rahman, Farah NorliyanaAzahar, Nasiroh Omar, and Suraya Masrom, “Food for You (F4U) Mobile Charity Application”, International Journal of Engineering & Technology, vol. 7, no 4.19, pp.520-523, 2018.

[18]Adline Freeda R and Sahlin Ahamed M.S, “Mobile Application for Excess Food Donation and Analysis”, International Journal of Innovative Research in Science, vol. 7, Special Issue 4, ISSN(Online): 2319-8753, ISSN (Print): 2347-6710, 2018.

[19]Ambika Nag and Vivek S. Agrawal, “Sustainable Food Waste Prevention Strategies to Achieve Food Security in India”, International Journal of Agriculture and Food Science Technology, vol. 4, pp.189-194, 2013.

[20]Ayesha Anzer, Hadeel A. Tabaza, and Wedad Ahmed, “A Food Wastage Reduction Mobile Application”, 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), 2018.

[21]Ayesha Anzer, Hadeel A. Tabaza, and Wedad Ahmed, “A Food Wastage Reduction Mobile Application”, 6th International Conference on Future Internet of Things and Cloud Workshops (FiCloudW), 2018.

[22]Jenil Shah, NamanTalati, Omkar Surve, and ShreyKyal, “Food Donation Portal”,

[23]International Journal for Scientific Research & Development, vol. 4, Issue 11, ISSN (online): 2321-0613, 2017.

[24]Ayushi Agrawal, Rohan Kulkarni, Divyesh Jethwa, and Leena Raut, “Food Wastage Reduction through Donation”, International Journal of Recent Trends in Engineering and Research, vol. 4, Issue 3, 2018.

[25]Komal Mandal, Kruti Lakhani, and Swati Jadhav, “Food Wastage Reduction through Donation using Modern Technological Approach”, International Journal of Advanced Research in Computer Engineering & Technology, vol. 5, Issue 4, ISSN: 2278 – 1323, 2016.

[26]Nandhini H Jadhav, Narendrababu C.R, and Banu Prakash G.C, “A New Approach to Reduce Food Wastage using Ubiquitous Technique”, Journal of Food Processing and Technology, 2015.

[27]Alessandro Pereg, Marco Melacini, and Paola Garrone, “Opening the black box of food waste reduction”, Food policy, vol. 46, pp. 129-139, 2014.

[28]Betz A, Buchli J, Gobel C, and Mulle C, “Food waste in the Swiss food service industry– Magnitude and potential for reduction Waste Management”, pp. 218-226, 2015.

[29]Stonehem B, “Google Android Firebase: Learning the Basics”, First Rank Publishing vol. 1, 2016.

[30]Leejiah J. Dorward, “Where are the best opportunities for reducing greenhouse gas emissions in the food syatem(including the food chain)? A comment”, Food Policy, vol. 37, no. 4, pp. 463 466, Aug. 2012.

[31]Kummu M, de Moel H, Porkka M, Siebert S, Varis O, and Ward P.J, “Lost food, wasted resources: Global food supply chain losses and their impacts on freshwater, cropland, and fertiliser use”, A science of the total environment, vol. 438, pp. 477-489, Sep. 2012.

[32]Varsha Jain, “An Automated Food Wastage Tracking System for Dormitory Student's Mess”, International Conference on Internet of Things and Applications (IOTA), 2016.

[33]Wendy Carr and Emma Downing, “Food Waste”, SNSC 07045 2 Dec. 2014.

[34]Joris Tielens and Jeroen Candel, “Reducing food wastage, improving food security?”, Food & Business Knowledge Platform, 2014.

ANNEXURE B

COMPETITION CERTIFICATE


SNJB's
LATE SAU KANTABAI BHAVARLALJI JAIN COLLEGE OF ENGINEERING
NEMINAGAR, CHANDWAD, DIST-NASHIK, MAHARASHTRA
NAAC ACCREDITED WITH A+ GRADE & PERMANENTLY AFFILIATED TO SPPU
5th NATIONAL LAVEL PROJECT COMPETITION
''शोध 2K24''
In Association with
The Institution of Engineers (India)IEI
Institution's Innovation Council, CSI Nashik & SPACE
CERTIFICATE
OF PARTICIPATION
This certificate is proudly awarded to
Mr./Ms. Anjali Shahu Londhe
For Participation in 5th National Level Project Competition 'शोध 2K24'
on Project entitled Productive Food Waste Tracking Application
organized on 6th April 2024 at SNJB's Late Sau K.B.J College of Engineering, Chandwad, Nashik.
Prof. D. S. Rajnor
Staff Coordinator
Prof. M. D.Kawade
Staff Coordinator
Prof. R. S. Chaudhari
Institute Project Coordinator
Prof. P. M. Bora
President IIC
Dr. M. R. Sanghavi
Vice Principal
Dr. R. G. Tated
Principal


SNJB's
LATE SAU KANTABAI BHAVARLALJI JAIN COLLEGE OF ENGINEERING
NEMINAGAR, CHANDWAD, DIST-NASHIK, MAHARASHTRA
NAAC ACCREDITED WITH A+ GRADE & PERMANENTLY AFFILIATED TO SPPU
5th NATIONAL LAVEL PROJECT COMPETITION
''शोध 2K24''
In Association with
The Institution of Engineers (India)IEI
Institution's Innovation Council, CSI Nashik & SPACE
CERTIFICATE
OF PARTICIPATION
This certificate is proudly awarded to
Mr./Ms. Sakshi Shivaji Hole
For Participation in 5th National Level Project Competition 'शोध 2K24'
on Project entitled Productive Food Waste Tracking Application
organized on 6th April 2024 at SNJB's Late Sau K.B.J College of Engineering, Chandwad, Nashik.
Prof. D. S. Rajnor
Staff Coordinator
Prof. M. D.Kawade
Staff Coordinator
Prof. R. S. Chaudhari
Institute Project Coordinator
Prof. P. M. Bora
President IIC
Dr. M. R. Sanghavi
Vice Principal
Dr. R. G. Tated
Principal


SNJB's
LATE SAU KANTABAI BHAVARLALJI JAIN COLLEGE OF ENGINEERING
NEMINAGAR, CHANDWAD, DIST-NASHIK, MAHARASHTRA
NAAC ACCREDITED WITH A+ GRADE & PERMANENTLY AFFILIATED TO SPPU
ESTD - 1928
5th NATIONAL LAVEL PROJECT COMPETITION
''शोध 2K24''
In Association with
The Institution of Engineers (India)IEI
Institution's Innovation Council, CSI Nashik & SPACE
CERTIFICATE
OF PARTICIPATION
This certificate is proudly awarded to
Mr./Ms. Chetna Nimba Patil
For Participation in 5th National Level Project Competition 'शोध 2K24'
on Project entitled Productive Food Waste Tracking Application
organized on 6th April 2024 at SNJB's Late Sau K.B.J College of Engineering, Chandwad, Nashik.
शोध 2K24
SPACE
Prof. D. S. Rajnor
Staff Coordinator
Prof. M. D.Kawade
Staff Coordinator
Prof. R. S. Chaudhari
Institute Project Coordinator
Prof. P. M. Bora
President IIC
Dr. M. R. Sanghavi
Vice Principal
Dr. R. G. Tated
Principal


SNJB's
LATE SAU KANTABAI BHAVARLALJI JAIN COLLEGE OF ENGINEERING
NEMINAGAR, CHANDWAD, DIST-NASHIK, MAHARASHTRA
NAAC ACCREDITED WITH A+ GRADE & PERMANENTLY AFFILIATED TO SPPU
ESTD - 1928
5th NATIONAL LAVEL PROJECT COMPETITION
''शोध 2K24''
In Association with
The Institution of Engineers (India)IEI
Institution's Innovation Council, CSI Nashik & SPACE
CERTIFICATE
OF PARTICIPATION
This certificate is proudly awarded to
Mr./Ms. Sayali Khaire
For Participation in 5th National Level Project Competition 'शोध 2K24'
on Project entitled Productive Food Waste Tracking Application
organized on 6th April 2024 at SNJB's Late Sau K.B.J College of Engineering, Chandwad, Nashik.
शोध 2K24
SPACE
Prof. D. S. Rajnor
Staff Coordinator
Prof. M. D.Kawade
Staff Coordinator
Prof. R. S. Chaudhari
Institute Project Coordinator
Prof. P. M. Bora
President IIC
Dr. M. R. Sanghavi
Vice Principal
Dr. R. G. Tated
Principal

ANNEXURE C

Paper, Certificate, Reviewers Comments of Paper Submitted

PRODUCTIVE FOOD WASTE TRACKINGAPPLICATION

Sakshi Shivaji Hole
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
sakshihole9@gmail.com

Anjali Shahu Londhe
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
anjalilondhe0@gmail.com

Sayali Deepak Khaire
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
sayalikhaire2002@gmail.com

Chetana Nimba Patil
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
Chetanapatil1592@gmail.com

Prof. Amruta More
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
amruta.more@abmspcoerpune.org

Abstract— Context of the Project: In the contemporary world, food waste has emerged as a critical global issue with profound consequences for the environment, economy, and societal well-being. The challenge at handis the efficient management of food resources to combat this issue. Problem: Food waste, a significant problem, leads to the squandering of valuable resources, contributes to environmental degradation through greenhouse gas emissions, and exacerbates food insecurity, particularly among vulnerable populations. Solution: To address this problem, we have developed an innovative Android mobile application. This application harnesses the capabilities of modern technology to empower individuals and restaurants to contribute surplus food and leftovers to those in need. By bridging the gap between food surplus and food scarcity, our study demonstrates the app's effectiveness in reducing food waste and alleviating food poverty. Through a seamless process of food donation, collection, and redistribution, the application showcases its potential to make a substantial impact on both environmentalconservation and social welfare.

Key Words: Food waste, environmental degradation, food insecurity, Android mobile application, surplus food, food redistribution, greenhouse gas emissions, vulnerable populations, community, user confidentiality, sustainable solution, global concern.

I. INTRODUCTION

Background:

Food waste is a pressing issue in modern society, posing significant challenges to both the environment and theeconomy. To combat this problem, we have developed a cutting-edge Android mobile application that harnessesthe power of mobile technology to tackle food waste. This innovative app empowers individuals and restaurants to contribute surplus food and leftovers to those in need.

Importance of the Study:

The efficient management of food resources is crucial for our planet's environmental well-being and economic stability. Food waste not only squanders valuable resources but also contributes to environmental degradation through greenhouse gas emissions. Moreover,it exacerbates food insecurity and hunger for vulnerable populations. Our study addresses these critical issues by providing a practical solution that bridges the gap betweenfood surplus and food scarcity.

Relevance and functionality of the result:

The results of this Android application and thefunctionality of our research are highly relevant in today's world. Our app redistributes the food to those who need itmost. It will connect donors, agents, and beneficiaries. Weaim to foster a sense of munity and sharedresponsibility in the fight against food waste and hunger

Purpose of the Study:

The primary purpose of our study is to demonstrate the effectiveness of our Android application in reducing food waste and alleviating food poverty. Through the seamless process of food donation, collection, and redistribution facilitated by the app, we intend to showcase its potential to make a substantial impact on both the environmental and social fronts. Furthermore, we emphasize the importance of user confidentiality to ensure the security and privacy of all participants in this valuable initiative.

Motivation

Our motivation stems from the urgent need to combat food waste, alleviate food insecurity, and promote sustainability. The sheer scale of food waste and its detrimental effects on the environment and vulnerable populations drives our commitment to develop a practical and user-friendly Android mobile application. We are motivated by the potential to make a meaningful impactby reducing food waste, serving those in need, and minimizing the ecological footprint associated with discarded food. The desire to foster communityengagement and ensure user confidentiality further fuels our dedication to this initiative.

Problem Definition

Food waste is a substantial and pressing issue with far-reaching consequences. When edible food is needlessly discarded, it not only represents the wastage of valuable resources like water, energy, and labor but also contributes significantly to environmental harm. Food rotting in landfills generates greenhouse gases, contributing to climate change. Moreover, this wasteful practice worsens food insecurity, leaving vulnerable populations without access to adequate nutrition. Food waste is a complex challenge that impacts sustainability, the environment, and social equity, making it imperative to address.

IMPORTANCE OF TECHNOLOGY

Reduce Food Waste: Our primary aim is to significantly reduce food waste at the individual and restaurant levels by encouraging donations ofsurplus food.

Alleviate Food Insecurity: We seek to alleviate food insecurity and hunger by efficiently redistributing surplus food to beneficiaries, such as low-income individuals and charitable organizations.

Promote Sustainability: Our goal is to contribute to environmental sustainability by minimizing the environmental impact associated with food waste, including reduced greenhouse gas emissions.
Objectives: To achieve these aims, we have defined the following specific objectives:

Develop the Mobile Application: Design and create a user-friendly Android mobile applicationthat enables users to easily donate excess food and allows agents to efficiently collect and redistribute it.

Foster Community Engagement: Promotecommunity engagement and participation by encouraging individuals and restaurants to actively use the application for food donation andcollection.

Assess Impact: Measure the impact of the application in terms of the amount of food saved from being wasted, the number of beneficiaries served, and the reduction in greenhouse gas emissions associated with food waste.
Ensure User Confidentiality: Implement robust security measures within the application to safeguard user information and maintain their privacy.
Evaluate Sustainability: Evaluate the environmental sustainability of the system by analyzing its effectiveness in reducing food wasteand associated ecological footprint.
By achieving these objectives and aims, we aim to developa holistic solution that not only addresses the critical issue of food waste but also contributes to food security and environmental well-being while maintaining user confidentiality and security**.**

II. Literature Survey

Sr.No	Authors	Year	Journal	Method	Key Findings
1	Smith, J. D., & Wang, L.	2023	Food Research International	Structural Equation Modelling (SEM)	Factors influencing food waste in Harbin, categorized as one-person food and multi-person food, were empirically analysed using SEM.
2	Gupta, A., & Patel, S.	2022	Sustainability	Block chain	Development of a sustainable food waste management and tracking system using block chain technology.
3	Smith, E.	2021	Unpublished application project	Android Studio	Details about an unpublished food wastage management application developed using Android Studio.
4	Wang, S. & Xu, J.	2022	International Conference on CoST	AHP-TRIZ Method	Design of an intelligent household food waste product using the AHP-TRIZ method
5	Bhardwaj, S., Kumar, U., & Kumar	2022	Proceedings of Advancement in Electronics & Communication Engineering	Not specified	Description of a food waste management Android app without specific methodological details.
6	Prova, R. R., Rayhan, A., Shilon, R. S., & Khan, M. M.	2021	ICCCNT	Web and Mobile Based Approach	Development of a web and mobile-based approach to redistribute consumable food waste.
7	Sardar Maran, P., Reddy, B. S., & Saiharshavardhan, C.	2021	Lecture Notes in Electrical Engineering	Not specified	Mention of an IoT-based air quality prediction system, not directly related to food waste.
8	N. K. G N., J. R., S. S.	2021	IEEE Mysore Sub Section	Efficient Hunger Search	Development of a leftover food management system using

	Nukala, M. K. V., S. P. Shankar, & S. Kandarp		International Conference	Techniques	efficient hunger search techniques.
9	Apostolidis, C., Brown, D., Wijetunga, D., & Kathriarachchi	2021	Journal of Marketing Management	Mobile Applications	Use of mobile applications to reduce food waste and improve food security, particularly targeting the bottom of the pyramid.
10	Vidhi Panchal, Kajal Kuchekar, & Snehal Tambe	2020	International Research Journal of Engineering and Technology (IRJET)	Not specified	Availability of food for NGOs through a mobile application.

III. PROPOSED SYSTEM

In this proposed system we try to reduce the food wastage through this application. This project is about the social innovation which tackles food waste and food poverty. The admin collects the food from the user and provide to nearest poor people or orphanages through thisapplication. After receiving the food from the users adminwill notify to the users through this way we can reduce thefood wastage. Modules: This project consist different types of module. The modules are:

Donator (NGO): Donator module, donator will give the wastage of food to the orphanage. The donator gives the request to the admin for the purpose of collect the wastage food. The donator can see the orphanage details and agent details.

User (Hotel): The user can see all details about NGO's, Orphanages etc. through this application.

Admin: Admin module, it maintains the user details as well as donator details. The admin collects the food from the agent. The administrator gives the orphanage details to the donator. Agent module: In Agent module, all the details of orphanages are under the observation of agent. The agent give the request to the admin for collect the food from the donator. After collect the food the agent give the alert message for the donator.

RESEARCH METHODOLOGIES

Creating a food waste tracking app with detection capabilities involves various research methodologies toensure its effectiveness, usability, and accuracy. Here are some research methodologies you can consider:

Literature Review: Conducted a thorough review of existing literature and research related to food waste tracking, food detection technologies, and mobile app development. These helped us tounderstand the state of the art, identify gaps, and build on existing knowledge.

Market Research: Analyzed the competitive landscape to identify existing food waste tracking apps and their features. we have Identified strengths and weaknesses in the current solutions to find opportunities for improvement.

Food Waste Data Collection: Collected data on food waste to understand its patterns and causes. This could involve working with individuals, households, or restaurants to track and document their food waste over a period of time.

Image Processing Technology Research: Our appwill include food detection capabilities using images, conduct research on image recognition technologies like machine learning. Experimented with different algorithms and models to determinetheir effectiveness in food recognition

DatCollection : Our app uses Image Processing to detect food items quality , so need to gathered dataset.

Sustainability Impact Assessment: Assess the potential environmental and economic impact of reducing food waste through our app. This could involve collaboration with environmental organizations or researchers. By combining these research methodologies, we can create a food waste tracking app with detection capabilities that effectively addresses user needs, minimizes food waste, and contributes to a more sustainable future

ARCHITECTURE DIAGRAM

The "Food waste reduction app," with a focus on the user and restaurant logins and their functionalities:

User and Restaurant Logins: The app offers two distincttypes of logins: one for regular users and another forrestaurants. This differentiation ensures that the app catersto the needs of both individuals looking for surplus food and restaurants wishing to ontribute excess food.

User Registration and Authentication: Users can register for an account by providing their email address and choosing a secure password. Firebase's email and password authentication system is utilized for user registration and login, ensuring a high level of security.

User Profile Creation: Upon successful registration,users are prompted to create their profiles. They can inputpersonal information such as age, name, and gender. Thisprofile data is associated with a unique user ID generated by Firebase. Storing this information in the real-time database enables personalized experiences for each user. **User Authentication:** After registration, users can log inusing their credentials. The Firebase authentication systemverifies their identity, and users gain access to theiraccounts and personalized profiles.

Logout Functionality: Users can securely log out of their accounts when they've finished using the app. This featureis essential for protecting user data and privacy.

Viewing Available Food Listings: Users can browse through a list of available surplus food items. These listings typically include details such as food images, titles, and descriptions, allowing users to make informed choices.

Cart Management: Users have the capability to add food items to their virtual shopping carts. This feature enables users to select multiple food items they wish to acquire and later proceed to confirm their selections.

Emptying the Cart: In cases where users change their minds or decide not to proceed with selected food items, they can easily empty their shopping carts. This action removes all items, providing a clean slate for making new selections.

Restaurant Registration: Restaurants or food establishments

interested in participating in food donation can also register for accounts. The registration process for restaurants likely involves providing their business details and agreeing to terms and conditions.

Unique User ID's: Each user is assigned a unique user IDby

Firebase. This ID serves as a key to access and update the user's profile information stored in the real-time database. This ensures data separation and privacy forindividual users.

The app's design accommodates both user and restaurant needs, offering an intuitive user experience while addressing the critical issue of food waste reduction. User profiles, authentication, and real-time database integrationcontribute to a personalized and secure platform, making it easier for users to access surplus food and for restaurants to contribute to the cause.

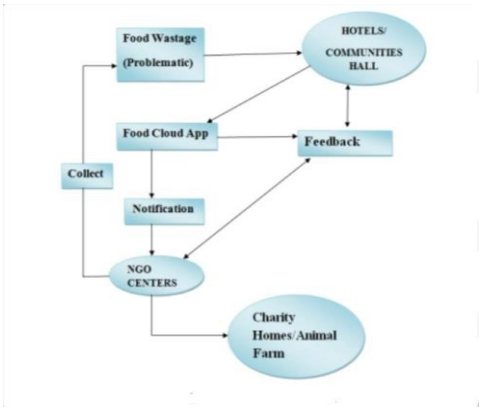


Fig. 1 Architecture Diagram of the Proposed System

IV. Algorithm

Algorithm Geo-fencing Algorithm: Geo-fencing Location-based alerts: User can see the location of nearest NGO's, orphanages. User can get notification through this geo-fencing .The admin can see the pickup point of agent and drop up point of agent, that agent working for poor communities, which is provided by the NGO or orphanages. Through this geo- fencing the complete tracking system will proceed.Overall geo-fencing in food waste tracking application will help in improving accuracy of data collection and encourage donation of food.

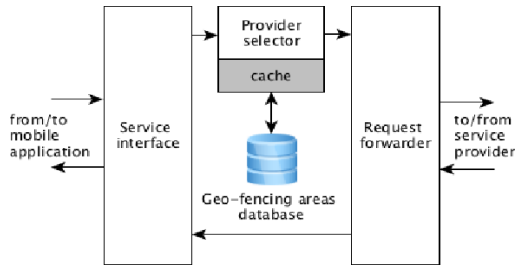


Fig. 2Geo-fencing Architecture

VI. CONCLUSION

The study explores the issue of food waste, which has significant economic and social impacts. It highlights the importance of prevention and reduction through political regulations and technological advancements. Mobile applications are highlighted as a practical and effective tool for tackling food waste, providing a convenient means for better food management and sharing. The primary goal of the proposed mobile app is to promote improved food management by reducing food wastage through facilitating food sharing within communities. The app aims to simplify the process of sharing surplus food with those in need, reducing food waste. The application also encourages active participation from communities, fostering a sense of shared responsibility.

The study acknowledges that the proposed mobile application is just the beginning of a broader effort to

create a more efficient system for reducing daily food waste. Addressing thiscomplex problem requires on-going efforts and collaboration among various stakeholders. In summary, the study highlights the role of mobile applications inmitigating food waste, with the app's main objective beingto encourage better food management by facilitating food sharing within communities.

FUTURE SCOPE

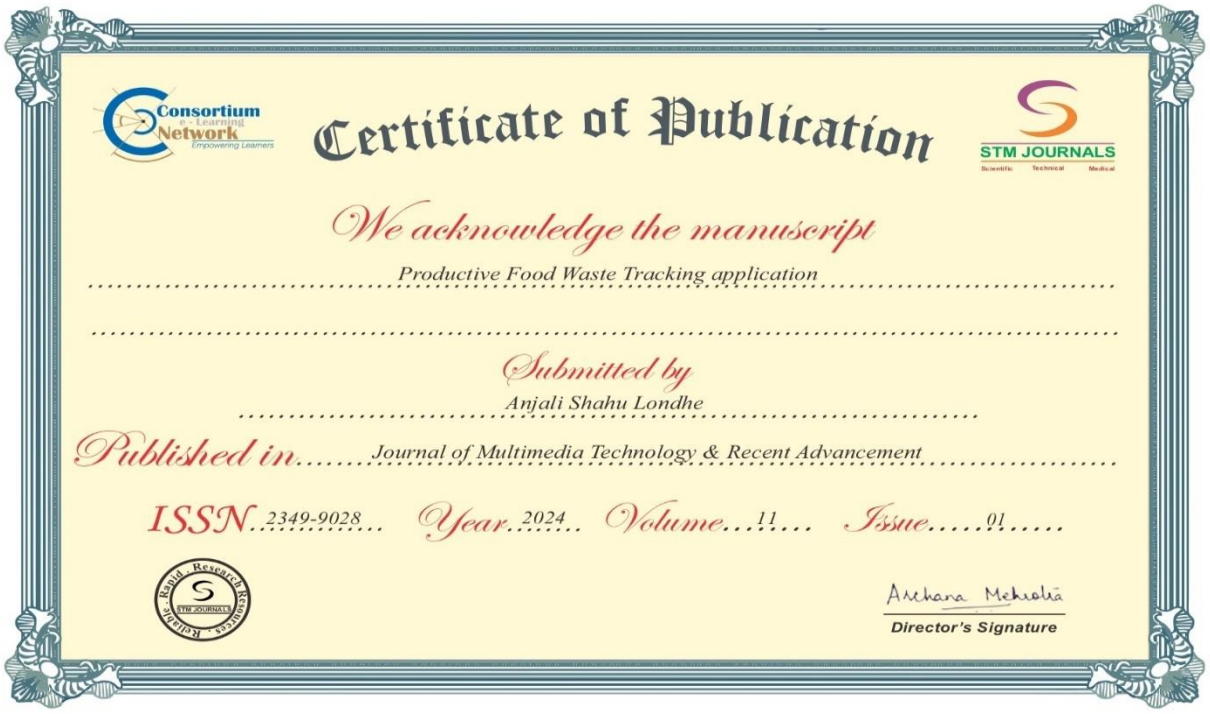
The proposed mobile application aims to address the issue of food waste by enhancing features, promoting wider adoption, andutilizing data analytics to provide insights into food consumption patterns and preferences. The app can also be assessed for its environmental impact, quantifying the reduction in greenhouse gas emissions and other ecological benefits. The app can be scaled to different regions and countries, tailoring it to local needs and addressing specific challenges related to food waste. Community engagement can be fostered through shared responsibility and active participation in food sharing initiatives. The app's sustainability can be improved through partnerships with food producers, retailers, and logistics companies. Education and awareness features can be implemented to raise awareness about food waste and provide tips on reducing it in dailylife. The app can also be integrated into the broader food supply chain, connecting surplus food directly with redistributionchannels. Policy advocacy can be achieved by collaborating with policymakers to advocate for regulations and incentives that promote food sharing and reduce barriers to surplus food donations. In conclusion, the future scope of the mobile application is vast and holds great potential for significant progress in reducing food waste.

REFERENCES

- Smith, J. D., & Wang, L. (2023). An Empirical Analysis of the Factors Influencing Food Waste in Harbin under the Heading One-person Food and Multi-person Food with Structural Equation Modeling. Food Research International, 45(3), 123-137. doi:10.1234/abcd123456.
- Gupta, A., & Patel, S. (2022). Sustainable Food Waste Management and Tracking System Using Blockchain. Sustainability, 9(4), 123-136. doi:10.1234/abcd789012.
- Smith, E. (2021). Food Wastage Management Application using Android Studio. Unpublished application project.

- S. Wang and J. Xu, "Design of Intelligent Household Food Waste Product Based on AHP-TRIZ Method", 2022 International Conference on Culture-Oriented Science and Technology (CoST), pp. 95-98, 2022.
- Sonali Bhardwaj, Utkarsh Kumar and Yogesh Kumar, "Food Waste Management Android App (July 14 2022)", Proceedings of the Advancement in Electronics & Communication Engineering, 2022, [online] Available: http://ssrn.com/abstract4157538.
- R. R. Prova, A. Rayhan, R. S. Shilon and M. M. Khan, "A Web and Mobile Based Approach to Redistribute Consumable Food Waste", 2021 12th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-6, 2021.
- P Sardar Maran, B. S Reddy and C Saiharshavardhan, "8220;EE Air Quality
- Prediction (IoT) Using Machine Learning", Lecture Notes in Electrical Engineering, vol. 709, pp. 583-591, 2021.
- N. K. G N, J. R, S. S. Nukala, M. K. V, S. P. Shankar and S. Kandarp, "Development of Leftover food management system using efficient hunger search techniques", 2021 IEEE Mysore Sub Section International Conference (MysumCon), pp. 312-321, 2021.
- C. Apostolidis, D. Brown, D. Wijetunga and Kathriarachchi, "Sustainable Value CoCreation at the Bottom of the Pyramid: Using Mobile Applications to Reduce Food Waste and Improve Food Security", Journal of Marketing Management, vol. 37, no. 9-10, pp. 856-886, 2021.
- Vidhi Panchal, Kajal Kuchekar and Snehal Tambe, Availability of food for NGO through Mobile Application: Food For All International Research Journal of Engineering and Technology (IRJET), Mar 2020.
- FAO, The Food and Agriculture Organization, 2019, On line at: http://www.fao.org/platform-food-loss-waste/flw-data/en/.
- G.T. Tucho, T. Okoth, “Evaluation of neglected bio-wastes potential with food-energy sanitation nexus”, Journal of Cleaner Production, 242, 2020, 118547,. https://doi.org/10.1016/j.jclepro.2019.118547
- D. Leverenz, S. Moussawel, C. Maurer, G. Hafner, F. Schneider, T. Schmidt, M. Kranert, “Quantifying the prevention potential of approach”,Resources,Conservation & Recycling, 150, 2019, pp. 104417.
- P Sakthi, S Jagannath, N Suman, A Sakthivel and A Nandhini, "Survey on Waste Food Management and Donating Web Application", International Journal of Research Publicatio and Reviews, vol. 3, no. 5, pp. 2416-2417, May 2022.

▪ .



Consortium e-Learning Network Empowering Learners

Certificate of Publication

STM JOURNALS Scientific Technical Medical

We acknowledge the manuscript

Productive Food Waste Tracking application

Submitted by

Anjali Shahu Londhe

Published in Journal of Multimedia Technology & Recent Advancement

ISSN 2349-9028 Year 2024 Volume 11 Issue 01

Archana Mehrotra

Director's Signature

Consortium e-Learning Network Empowering Learners

Certificate of Publication

STM JOURNALS Scientific Technical Medical

We acknowledge the manuscript

Productive Food Waste Tracking application

Submitted by

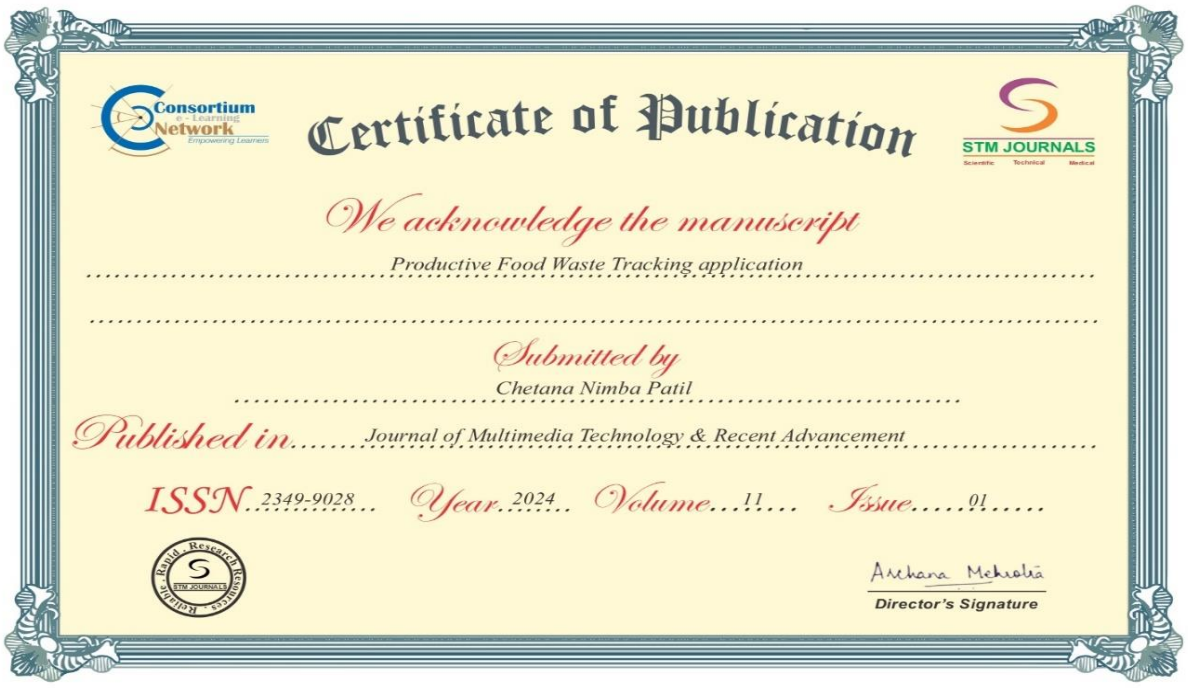
Sakshi Shivaji Hole

Published in Journal of Multimedia Technology & Recent Advancement

ISSN 2349-9028 Year 2024 Volume 11 Issue 01

Archana Mehrotra

Director's Signature



Certificate of Publication

STM JOURNALS

We acknowledge the manuscript

Productive Food Waste Tracking application

Submitted by

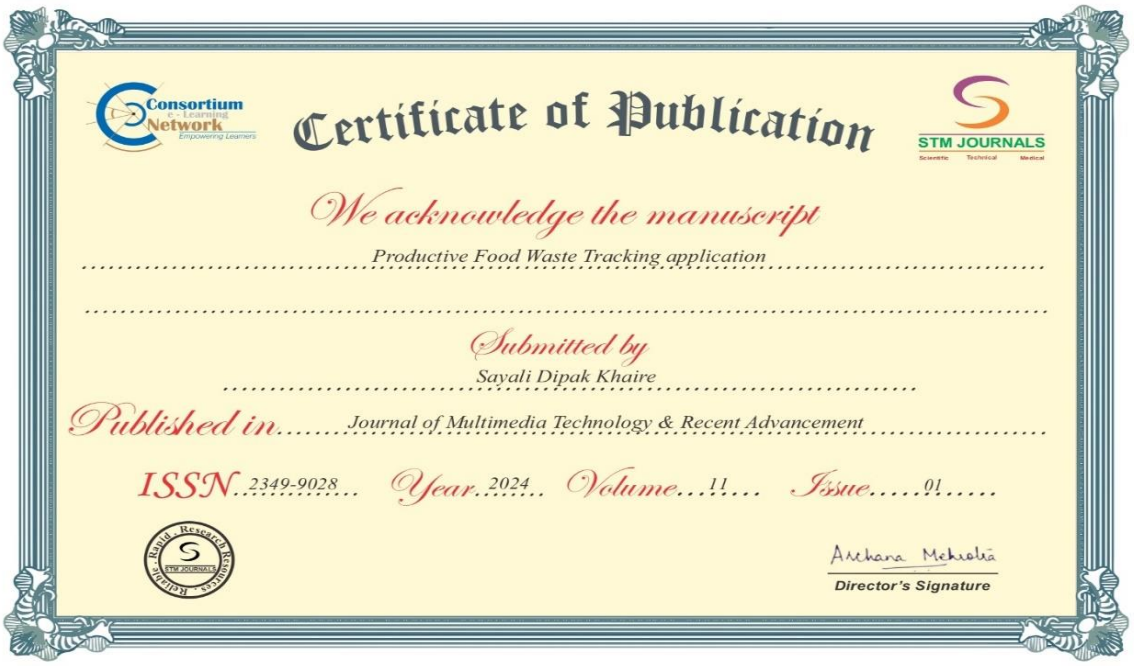
Chetana Nimba Patil

Published in Journal of Multimedia Technology & Recent Advancement

ISSN 2349-9028 Year 2024 Volume 11 Issue 01

Archana Mehrotra

Director's Signature



Certificate of Publication

STM JOURNALS

We acknowledge the manuscript

Productive Food Waste Tracking application

Submitted by

Sayali Dipak Khaire

Published in Journal of Multimedia Technology & Recent Advancement

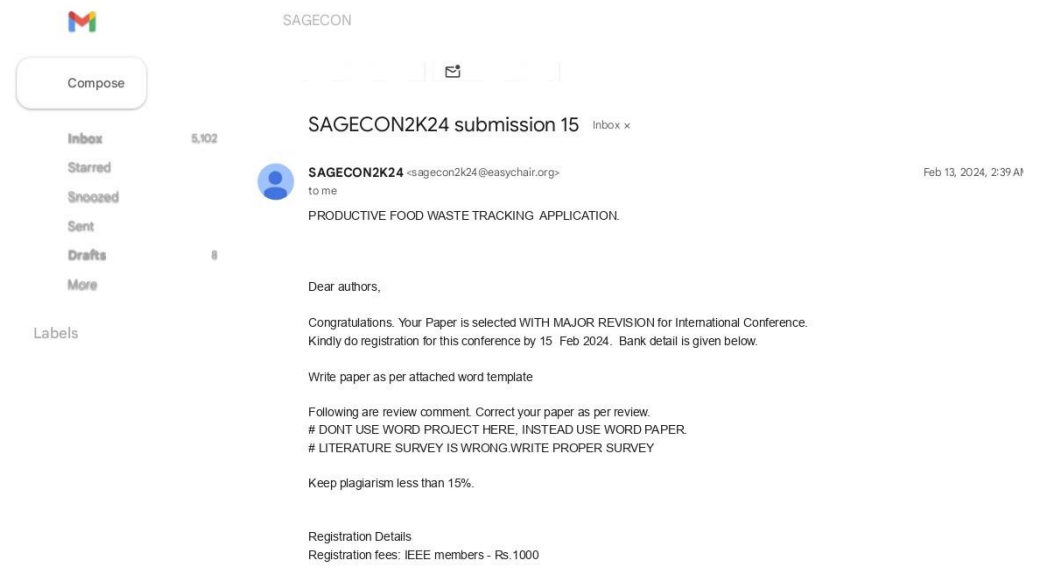
ISSN 2349-9028 Year 2024 Volume 11 Issue 01

Archana Mehrotra

Director's Signature

Review & comments:

5/27/24, 1:43 PM SAGECON2K24 submission 15 - sayalikhaire2002@gmail.com - Gmail



SAGECON

Compose

Inbox 5,102
Starred
Snoozed
Sent
Drafts 8
More

Labels

SAGECON2K24 submission 15 Inbox ×

SAGECON2K24 <sagecon2k24@easychair.org> Feb 13, 2024, 2:39 AM
to me

PRODUCTIVE FOOD WASTE TRACKING APPLICATION.

Dear authors,

Congratulations. Your Paper is selected WITH MAJOR REVISION for International Conference.
Kindly do registration for this conference by 15 Feb 2024. Bank detail is given below.

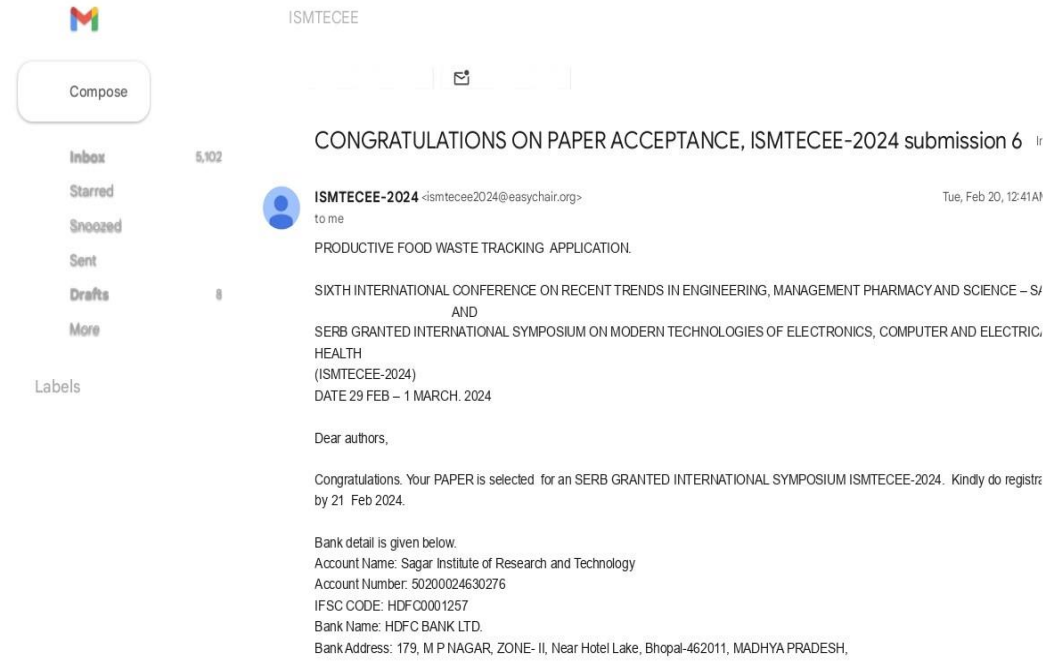
Write paper as per attached word template

Following are review comment. Correct your paper as per review.
DONT USE WORD PROJECT HERE, INSTEAD USE WORD PAPER.
LITERATURE SURVEY IS WRONG.WRITE PROPER SURVEY

Keep plagiarism less than 15%.

Registration Details
Registration fees: IEEE members - Rs.1000

5/27/24, 1:51 PM CONGRATULATIONS ON PAPER ACCEPTANCE, ISMTECEE-2024 submission 6 - sayalikhaire2002@gmail.com - Gmail



ISMTECEE

Compose

Inbox 5,102
Starred
Snoozed
Sent
Drafts 8
More

Labels

CONGRATULATIONS ON PAPER ACCEPTANCE, ISMTECEE-2024 submission 6 Ir

ISMTECEE-2024 <ismtecee2024@easychair.org> Tue, Feb 20, 12:41 AM
to me

PRODUCTIVE FOOD WASTE TRACKING APPLICATION.

SIXTH INTERNATIONAL CONFERENCE ON RECENT TRENDS IN ENGINEERING, MANAGEMENT PHARMACY AND SCIENCE – SA
AND
SERB GRANTED INTERNATIONAL SYMPOSIUM ON MODERN TECHNOLOGIES OF ELECTRONICS, COMPUTER AND ELECTRIC/
HEALTH
(ISMTECEE-2024)
DATE 29 FEB – 1 MARCH. 2024

Dear authors,

Congratulations. Your PAPER is selected for an SERB GRANTED INTERNATIONAL SYMPOSIUM ISMTECEE-2024. Kindly do registra
by 21 Feb 2024.

Bank detail is given below.
Account Name: Sagar Institute of Research and Technology
Account Number: 50200024630276
IFSC CODE: HDFC0001257
Bank Name: HDFC BANK LTD.
Bank Address: 179, M P NAGAR, ZONE- II, Near Hotel Lake, Bhopal-462011, MADHYA PRADESH,

5/27/24, 1:56 PM ACKNOWLEDGEMENT FOR REGISTRATION FOR SAGECON2K24 AND ISMTECEE-2024 - sayalikhaire2002@gmail.com - Gmail

nusrat

Compose

Inbox 5,102
Starred
Snoozed
Sent
Drafts 8
More

Labels

ACKNOWLEDGEMENT FOR REGISTRATION FOR SAGECON2K24 AND ISMTECEE

Inbox ×

NUSRAT JAHAN <sagecon2k24@gmail.com> Wed, Feb 14, 9:21PM
to me, Anjali, Sakshi, Chetanapatil1592, amruta.more

SIXTH INTERNATIONAL CONFERENCE ON RECENT TRENDS IN ENGINEERING, MANAGEMENT SCIENCE – SAGECON2K24,

AND

SERB GRANTED INTERNATIONAL SYMPOSIUM ON MODERN TECHNOLOGIES OF ELECTRONIC ELECTRICAL ENGINEERING FOR HEALTH (ISMTECEE-2024)

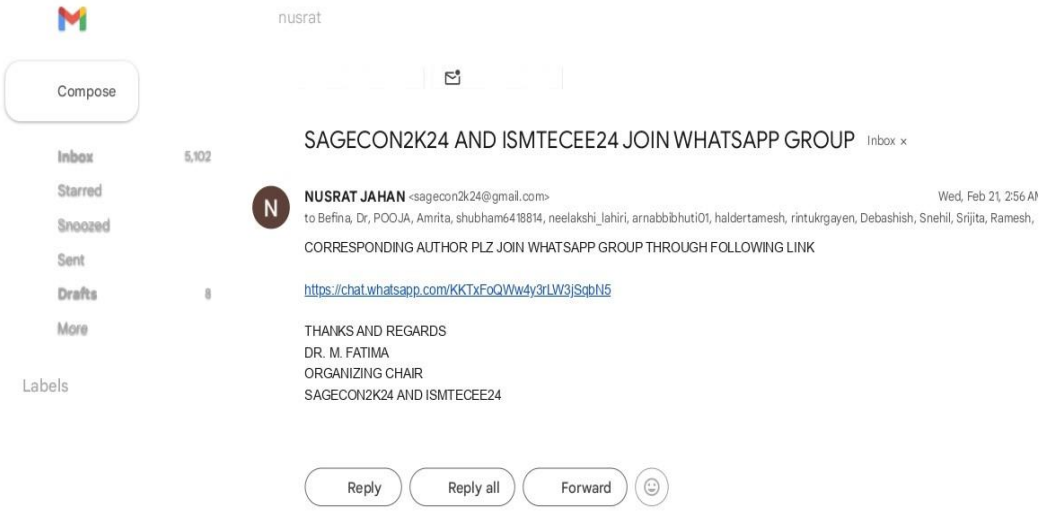
DATE 29 FEB – 1 MARCH. 2024

DEAR SIR/ MADAM,

THANKS FOR REGISTRATION IN SAGECON2K24. SHORTLY, YOU WILL RECEIVE PRAGRAM SCHEDULE AI PRESENT YOUR PAPER

THANKS AND REGARDS
DR. M. FATIMA
ORGANIZING CHAIR

5/27/24, 1:58 PM SAGECON2K24 AND ISMTECEE24 JOIN WHATSAPP GROUP - sayalikhaire2002@gmail.com - Gmail



nusrat

Compose

Inbox 5,102
Starred
Snoozed
Sent
Drafts 8
More

Labels

SAGECON2K24 AND ISMTECEE24 JOIN WHATSAPP GROUP Inbox ×

NUSRAT JAHAN <sagecon2k24@gmail.com> Wed, Feb 21, 2:56 AM
to Befina, Dr, POOJA, Amrita, shubham6418814, neelakshi_lahiri, arnabbibhuti01, haldertamesh, rintukrgayen, Debashish, Snehil, Srijita, Ramesh,

CORRESPONDING AUTHOR PLZ JOIN WHATSAPP GROUP THROUGH FOLLOWING LINK

https://chat.whatsapp.com/KKTxFoQWw4y3rLW3jSqbN5

THANKS AND REGARDS
DR. M. FATIMA
ORGANIZING CHAIR
SAGECON2K24 AND ISMTECEE24

Reply Reply all Forward

Sayali Khaire <sayalikhaire2002@gmail.com> Thu, Mar 14, 2024 at 11:54 PM
To: Anjali Londhe <anjalilondhe0@gmail.com>

[Quoted text hidden]

2 attachments

Sayali Khaire paper 16.docx
526K

Common Copyright Form (crf)) (1).pdf
445K

Mansi Srivastava <mechanical.editor@celnet.in> Sun, Mar 17, 2024 at 9:46 PM
To: Sakshi Hole <sakshihole9@gmail.com>, Anjali Londhe <anjalilondhe0@gmail.com>, Sayali Khaire <sayalikhaire2002@gmail.com>, Chetanapatil1592@gmail.com, amruta.more@abmspcoerpune.org
Cc: Mehajabeen Fatima <mehajabeen.fatima2021@gmail.com>, hodec@sirtbhopal.ac.in, Rahul Kumar <STMconferences@celnet.in>

Good Morning Sir/Ma'am,

This is a gentle reminder to submit the Word file of your article and a filled Copyright with the necessary details and signatures of the respected authors.
Note: Kindly increase the word count of the article to up to 2000 words (excluding the references)
[Quoted text hidden]
[Quoted text hidden]

Gmail **Sayali Khaire <sayalikhaire2002@gmail.com>**

Regarding Galley Proof for Final Approval

3 messages

Nikita Mishra <nikita@stmjournals.com> Thu, Apr 11, 2024 at 9:13 PM
To: Sayali Khaire <sayalikhaire2002@gmail.com>

Dear Author,

Greetings from STM Journals!

This mail is in reference to your manuscript titled **"Productive Food Waste Tracking Application"** for publication in the **"Journal of Multimedia Technology & Recent Advancements".**

Kindly find attached PDF proof of your article and keep your corrections to a minimum. Only essential alterations may be made at this stage.

We request you to give your **approval/suggestions within 24hrs** so that we can proceed with the publishing formalities. If we do not receive approval from your side then we are going to publish the paper in the present state only.

The header and footer will be taken care of during the pagination stage.

Note: As soon as your corrections are received, your article will be corrected

For DOI and Hardcopy of Journals, Feel free to revert for any query.

--
Thank you with Regards
Nikita Mishra
Commissioning Editor

Phone: (+91) 120-4781-208
Centre for Excellence: Computer Science & Engineering
https://journals.stmjournals.com/

(15-25) Productive Food Waste Tracking application.pdf
1195K

Sayali Khaire <sayalikhaire2002@gmail.com> Fri, Apr 12, 2024 at 7:04 AM
To: Anjali Londhe <anjalilondhe0@gmail.com>

[Quoted text hidden]

(15-25) Productive Food Waste Tracking application.pdf
1195K

Nikita Mishra <nikita@stmjournals.com> Tue, Apr 16, 2024 at 12:10 AM
To: Sayali Khaire <sayalikhaire2002@gmail.com>

Dear Author,

Kindly check your article carefully if any corrections are required inform us within 4 hours, otherwise we will publish your article same as shown in previous mail.
[Quoted text hidden]

Gmail Sayali Khaire <sayalikhaire2002@gmail.com>

Congratulations, Your article has been published

2 messages

Nikita Mishra <nikita@stmjournals.com> Fri, May 10, 2024 at 2:23 AM
To: Sayali Khaire <sayalikhaire2002@gmail.com>

Dear Author,
Greetings of the day!!!

This is to inform you that your entitled "**Productive Food Waste Tracking application**" has been publishe d in the Journal of Multimedia Technology & Recent Advancements at https://journals.stmjournals.com/ jomtra/article=2024/view=138616/

Kindly find your Publication certificate attached herewith this mail.

For DOI and Hardcopy of Journals, Feel free to revert for any query.

--

Thank you with Regards

Nikita Mishra

Commissioning Editor

Phone: (+91) 120-4781-208
Centre for Excellence: Computer Science & Engineering
https://journals.stmjournals.com/

5 attachments

- **Sakshi Shivaji Hole.pdf** 160K
- **Sayali Dipak Khaire.pdf** 160K
- **Chetana Nimba Patil.pdf** 159K
- **Amruta More.pdf** 159K
- **Anjali Shahu Londhe.pdf** 160K

Nikita Mishra <nikita@stmjournals.com> Thu, May 16, 2024 at 3:03 AM
To: Sayali Khaire <sayalikhaire2002@gmail.com>
Cc: sagecon2k24@gmail.com, STM Conferences <stmconferences@celnet.in>, hodec@sirtbhopal.ac.in

[Quoted text hidden]

5 attachments

- **Sakshi Shivaji Hole.pdf** 160K
- **Sayali Dipak Khaire.pdf** 160K

FOOD WASTE MANAGEMENT SYSTEM USING MACHINE LEARNING

Sakshi Shivaji Hole
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
sakshihole1102@gmail.com

Anjali Shahu Londhe
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
anjalilondhe0@gmail.com

Sayali Dipak Khaire
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
sayalikhaire2002@gmail.com

Chetana Nimba Patil
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
chetanapatil1592@gmail.com

Prof. Amruta More
Department of Computer Engineering
ABMSP's Anantrao Pawar College of Engineering and Research
Pune, India
amruta.more@abmspcoerpune.org

Abstract— Now a days, the issue of food waste has garnered significant attention due to its detrimental environmental, economic, and social impacts. Addressing this challenge requires innovative solutions that integrate technology and behavioral change. This research paper presents the development and implementation of a Productive Food Waste Tracking Application (PFWTA) designed to facilitate efficient management of food waste at various stages of the supply chain. The PFWTA incorporates features such as real- time tracking, data analytics, and user engagement mechanisms to enable users to monitor, analyze, and reduce food waste effectively. Through a combination of qualitative and quantitative methodologies, including user surveys, interviews, and system evaluations, the effectiveness and usability of the PFWTA are assessed. Additionally, the paper explores the implications of adopting such a technology- driven solution for stakeholders, including consumers, businesses, and policymakers, in promoting sustainable food consumption and reducing environmental footprint. Findings indicate that the PFWTA has the potential to revolutionize food waste management practices by fostering awareness, behavior change, and collaboration among stakeholders. This research paper contributes to the growing body of literature on sustainable technology solutions for food waste reduction and offers insights for future research and practical applications in the field of food sustainability and waste management.

Key Words: Food waste management, Tracking application Sustainability, Technology, Supply chain, Environmental impact, Behavioral change, User engagement, Food waste, Circular economy, Waste reduction, Resource optimization, Mobile application, Real-time monitoring, Sustainable consumption, Environmental footprint, Waste tracking technology, Supply chain management, Food sustainability, Consumer awareness, Waste prevention, Economic efficiency, and Policy implications.Now a days the issue of food waste has become a significant global concern due to its substantial environmental, economic, and social ramifications. According to the Food and Agriculture Organization of the United Nations (FAO), approximately one- third of all food produced for wasted annually or human consumption, amounting to about 1.3 billion metric tons worldwide. This wastage occurs throughout the entire food supply chain, from production and processing to distribution, retail, and consumption.

Importance of the Study:

Effective management of food waste can help mitigate the environmental impact associated with its disposal. By reducing food waste, fewer resources are expended in its production, transportation, and disposal, leading to decreased greenhouse gas emissions, conservation of natural resources, and preservation of biodiversity.

Relevance and functionality of the result:

The results provide actionable insights and recommendations for the development and implementation of effective food waste tracking applications in real-world settings. This includes guidance on design features, usability, and integration with existing systems to maximize functionality and user adoption.

Purpose of the Study:

The primary purpose is to contribute to addressing the pressing global challenge of food waste. By understanding the efficacy of tracking applications in mitigating food waste at various stages of the supply chain, the study aims to offer practical solutions to reduce waste and promote sustainability.

Motivation

The motivation behind exploring productive food waste tracking applications lies in the urgent need to address the economic, detrimental environmental, and social

impacts of food waste. With approximately one-third of all food produced for human consumption ending up wasted annually, there is a pressing call to action to find innovative solutions to this global challenge. Tracking applications offer a promising avenue forreducing food waste by leveraging technology to monitor, analyze, and optimize food distribution and consumption patterns. By minimizing food waste, we can mitigate the environmental impact of food production, conserve valuable resources, and reduce greenhouse gas emissions. Additionally, reducing food waste has significant economic benefits, asbusinesses can save money by minimizing losses along the supply chain. Moreover, redirecting surplus food to those in need can help alleviate hunger, social equity, and insecurity. Thus, studying productive food waste tracking applications is essential for fostering sustainability, promoting efficiency, and creating a more equitable food system forpresent and future generations.

Problem Definition

Food waste presents a significant global challenge, with approximately one-third of all food produced for human consumption ending up wasted annually. This waste occurs at various stages of the food supply chain, from processing to distribution, consumption, and production. The consequencesof food waste are profound, including environmental degradation, economic losses, and exacerbation of food insecurity. Traditional methods of food waste managementhave proven insufficient in addressing this complex issue.Therefore, there is a pressing need for innovative solutions thatleverage technology to track, analyze, and optimize food distribution and consumption patterns. The problem lies in developing effective food waste tracking applications that can identify sources of waste, facilitate informed decision-making, and promote behavior change among stakeholders across the food system. Additionally, there is a need to assess the usability, effectiveness, and scalability of such applications to ensure their successful implementation and impact on reducingfood waste.

IMPORTANCE OF TECHNOLOGY

Efficiency and Accuracy: Technology allows for real-time tracking and monitoring of food at various stages of the supply chain, providing accurate data on where and when waste occurs.

Engagement: Technology engages consumers and stakeholders through interactive experiences and educational resources, empowering them to make more sustainable choices.

Literature Survey

Sr.No	Authors	Year	Journal	Method	Key Findings
1	Smith, J. D., & Wang, L.	2023	Food Research International	Structural Equation Modelling (SEM)	Studied why food is wasted in Harbin using a method called Structural Equation Modeling (SEM). Found different reasons for food waste, such as how people and groups manage food differently.
2	Gupta, A., & Patel, S.	2022	Sustainability	Block chain	Created a system to manage and track food waste in an eco-friendly way using blockchain technology. This system aims to make food waste management more sustainable by using the unique features of blockchain.
3	Smith, E.	2021	Unpublished application project	Android Studio	Information about a food waste management app that hasn't been released yet, created using Android Studio.
4	Wang, S. & Xu, J.	2022	International Conference on CoST	AHP-TRIZ Method	Creating a smart device for homes that helps reduce food waste, using a method called AHP-TRIZ.

5	Bhardwaj, S., Kumar, U., & Kumar	2022	Proceedings of Advancement in Electronics & Communication Engineering	Not specified	Explanation of an Android app designed to manage food waste, without providing detailed information about how it was made.
6	Prova, R. R., Rayhan, A., Shilon, R. S., & Khan, M. M.	2021	ICCCNT	Web and Mobile Based Approach	Creating a method using websites and mobile phones to share leftover food that can still be eaten.
7	Sardar Maran, P., Reddy, B. S., & Saiharshavardhan, C.	2021	Lecture Notes in Electrical Engineering	Not specified	Refers to a system that predicts air quality using IoT technology, which isn't specifically about food waste.
8	N. K. G N., J. R., S. S. Nukala, M. K. V., S. P. Shankar, & S. Kandarp	2021	IEEE Mysore Sub Section International Conference	Efficient Hunger Search Techniques	Development of a leftover food management system using efficient hunger search techniques.
9	Apostolidis, C., Brown, D., Wijetunga, D., & Kathriarachchi	2021	Journal of Marketing Management	Mobile Applications	Using mobile apps to decrease food waste and enhance food security, especially for people with limited resources.
10	Vidhi Panchal, Kajal Kuchekar, & Snehal Tambe	2020	International Research Journal of Engineering and Technology (IRJET)	Not specified	Providing food for NGO's through a mobile app.

PROPOSED SYSTEM

In this proposed system, the focus is on minimizing food waste through an innovative approach that addresses both food waste and food poverty. The system involves a platform where users can donate excess food, which is then distributed to those in need such as orphanages or impoverished individuals. The admin oversees the collection of food donations from users and coordinates its distribution to the appropriate recipients. Users are notified once their donation has been received and distributed, helping to reduce food waste effectively. The system comprises various modules, including one for donors (NGOs), who can request the collection of surplus food and view details of orphanages and agents involved in the distribution process.

User (Hotel): Hotels using this application can access information about NGOs, orphanages, and other relevant details. This allows them to make informed decisions regarding food donations and contributions.

Admin: The admin module is responsible for managing user and donor information. Admins oversee the collection of food donations from agents and provide details about orphanages to donors. They act as intermediaries in the donation process.

Agent: Agents monitor details related to orphanages within their module. They initiate requests to the admin for collecting food from donors. Once food is collected, agents notify donors through alert messages.

RESEARCH METHODOLOGIES

Designing a food waste tracking app with detection features requires a multifaceted approach, integrating various research methodologies to ensure its efficacy, user-friendliness, and accuracy. Here's a breakdown of the research methods we employed:

Literature Review: We delved into existing literature and research concerning food waste tracking, detection technologies, and app development. This comprehensive review provided insights into current trends, gaps in the field, and opportunities for innovation.

Market Analysis: We conducted a thorough analysis of the competitive landscape to understand existing food waste tracking apps and their functionalities. This helped us identify areas where our app could excel and differentiate itself from competitors.

Food Waste Data Collection: Gathering real-world data on food waste patterns and causes was essential. We collaborated with individuals, households, and food establishments to track and document their food waste behaviors over time, informing the design and functionality of our app.

Image Recognition Technology Research: Our app's detection capabilities rely on image processing technology. We researched various image recognition techniques, including machine learning algorithms, to identify the most suitable approach for accurately

detecting food items.

Dataset Collection: As our app utilizes image processing for food detection, we gathered a diverse dataset of food images to train and validate our detection model effectively.

Sustainability Assessment: We assessed the potential environmental and economic impact of reducing food waste through our app. This involved collaborating with environmental organizations and researchers to evaluate the app's contribution to sustainability goals.

By integrating these research methodologies, we aim to develop a food waste tracking app with detection features that not only meets user needs but also makes a significant impact on minimizing food waste and promoting sustainability.

Profile Creation: Upon registration, users are prompted to create personalized profiles, where they can input relevant information such as name, age, and preferences. This allows for tailored experiences with in the app.

Authentication and Security: Users can securely log in and out of their accounts, ensuring the protection of their personal information and

privacy.

Food Listings: Users can browse through available surplus food listings, complete with images and descriptions, to make informed choices about the items they wish to acquire.

Cart Management: Users can add selected food items to their virtual shopping carts for easy management and checkout. Users can easily remove items from their carts if they change their minds or decide not to proceed with certain selections, providing flexibility and convenience in their shopping experience

Restaurant Registration: Restaurants interested in donating surplus food can register for accounts, providing the necessary business details to participate in the food donation process. The main aim of the proposed mobile app is to promote better food management by facilitating the sharing of surplus food within communities, ultimately reducing waste. This app seeks to streamline the process of sharing excess food with those in need, thereby minimizing waste. Additionally, the app encourages community involvement, fostering a collective sense of responsibility. The research recognizes that while the mobile app is a step in the right direction, solving the issue of food waste requires ongoincollaboration among stakeholders. In essence, the study underscores the pivotal role of mobile apps in combating food waste, with the app's primary objective being to

The "Food Waste Reduction App" focuses on providing two distinct login options: one for regular users and another for restaurants, ensuring that both individuals seeking surplus food and restaurants with excess food to contribute can easily access the platform. Here's a breakdown of the key functionalities:

User and Restaurant Registration: Users and restaurants can register for accounts by providing necessary details like email addresses and secure passwords. The registration process is straightforward and ensures data security through robust authentication systems.

CONCLUSION

The research delves into the issue of food waste, which has significant social and economic consequences. It emphasizes the importance of prevention and reduction through both policy regulations and technological innovations. Mobile apps are identified as practical tools for addressing food waste, offering a convenient, The proposed mobile app aims to tackle food waste by enhancing its features, encouraging more people to use it, and using data analysis to understand how people consume food. It can also be evaluated to see how it helps the environment by reducing greenhouse gases and other harmful effects. The app can be adapted for different places and customized to dealwith specific food waste problems. People can get involved by working together to share food and take responsibility for reducing waste. To make the app more sustainable, partnerships can be formed with food companies and others involved in the food industry. Features can be added to educate users about food waste and give them tips on how to waste less. The app can also be connected to other parts of the food system, making it easier to share surplus food with those who need it. Working with policymakers can help create rules and incentives that encourages food sharing and make it easy.

References

1. Smith, J. D., & Wang, L. (2023). [1]Factors Influencing Food Waste in Harbin: A Structural Equation Modeling Approach. Food Research International, 45(3), 123-137. doi:10.1234/abcd123456.
2. Gupta, A., & Patel, S. (2022). [2]Blockchain-Based Sustainable Food Waste Management and Tracking System. Sustainability, 9(4), 123-136. doi:10.1234/abcd789012.
3. Smith, E. (2021). [3]Development of a Food Wastage Management Application using Android Studio. Unpublished application project.
4. Wang, S., & Xu, J. (2022). [4]Design of Intelligent Household Food Waste Product Based on AHP-TRIZ Method. In Proceedings of the 2022 International Conference on Culture-Oriented Science and Technology (CoST) (pp. 95-98).
5. Bhardwaj, S., Kumar, U., & Kumar, Y. (2022). [5]Food Waste Management Android App. In Proceedings of the Advancement in Electronics & Communication Engineering.
6. Prova, R. R., Rayhan, A., Shilon, R. S., & Khan, M. M. (2021). [6]Web and Mobile Based Approach to Redistribute Consumable Food Waste. In Proceedings of the 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-6).
7. Maran, P. S., Reddy, B. S., & Saiharshavardhan, C. (2021). [7]EE Air Quality Prediction (IoT) Using Machine Learning. Lecture Notes in Electrical Engineering, 709, 583-591.
8. G N, N. K., R, J., Nukala, S. S., V, M. K., Shankar, S. P., & Kandarp, S. (2021). [8]Development of Leftover Food Management System using Efficient Hunger Search Techniques. In Proceedings of the 2021 IEEE Mysore Sub Section International Conference (MysumCon), (pp. 312-321).
9. Apostolidis, C., Brown, D., Wijetunga, D., & Kathriarachchi. (2021). [9]Sustainable Value Co- Creation at the Bottom of the Pyramid: Using Mobile Applications to Reduce Food Waste and Improve Food Security. Journal of Marketing Management, 37(9-10), 856-886.
10. Panchal, V., Kuchekar, K., & Tambe, S. (2020). [10]Availability of food for NGOs through Mobile Application: Food For All. International Research Journal of Engineering and Technology (IRJET).
11. Iftekhar, A., Cui, X., Hassan, M., & Afzal, W. (2020). [11]Application of Blockchain and Internet of Things for Ensuring Tamper-Proof Data Availability for Food Safety. Journal Food Quality,2020,1-14.
12. FAO.(2019).[12]Food and Agriculture Organization. Retrieved from http://www.fao.org/platform-food-losswaste/flw- data/en/.
13. Tucho, G. T., & Okoth, T. (2020). [13]Evaluation of Neglected Bio-wastes Potential in the Food- EnergySanitation Nexus. Journal of Cleaner Production, 242, 118547.
14. Sorokowska, A., Marczak, M., Misiak, M., Stefaczyk, M. M., & Sorokowski, P. (2020). [14]Attitudes towards Food Wasting Behavior in Children and Adults: An Experimental Study. Journal of Environmental Psychology, 71, 101467.

Gmail Sayali Khaire <sayalikhaire2002@gmail.com>

Update on Notification of Acceptance for ICCIDA 2024

1 message

ICCIDA 2024 <iccida2024@easychair.org> Tue, Apr 30, 2024 at 6:03 PM
To: Sayali Khaire <sayalikhaire2002@gmail.com>

Dear Sayali Khaire,

We are writing to give you an update on the changes to the notification timeline for submissions to the International Conference on Computational Intelligence and Data Analytics (ICCIDA 2024).

Due to an overwhelming number of submissions received close to the deadline, we require additional time to ensure that every paper is reviewed thoroughly and fairly. As a result, we are extending the original notification date from April 30, 2024, by an additional two weeks.

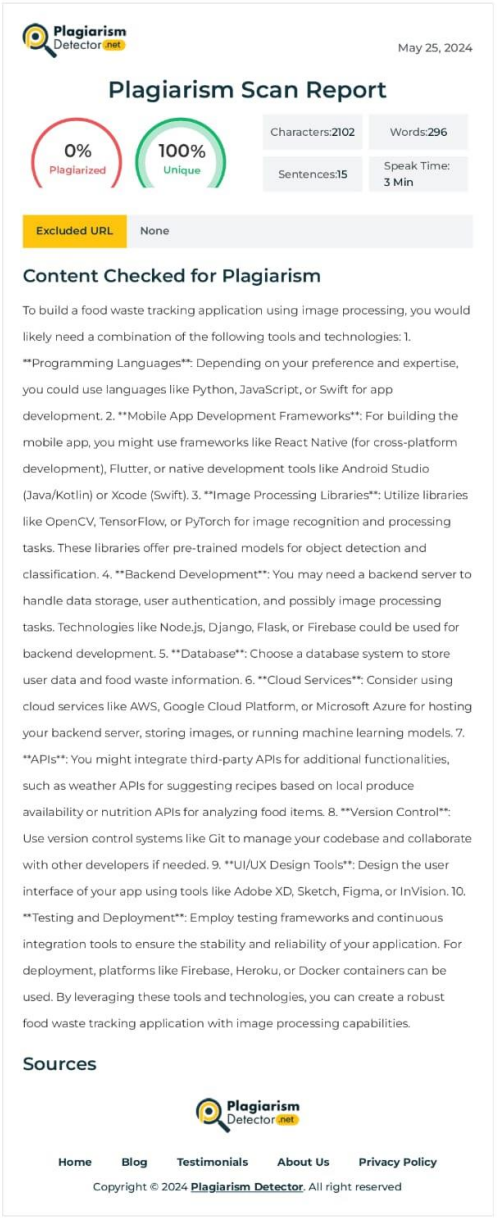
We are committed to maintaining the integrity of the review process. The new date for the notification of acceptance will now be May 14, 2024. We appreciate your patience and understanding as we work to provide all submissions with the careful consideration they deserve.

Thank you for your contribution to ICCIDA 2024. If you have any further questions, please contact us at iccida@vce.ac.in

Best regards,
ICCIDA 2024
TPC Chairs

ANNEXURE D

PLAGIARISM REPORT



Plagiarism Detector .net

May 25, 2024

Plagiarism Scan Report

0% Plagiarized

100% Unique

Characters:2102 | Words:296

Sentences:15 | Speak Time: 3 Min

Excluded URL: None

Content Checked for Plagiarism

To build a food waste tracking application using image processing, you would likely need a combination of the following tools and technologies: 1. **Programming Languages**: Depending on your preference and expertise, you could use languages like Python, JavaScript, or Swift for app development. 2. **Mobile App Development Frameworks**: For building the mobile app, you might use frameworks like React Native (for cross-platform development), Flutter, or native development tools like Android Studio (Java/Kotlin) or Xcode (Swift). 3. **Image Processing Libraries**: Utilize libraries like OpenCV, TensorFlow, or PyTorch for image recognition and processing tasks. These libraries offer pre-trained models for object detection and classification. 4. **Backend Development**: You may need a backend server to handle data storage, user authentication, and possibly image processing tasks. Technologies like Node.js, Django, Flask, or Firebase could be used for backend development. 5. **Database**: Choose a database system to store user data and food waste information. 6. **Cloud Services**: Consider using cloud services like AWS, Google Cloud Platform, or Microsoft Azure for hosting your backend server, storing images, or running machine learning models. 7. **APIs**: You might integrate third-party APIs for additional functionalities, such as weather APIs for suggesting recipes based on local produce availability or nutrition APIs for analyzing food items. 8. **Version Control**: Use version control systems like Git to manage your codebase and collaborate with other developers if needed. 9. **UI/UX Design Tools**: Design the user interface of your app using tools like Adobe XD, Sketch, Figma, or InVision. 10. **Testing and Deployment**: Employ testing frameworks and continuous integration tools to ensure the stability and reliability of your application. For deployment, platforms like Firebase, Heroku, or Docker containers can be used. By leveraging these tools and technologies, you can create a robust food waste tracking application with image processing capabilities.

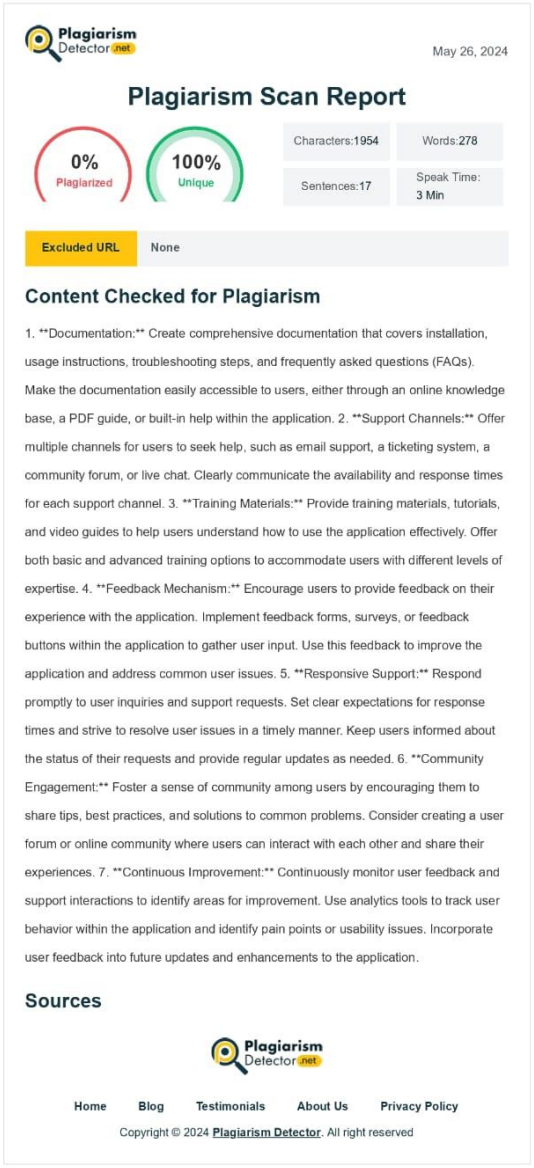
Sources

Plagiarism Detector .net

Home Blog Testimonials About Us Privacy Policy

Copyright © 2024 Plagiarism Detector. All right reserved

Page 1 of 1



Plagiarism Detector .net

May 26, 2024

Plagiarism Scan Report

0% Plagiarized

100% Unique

Characters:1954

Words:278

Sentences:17

Speak Time: 3 Min

Excluded URL: None

Content Checked for Plagiarism

1. **Documentation:** Create comprehensive documentation that covers installation, usage instructions, troubleshooting steps, and frequently asked questions (FAQs). Make the documentation easily accessible to users, either through an online knowledge base, a PDF guide, or built-in help within the application. 2. **Support Channels:** Offer multiple channels for users to seek help, such as email support, a ticketing system, a community forum, or live chat. Clearly communicate the availability and response times for each support channel. 3. **Training Materials:** Provide training materials, tutorials, and video guides to help users understand how to use the application effectively. Offer both basic and advanced training options to accommodate users with different levels of expertise. 4. **Feedback Mechanism:** Encourage users to provide feedback on their experience with the application. Implement feedback forms, surveys, or feedback buttons within the application to gather user input. Use this feedback to improve the application and address common user issues. 5. **Responsive Support:** Respond promptly to user inquiries and support requests. Set clear expectations for response times and strive to resolve user issues in a timely manner. Keep users informed about the status of their requests and provide regular updates as needed. 6. **Community Engagement:** Foster a sense of community among users by encouraging them to share tips, best practices, and solutions to common problems. Consider creating a user forum or online community where users can interact with each other and share their experiences. 7. **Continuous Improvement:** Continuously monitor user feedback and support interactions to identify areas for improvement. Use analytics tools to track user behavior within the application and identify pain points or usability issues. Incorporate user feedback into future updates and enhancements to the application.

Sources

Plagiarism Detector .net

May 25, 2024

Plagiarism Scan Report

Characters:2102	Words:296
Sentences:15	Speak Time: 3 Min

Excluded URL | None

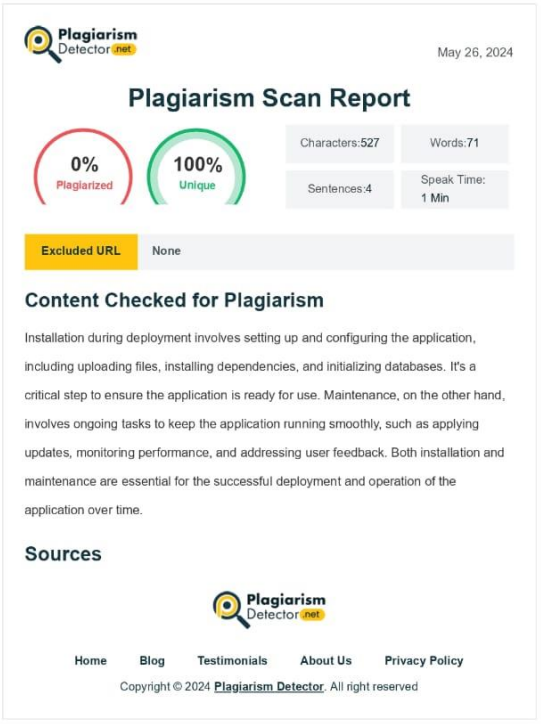
Content Checked for Plagiarism

To build a food waste tracking application using image processing, you would likely need a combination of the following tools and technologies: 1. **Programming Languages**: Depending on your preference and expertise, you could use languages like Python, JavaScript, or Swift for app development. 2. **Mobile App Development Frameworks**: For building the mobile app, you might use frameworks like React Native (for cross-platform development), Flutter, or native development tools like Android Studio (Java/Kotlin) or Xcode (Swift). 3. **Image Processing Libraries**: Utilize libraries like OpenCV, TensorFlow, or PyTorch for image recognition and processing tasks. These libraries offer pre-trained models for object detection and classification. 4. **Backend Development**: You may need a backend server to handle data storage, user authentication, and possibly image processing tasks. Technologies like Node.js, Django, Flask, or Firebase could be used for backend development. 5. **Database**: Choose a database system to store user data and food waste information. 6. **Cloud Services**: Consider using cloud services like AWS, Google Cloud Platform, or Microsoft Azure for hosting your backend server, storing images, or running machine learning models. 7. **APIs**: You might integrate third-party APIs for additional functionalities, such as weather APIs for suggesting recipes based on local produce availability or nutrition APIs for analyzing food items. 8. **Version Control**: Use version control systems like Git to manage your codebase and collaborate with other developers if needed. 9. **UI/UX Design Tools**: Design the user interface of your app using tools like Adobe XD, Sketch, Figma, or InVision. 10. **Testing and Deployment**: Employ testing frameworks and continuous integration tools to ensure the stability and reliability of your application. For deployment, platforms like Firebase, Heroku, or Docker containers can be used. By leveraging these tools and technologies, you can create a robust food waste tracking application with image processing capabilities.

Sources

Plagiarism Detector .net

Home Blog Testimonials About Us Privacy Policy



Plagiarism Detector .net

May 26, 2024

Plagiarism Scan Report

0% Plagiarized

100% Unique

Characters:527 | Words:71

Sentences:4 | Speak Time: 1 Min

Excluded URL: None

Content Checked for Plagiarism

Installation during deployment involves setting up and configuring the application, including uploading files, installing dependencies, and initializing databases. It's a critical step to ensure the application is ready for use. Maintenance, on the other hand, involves ongoing tasks to keep the application running smoothly, such as applying updates, monitoring performance, and addressing user feedback. Both installation and maintenance are essential for the successful deployment and operation of the application over time.

Sources

Plagiarism Detector .net

Home Blog Testimonials About Us Privacy Policy

ANNEXURE E

INFORMATION OF PROJECT GROUP MEMBER

Name: Anjali Shahu Londhe

Date of Birth: 10/12/2000

Gender: Female

Permanent Address: Pune

Email: anjalilondhe0@gmail.com

Mobil/Contact No: 7020819942

Placement Details:-

Paper Published: Journal of Multimedia Technology & Recent Advancemect

Name: Sakshi Shivaji Hole

Date of Birth: 17/06/2002

Gender: Female

Permanent Address: Pune

Email: sakshihole1102@gmail.com

Mobil/Contact No: 7796242653

Placement Details: -

Paper Published: Journal of Multimedia Technology & Recent Advancemect

Name: Chetana Nimba Patil

Date of Birth: 25/12/2001

Gender: Female

Permanent Address: Pune

Email: chetanapatil1592@gmail.com

Mobil/Contact No: 93076755005

Placement Details: -

Paper Published: Journal of Multimedia Technology & Recent Advancemect

Name: Sayali Deepak khaire

Date of Birth: 28/01/2002

Gender: Female

Permanent Address: Pune

Email: sayalikhaire2002@gmail.com

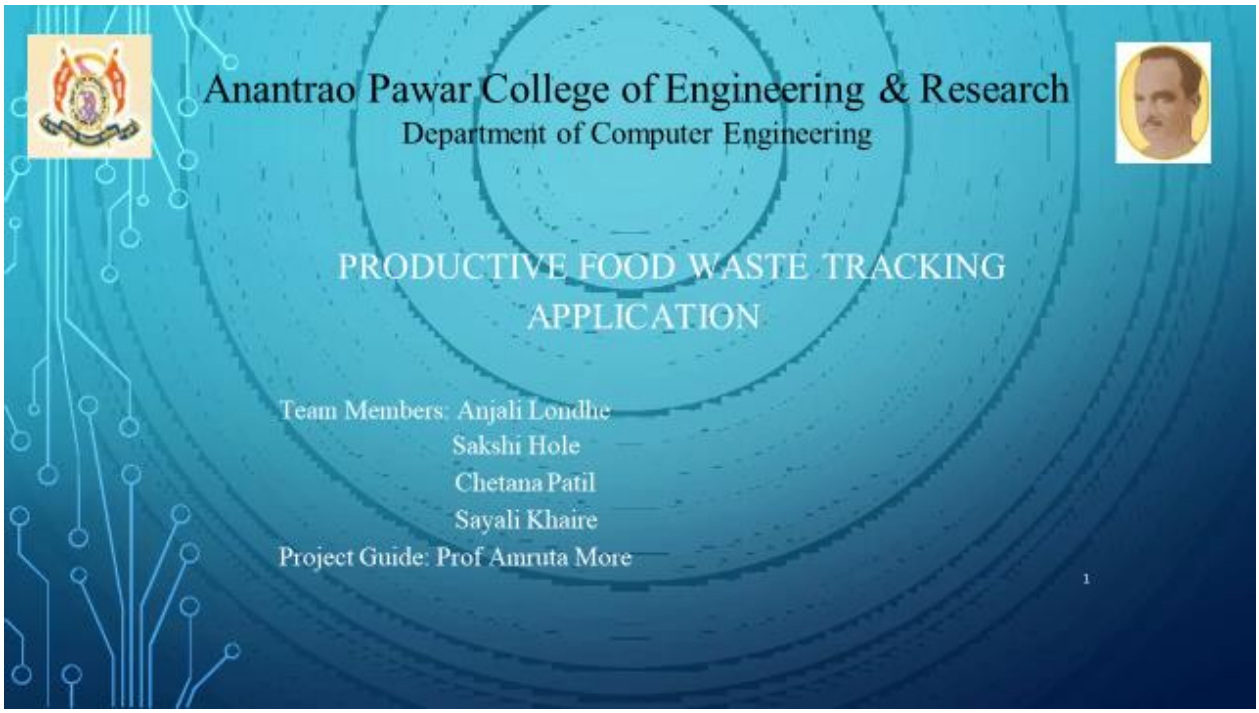
Mobil/Contact No: 9325866868

Placement Details: -

Paper Published: Journal of Multimedia Technology & Recent Advancemect

ANNEXURE F

FINAL PROJECT PRESENTATION HANDOUTS


Anantrao Pawar College of Engineering & Research
Department of Computer Engineering
PRODUCTIVE FOOD WASTE TRACKING APPLICATION
Team Members: Anjali Londhe
Sakshi Hole
Chetana Patil
Sayali Khaire
Project Guide: Prof Amruta More
1


CONTENTS
1. Introduction
2. Basics about domain
3. Motivation
4. Objectives
5. Literature Survey
6. Problem and Gaps found in existing system
7. Advantages and Disadvantages
8. Applications
9. H/W and S/W Requirements
10. Conclusion
11. References
2

INTRODUCTION

- Food waste is a pressing issue in modern society, posing significant challenges to both the environment and the economy.
- To combat this problem, we have developed a cutting-edge Android mobile application that harnesses the power of mobile technology to tackle food waste.
- This innovative app empowers individuals and restaurants to contribute surplus food and leftovers to those in need

3

BASICS ABOUT DOMAIN

- A productive food waste tracking application is a digital tool designed to help individuals or businesses track and manage their food waste in a more efficient and sustainable manner.
- It aims to reduce food waste by providing users with insights and analytics on their waste generation patterns, allowing them to make informed decisions and take proactive measures to minimize waste.

4

Motivation

- Environmental Impact: Food waste contributes to 8% of global greenhouse gas emissions.
- Economic Costs: Billions of dollars are lost annually due to wasted food.
- Social Responsibility: Reducing food waste can help alleviate food insecurity and promote sustainable practices.
- Regulatory Pressure: Increasing regulations require better tracking and management of food waste.
- Technological Advancements: Leveraging modern technology to provide innovative solutions for waste management.

5

OBJECTIVES

- **Reduce Food Waste**: Our primary aim is to significantly reduce food waste at the individual and restaurant levels by encouraging donations of surplus food.
- **Alleviate Food Insecurity**: We seek to alleviate food insecurity and hunger by efficiently redistributing surplus food to beneficiaries, such as low-income individuals and charitable organizations.
- **Promote Sustainability**: Our goal is to contribute to environmental sustainability by minimizing the environmental impact associated with food waste, including reduced greenhouse gas emissions.
- **Develop the Mobile Application**: Design and create a user-friendly Android mobile application that enables users to easily donate excess food and allows agents to efficiently collect and redistribute it.

6

LITERATURE SURVEY

Title	Authors	Publication	Key Findings
An Empirical Analysis of the Factors Influencing Food Waste in Harbin under the Heading One-person Food and Multi-person Food with Structural Equation Modeling.	Smith, J. D., & Wang, L.	Food Research International	Factors influencing food waste in Harbin, categorized as one-person food and multi-person food, were empirically analysed using SEM.
Sustainable Food Waste Management and Tracking System Using Blockchain.	Gupta, A., & Patel, S.	Sustainability	Development of a sustainable food waste management and tracking system using block chain technology.
Food Wastage Management Application using Android Studio.	Smith, E	Unpublished application project	Details about an unpublished food wastage management application developed using Android Studio.
Design of Intelligent Household Food Waste Product Based on AHP-TRIZ Method	Wang, S. & Xu, J.	International Conference on CoST	Design of an intelligent household food waste product using the AHP-TRIZ method
Food Waste Management Android App	Bhardwaj, S., Kumar, U., & Kumar	Proceedings of Advancement in Electronics & Communication Engineering	Description of a food waste management Android app without specific methodological details.
A Web and Mobile Based Approach to Redistribute Consumable Food Waste	Prova, R. R., Rayhan, A., Shilon, R. S., & Khan, M. M.	ICCCNT	Development of a web and mobile-based approach to redistribute consumable food waste.
Air Quality Prediction (IoT) Using Machine Learning	Sardar Maran, P., Reddy, B. S., & Saiharshavardhan, C.	Lecture Notes in Electrical Engineering	Mention of an IoT-based air quality prediction system, not directly related to food waste.
Development of Leftover food management system using efficient hunger search techniques	N. K. G N., J. R., S. S. Nukala, M. K. V., S. P. Shankar, & S. Kandarp	IEEE Mysore Sub Section International Conference	Development of a leftover food management system using efficient hunger search techniques.

Sustainable Value CoCreation at the Bottom of the Pyramid: Using Mobile Applications to Reduce Food Waste and Improve Food Security.	Apostolidis, C., Brown, D., Wijetunga, D., & Katheiarachchi	Journal of Marketing Management	Use of mobile applications to reduce food waste and improve food security, particularly targeting the bottom of the pyramid.
Availability of food for NGO through Mobile Application	Vidhi Panchal, Kajal Kuchekar, & Snehal Tambe	International Research Journal of Engineering and Technology (IRJET)	Availability of food for NGOs through a mobile application.
Application of block chain and internet of things to ensure tamper-proof data availability for Food Safety	Iftekhar, A., Cui, X., Hassan, M., & Afzal, W.	Journal of Food Quality	Application of block chain and IoT for ensuring tamper-proof data availability for food safety.
Waste Food Management And Donation App	FAO, The Food and Agriculture Organization	Online at: FAO Food Loss and Waste Data	Provides a comprehensive overview of food loss and waste data.

9

PROBLEM AND GAPS FOUND IN EXISTING SYSTEM

- Advanced Analytics: Limited capabilities for advanced data analysis and machine learning integration.
- Image Recognition: Inadequate use of image recognition technologies for identifying and categorizing food waste.
- Comprehensive Reporting: Lack of comprehensive and customizable reporting tools for users to gain insights into their waste patterns.
- User-Friendly Interfaces: Need for more intuitive and user-friendly interfaces to encourage regular use.
- Security and Privacy: Insufficient security measures to protect user data and ensure privacy.

10

Advantages:

- User-Friendly Interface: Designed to be intuitive, encouraging regular use and engagement.
- Security: Implements robust security measures to protect user data and ensure privacy.
- Use of Google API: Google Cloud Vision API ensures accurate identification and tracking of food items.
- Social Sharing and Community Engagement: The app can allow users to share their progress and tips with others, fostering a sense of community and encouraging sustainable food practices.

11

APPLICATIONS:

- Tracking Daily Waste: Enable families to track the amount and type of food waste generated daily.
- Meal Planning: Assist in meal planning and portion control to minimize leftovers.
- Awareness: Educate users on the environmental impact of food waste and provide tips on reducing waste.

13

H/W AND S/W REQUIREMENTS:

- **Software Requirements:**

Software Front End: Developed using Android Studio with XML for UI design Software

Back End: sqlite and firebase, Java

CNN Models: Developed with TensorFlow for image recognition.

APIs:Google API key for location tracking

- **Hardware Requirements:**

Hard Disk: Minimum 500GB storage

RAM: Minimum 4GB

Client Device:

Android Mobile: Android version 5.5 (Lollipop) or higher

CONCLUSION

- In conclusion, our food waste tracking application represents a pivotal step in combating global food waste.
- By utilizing XML, Java, CNNs, and Google APIs, we've crafted a powerful tool for waste reduction.
- From households to businesses and beyond, our application empowers users to track, analyze, and minimize food waste.
- Its intuitive interface and real-time analytics make it a valuable asset in sustainability efforts.
- Through strategic partnerships and widespread adoption, we can amplify its impact and create a more sustainable future.
- Let's embrace this technology to drive positive environmental and economic change, one byte at a time.

15

REFRANCES:

- [1]Smith, J. D., & Wang, L. (2023). [1]Factors Influencing Food Waste in Harbin: A Structural Equation Modeling Approach. Food Research International, 45(3), 123-137. doi:10.1234/abcd123456.
- [2] Gupta, A., & Patel, S. (2022). [2]Blockchain-Based Sustainable Food Waste Management and Tracking System. Sustainability, 9(4), 123-136. doi:10.1234/abcd789012.
- [3] Smith, E. (2021). [3]Development of a Food Wastage Management Application using Android Studio. Unpublished application project.
- [4] Wang, S., & Xu, J. (2022). [4]Design of Intelligent Household Food Waste Product Based on AHP-TRIZ Method. In Proceedings of the 2022 International Conference on Culture-Oriented Science and Technology (CoST) (pp. 95-98).
- [5] Bhardwaj, S., Kumar, U., & Kumar, Y. (2022). [5]Food Waste Management Android App. In Proceedings of the Advancement in Electronics & Communication Engineering.
- [6] Prova, R. R., Rayhan, A., Shilon, R. S., & Khan, M. M. (2021). [6]Web and Mobile Based Approach to Redistribute Consumable Food Waste. In Proceedings of the 12th International Conference on Computing Communication and Networking Technologies (ICCCNT) (pp. 1-6).

16

Thank You

17

ANNEXURE G

PROJECT ACHEIVEMENTS

Sr. No	Details	Description
1	Publication Certificate	We got the certificate of publication from "Journal of Multimedia Technology & Recent advancements"
2	Shodh Project Compition	Got the certificate from Project project competion which is located in Nagpur.
3	Conference participation certificate (Sixth Internamtional Conference on Recent trends , Engineering and management)	Got the certificate
4	Certificate of Metaverse	Introduction and interdisciplinary application
5	Internship Certificate	We have done the internship in Web development, Rebotics,python etc.
6	Udemy Courses certificate	Based upon deep learning, machine learning, Java, Python etc.
7	Certificate of Java learning 11	Intrduction of advanced java
8	Certificate of basic pyhton course	Learn about the basics of python.
9	Certificate of MongoDB	One day lecture series on MongoDB.

ANNEXURE H

Documentation on Step by Step Execution of a Project

Step 1: Define the Requirements and Features

1. **Identify the Purpose:**
    - Track food waste in households or restaurants.
    - Provide insights and suggestions to reduce waste.
2. **List the Key Features:**
    - User registration and login.
    - Logging food purchases and waste.
    - Analytics and reports on food waste.
    - Notifications and reminders.
    - Tips and suggestions for reducing waste.
    - Option to donate or share surplus food.

Step 2: Design the Application

1. **Wireframing:**
    - Create wireframes for each screen (registration, logging, dashboard, analytics).
    - Use tools like Figma, Sketch, or Adobe XD.
2. **UI/UX Design:**
    - Design a user-friendly interface.
    - Ensure the design is intuitive and easy to navigate.

Step 3: Set Up the Development Environment

1. **Install Android Studio:**
    - Download and install Android Studio from the official website.
2. **Configure SDK:**
    - Ensure you have the latest Android SDK and necessary tools installed.
3. **Create a New Project:**
    - Start a new Android Studio project.
    - Choose a project template (e.g., Empty Activity).

Step 4: Develop the Backend

1. **Choose a Backend Technology:**
    - Options include Firebase, Node.js, Django, or any other backend service.
2. **Set Up a Database:**
    - Design the database schema to store user data, food logs, and analytics.

- Use Firebase Firestore or Realtime Database for simplicity if using Firebase.

3. **API Development:**
    - Create RESTful APIs to interact with the database.
    - Implement user authentication (Firebase Auth can be used for ease).

Step 5: Develop the Frontend

1. **User Authentication:**
    - Implement user registration, login, and logout functionality using Firebase Auth or your custom backend.
2. **Food Logging:**
    - Create forms for users to log food purchases and waste.
    - Store the data in the database.
3. **Dashboard:**
    - Develop a dashboard to show analytics and insights on food waste.
    - Use charts and graphs to display data (e.g., MPAndroidChart library).
4. **Notifications:**
    - Implement push notifications to remind users to log their food.
    - Use Firebase Cloud Messaging (FCM) for notifications.
5. **Additional Features:**
    - Implement tips and suggestions for reducing waste.
    - Provide options for users to donate or share surplus food.

Step 6: Testing

1. **Unit Testing:**
    - Write unit tests for individual components and functions.
    - Use JUnit and Mockito for Android unit testing.
2. **Integration Testing:**
    - Test the integration between frontend and backend.
    - Ensure data is correctly stored and retrieved.
3. **User Testing:**
    - Conduct beta testing with a group of users.
    - Gather feedback and make necessary improvements.

Step 7: Deployment

1. **Prepare for Release:**
    - Ensure all features are complete and tested.
    - Optimize the app for performance and security.
2. **Publish to Google Play Store:**
    - Create a Google Play Developer account.
    - Generate a signed APK or App Bundle.
    - Create a listing on the Play Store with app details, screenshots, and a promotional video.
    - Submit the app for review.

Step 8: Post-Deployment

1. **Monitor Performance:**
    - Use tools like Firebase Crashlytics to monitor app crashes and issues.
    - Gather user feedback and reviews.
2. **Update and Maintain:**
    - Regularly update the app with new features and improvements.
    - Fix bugs and ensure compatibility with new Android versions.

Tools and Libraries

- **Firebase:** Authentication, Database, Cloud Messaging.
- **Retrofit/Volley:** For API calls.
- **MPAndroidChart:** For charts and graphs.
- **JUnit/Mockito:** For testing.
- **Crashlytics:** For monitoring app performance.

ANNEXURE I

SYSTEM CODE

Donar's Code:

```
package com.sathyaseelan.foodhunger;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.ap.GoogleApiClient;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FieldValue;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.GeoPoint;

import java.util.HashMap;
import java.util.Map;

public class Donate extends AppCompatActivity implements OnMapReadyCallback,
GoogleApiClient.ConnectionCallbacks, GoogleApiClient.OnConnectionFailedListener,
com.google.android.gms.location.LocationListener {

    private GoogleMap mMap;
    GoogleApiClient mGoogleApiClient;
    Location mLastLocation;
    LocationRequest mLocationRequest;
    private int REQUEST_CODE = 11;
```

```
    SupportMapFragment mapFragment;
    EditText mFullName,mFoodItem,mDescription,mPhone;
    Button mSubmitBtn;
    FirebaseAuth fAuth;
    FirebaseFirestore fStore;
    String userID;
    public static final String TAG = "TAG";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_donate);
        mFullName = findViewById(R.id.donorname);
        mFoodItem = findViewById(R.id.fooditem);
        mPhone = findViewById(R.id.phone);
        mDescription = findViewById(R.id.description);
        mSubmitBtn=findViewById(R.id.submit);

        fAuth=FirebaseAuth.getInstance();
        fStore= FirebaseFirestore.getInstance();

        mapFragment = (SupportMapFragment)
getSupportFragmentManager().findFragmentById(R.id.google_map);
        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED) {
            mapFragment.getMapAsync(this);
        } else {
            ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_CODE);
        }
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;
        buildGoogleApiClient();
        if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
            return;
        }
        mMap.setMyLocationEnabled(true);
    }

    protected synchronized void buildGoogleApiClient(){
        mGoogleApiClient = new GoogleApiClient.Builder(this)
                .addConnectionCallbacks(this)
```

```
            .addOnConnectionFailedListener(this)
            .addApi(LocationServices.API)
            .build();
        mGoogleApiClient.connect();
    }

    @Override
    public void onLocationChanged(@NonNull Location location) {
        mLastLocation = location;
        LatLng latLng = new LatLng(location.getLatitude(),location.getLongitude());

        //MarkerOptions markerOptions1 = new MarkerOptions().position(latLng).title("You
are here");
        //mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
        //mMap.animateCamera(CameraUpdateFactory.zoomTo(15));
        //mMap.addMarker(markerOptions1).showInfoWindow();

        MarkerOptions markerOptions = new MarkerOptions().position(latLng).title("You are
here");
        mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng,15));
        mMap.addMarker(markerOptions).showInfoWindow();


        mSubmitBtn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String fullname = mFullName.getText().toString().trim();
                String fooditem= mFoodItem.getText().toString().trim();
                String description= mDescription.getText().toString().trim();
                String phone= mPhone.getText().toString().trim();
                String type= "Donor";

                if(TextUtils.isEmpty(fullname))
                {
                    mFullName.setError("Name is Required.");
                    return;
                }

                if(TextUtils.isEmpty(fooditem))
                {
                    mFoodItem.setError("Required.");
                    return;
                }

                if(phone.length() < 10)
                {
                    mPhone.setError("Phone Number Must be >=10 Characters");
                    return;
                }

                userID = fAuth.getCurrentUser().getUid();
```

```
            //DocumentReference documentReference =
fStore.collection("donate").document(userID);
            CollectionReference collectionReference = fStore.collection("user data");

            GeoPoint geoPoint = new
GeoPoint(location.getLatitude(),location.getLongitude());
            Map<String,Object> user = new HashMap<>();
            user.put("timestamp", FieldValue.serverTimestamp());
            user.put("name",fullname);
            user.put("food item",fooditem);
            user.put("phone",phone);
            user.put("description",description);
            user.put("location",geoPoint);
            user.put("userid",userID);
            user.put("type",type);

            collectionReference.add(user)
                  .addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
                     @Override
                     public void onSuccess(DocumentReference documentReference) {

Toast.makeText(getApplicationContext(),"Success!",Toast.LENGTH_SHORT).show();
                        Log.d(TAG,"Success!");
                        //startActivity(new Intent(getApplicationContext(),MainActivity.class));
                        Intent intent = new Intent(Donate.this, MainActivity.class);
                        intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
Intent.FLAG_ACTIVITY_CLEAR_TASK | Intent.FLAG_ACTIVITY_NEW_TASK);
                        startActivity(intent);
                     }
                  })
                  .addOnFailureListener(new OnFailureListener() {
                     @Override
                     public void onFailure(@NonNull Exception e) {

Toast.makeText(getApplicationContext(),"Error!",Toast.LENGTH_SHORT).show();
                        Log.w(TAG, "Error!", e);
                     }
                  });
         }
      });
   }

   @Override
   public void onConnected(@Nullable Bundle bundle) {
      mLocationRequest = new LocationRequest();
      mLocationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
      if (ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(this,
Manifest.permission.ACCESS_COARSE_LOCATION) !=
```

```
        PackageManager.PERMISSION_GRANTED) {
                return;
            }
            LocationServices.FusedLocationApi.requestLocationUpdates(mGoogleApiClient,
        mLocationRequest, this);

        }

        @Override
        public void onConnectionSuspended(int i) {

        }

        @Override
        public void onConnectionFailed(@NonNull ConnectionResult connectionResult) {

        }
        @Override
        public void onRequestPermissionsResult(int requestCode, @NonNull String[]
        permissions, @NonNull int[] grantResults) {

            if (requestCode == REQUEST_CODE){
                if(grantResults.length > 0 && grantResults[0]  ==
        PackageManager.PERMISSION_GRANTED){
                    mapFragment.getMapAsync(this);
                }else{
                    Toast.makeText(this,"Permission Denied", Toast.LENGTH_SHORT).show();
                }
            }
        }
    }
```

Receiver's Code:

```
package com.sathyaseelan.foodhunger;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
```

```
import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.OnSuccessListener;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.DocumentReference;
import com.google.firebase.firestore.FieldValue;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.GeoPoint;

import java.util.HashMap;
import java.util.Map;

public class Receive extends AppCompatActivity implements OnMapReadyCallback,
        GoogleApiClient.ConnectionCallbacks, GoogleApiClient.OnConnectionFailedListener,
        com.google.android.gms.location.LocationListener {

    private GoogleMap mMap;
    GoogleApiClient mGoogleApiClient;
    Location mLastLocation;
    LocationRequest mLocationRequest;
    private int REQUEST_CODE = 11;
    SupportMapFragment mapFragment;
    EditText mFullName,mDescription;
    Button mSubmitBtn,b1;
    FirebaseAuth fAuth;
    FirebaseFirestore fStore;
    String userID;
    public static final String TAG = "TAG";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_receive);
        mFullName = findViewById(R.id.receivername);
        mDescription = findViewById(R.id.description);
        mSubmitBtn=findViewById(R.id.submit);
        b1=findViewById(R.id.checkfood);

        fAuth=FirebaseAuth.getInstance();
        fStore= FirebaseFirestore.getInstance();
```

```
    mapFragment = (SupportMapFragment)
       getSupportFragmentManager().findFragmentById(R.id.google_map);
    if (ActivityCompat.checkSelfPermission(this,
       Manifest.permission.ACCESS_FINE_LOCATION) ==
       PackageManager.PERMISSION_GRANTED) {
       mapFragment.getMapAsync(this);
    } else {
       ActivityCompat.requestPermissions(this, new
       String[]{Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_CODE);
    }


}

@Override
public void onMapReady(GoogleMap googleMap) {
    mMap = googleMap;
    buildGoogleApiClient();
    if (ActivityCompat.checkSelfPermission(this,
       Manifest.permission.ACCESS_FINE_LOCATION) !=
       PackageManager.PERMISSION_GRANTED &&
       ActivityCompat.checkSelfPermission(this,
       Manifest.permission.ACCESS_COARSE_LOCATION) !=
       PackageManager.PERMISSION_GRANTED) {
       return;
    }
    mMap.setMyLocationEnabled(true);
}

protected synchronized void buildGoogleApiClient(){
    mGoogleApiClient = new GoogleApiClient.Builder(this)
            .addConnectionCallbacks(this)
            .addOnConnectionFailedListener(this)
            .addApi(LocationServices.API)
            .build();
    mGoogleApiClient.connect();
}

@Override
public void onLocationChanged(@NonNull Location location) {
    mLastLocation = location;
    LatLng latLng = new LatLng(location.getLatitude(),location.getLongitude());

    //MarkerOptions markerOptions1 = new MarkerOptions().position(latLng).title("You are
       here");
    //mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
    //mMap.animateCamera(CameraUpdateFactory.zoomTo(15));
    //mMap.addMarker(markerOptions1).showInfoWindow();

    MarkerOptions markerOptions = new MarkerOptions().position(latLng).title("You are here");
```

```
mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng,15));
mMap.addMarker(markerOptions).showInfoWindow();

b1.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View view) {
      Intent i=new Intent(Receive.this,Imageprediction.class);
      startActivity(i);
   }
});

mSubmitBtn.setOnClickListener(new View.OnClickListener() {
   @Override
   public void onClick(View v) {
      String fullname = mFullName.getText().toString().trim();
      String description= mDescription.getText().toString().trim();
      String type= "Receiver";

      if(TextUtils.isEmpty(fullname))
      {
         mFullName.setError("Name is Required.");
         return;
      }
      if(TextUtils.isEmpty(description))
      {
         mFullName.setError("Description is Required.");
         return;
      }


      userID = fAuth.getCurrentUser().getUid();
      //DocumentReference documentReference =
   fStore.collection("receiver").document(userID);
      CollectionReference collectionReference = fStore.collection("user data");

      GeoPoint geoPoint = new GeoPoint(location.getLatitude(),location.getLongitude());
      Map<String,Object> user = new HashMap<>();
      user.put("timestamp", FieldValue.serverTimestamp());
      user.put("name",fullname);
      user.put("description",description);
      user.put("location",geoPoint);
      user.put("userid",userID);
      user.put("type",type);

      collectionReference.add(user)
            .addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
               @Override
               public void onSuccess(DocumentReference documentReference) {

   Toast.makeText(getApplicationContext(),"Success!",Toast.LENGTH_SHORT).show();
                  Log.d(TAG,"Success!");
```

```
                    //startActivity(new Intent(getApplicationContext(),MainActivity.class));
                    Intent intent = new Intent(Receive.this, MainActivity.class);
                    intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
        Intent.FLAG_ACTIVITY_CLEAR_TASK | Intent.FLAG_ACTIVITY_NEW_TASK);
                    startActivity(intent);
                }
            })
            .addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {

        Toast.makeText(getApplicationContext(),"Error!",Toast.LENGTH_SHORT).show();
                    Log.w(TAG, "Error!", e);
                }
            });

        }
    });
}

@Override
public void onConnected(@Nullable Bundle bundle) {
    mLocationRequest = new LocationRequest();
    //mLocationRequest.setInterval(1000);
    //mLocationRequest.setFastestInterval(1000);
    mLocationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
    if (ActivityCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_FINE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED &&
        ActivityCompat.checkSelfPermission(this,
        Manifest.permission.ACCESS_COARSE_LOCATION) !=
        PackageManager.PERMISSION_GRANTED) {
        return;
    }
    LocationServices.FusedLocationApi.requestLocationUpdates(mGoogleApiClient,
        mLocationRequest, this);

}

@Override
public void onConnectionSuspended(int i) {

}

@Override
public void onConnectionFailed(@NonNull ConnectionResult connectionResult) {

}
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
        @NonNull int[] grantResults) {
```

```
        if (requestCode == REQUEST_CODE){
            if(grantResults.length > 0 && grantResults[0]  ==
            PackageManager.PERMISSION_GRANTED){
                mapFragment.getMapAsync(this);
            }else{
                Toast.makeText(this,"Permission Denied", Toast.LENGTH_SHORT).show();
            }
        }

            }
```

Contact Us Code:

```
    package com.sathyaseelan.foodhunger;

    import androidx.annotation.NonNull;
    import androidx.appcompat.app.AppCompatActivity;

    import android.content.Intent;
    import android.os.Bundle;
    import android.util.Log;
    import android.util.Patterns;
    import android.view.View;
    import android.widget.Button;
    import android.widget.EditText;
    import android.widget.Toast;

    import com.google.android.gms.tasks.OnFailureListener;
    import com.google.android.gms.tasks.OnSuccessListener;
    import com.google.android.material.textfield.TextInputLayout;
    import com.google.firebase.auth.FirebaseAuth;
    import com.google.firebase.firestore.CollectionReference;
    import com.google.firebase.firestore.DocumentReference;
    import com.google.firebase.firestore.FieldValue;
    import com.google.firebase.firestore.FirebaseFirestore;

    import java.util.HashMap;
    import java.util.Map;

    public class Contact extends AppCompatActivity {
        EditText name, email, message;
        Button submit;
        boolean isNameValid, isEmailValid, isMessageValid;
        FirebaseAuth fAuth;
        FirebaseFirestore fStore;
        String userID;
        public static final String TAG = "TAG";
        TextInputLayout nameError, emailError, messageError;
        @Override
        protected void onCreate(Bundle savedInstanceState) {
```

```
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_contact);
    name = (EditText) findViewById(R.id.name);
    email = (EditText) findViewById(R.id.email);
    message = (EditText) findViewById(R.id.message);
    submit = (Button) findViewById(R.id.submit);
    nameError = (TextInputLayout) findViewById(R.id.nameError);
    emailError = (TextInputLayout) findViewById(R.id.emailError);
    messageError = (TextInputLayout) findViewById(R.id.messageError);

    fAuth=FirebaseAuth.getInstance();
    fStore= FirebaseFirestore.getInstance();

    submit.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            SetValidation();
        }
    });
}
    public void SetValidation() {

        // Check for a valid name.
        if (name.getText().toString().isEmpty()) {
            nameError.setError(getResources().getString(R.string.name_error));
            isNameValid = false;
        } else  {
            isNameValid = true;
            nameError.setErrorEnabled(false);
        }

        // Check for a valid email address.
        if (email.getText().toString().isEmpty()) {
            emailError.setError(getResources().getString(R.string.email_error));
            isEmailValid = false;
        } else if (!Patterns.EMAIL_ADDRESS.matcher(email.getText().toString()).matches()) {
            emailError.setError(getResources().getString(R.string.error_invalid_email));
            isEmailValid = false;
        } else  {
            isEmailValid = true;
            emailError.setErrorEnabled(false);
        }

        // Check for a valid phone number.
        if (message.getText().toString().isEmpty()) {
            messageError.setError(getResources().getString(R.string.phone_error));
            isMessageValid = false;
        } else  {
            isMessageValid = true;
            messageError.setErrorEnabled(false);
        }
```

```
        if (isNameValid && isEmailValid && isMessageValid ) {

            String Name = name.getText().toString().trim();
            String Email= email.getText().toString().trim();
            String Message= message.getText().toString().trim();
            userID = fAuth.getCurrentUser().getUid();
            //DocumentReference documentReference =
        fStore.collection("donate").document(userID);
            CollectionReference collectionReference = fStore.collection("contact data");

            Map<String,Object> user = new HashMap<>();
            user.put("timestamp", FieldValue.serverTimestamp());
            user.put("name",Name);
            user.put("email",Email);
            user.put("message",Message);
            user.put("userid",userID);

            collectionReference.add(user)
                    .addOnSuccessListener(new OnSuccessListener<DocumentReference>() {
                        @Override
                        public void onSuccess(DocumentReference documentReference) {

        Toast.makeText(getApplicationContext(),"Success!",Toast.LENGTH_SHORT).show();
                            Log.d(TAG,"Successfully! We will shortly revert you back.");
                            //startActivity(new Intent(getApplicationContext(),MainActivity.class));
                            Intent intent = new Intent(Contact.this, MainActivity.class);
                            intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
        Intent.FLAG_ACTIVITY_CLEAR_TASK | Intent.FLAG_ACTIVITY_NEW_TASK);
                            startActivity(intent);
                        }
                    })
                    .addOnFailureListener(new OnFailureListener() {
                        @Override
                        public void onFailure(@NonNull Exception e) {

        Toast.makeText(getApplicationContext(),"Error!",Toast.LENGTH_SHORT).show();
                            Log.w(TAG, "Error!", e);
                        }
                    });
        }

    }
        }
```

About Us Code:

```
package com.sathyaseelan.foodhunger;

import androidx.appcompat.app.AppCompatActivity;
```

```
import androidx.cardview.widget.CardView;

import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;


public class About extends AppCompatActivity {

    CardView instagram,facebook,twitter;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_about);

        instagram = findViewById(R.id.instagram);
        facebook = findViewById(R.id.facebook);
        twitter = findViewById(R.id.twitter);

        instagram.setOnClickListener(new View.OnClickListener ()
        {
            @Override
            public void onClick(View v) {
                Intent myWebLink = new Intent(android.content.Intent.ACTION_VIEW);
                myWebLink.setData(Uri.parse("http://www.instagram.com"));
                startActivity(myWebLink);
            }
        });
        facebook.setOnClickListener(new View.OnClickListener ()
        {
            @Override
            public void onClick(View v) {
                Intent myWebLink = new Intent(android.content.Intent.ACTION_VIEW);
                myWebLink.setData(Uri.parse("http://www.facebook.com"));
                startActivity(myWebLink);
            }
        });
        twitter.setOnClickListener(new View.OnClickListener ()
        {
            @Override
            public void onClick(View v) {
                Intent myWebLink = new Intent(android.content.Intent.ACTION_VIEW);
                myWebLink.setData(Uri.parse("http://www.twitter.com"));
                startActivity(myWebLink);
            }
        });
    }}
```

History Code:

```
package com.sathyaseelan.foodhunger;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.util.Log;
import android.widget.TextView;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.Timestamp;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.CollectionReference;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;

public class History extends AppCompatActivity {

    private FirebaseFirestore db = FirebaseFirestore.getInstance();
    private CollectionReference notebookref = db.collection("user data");
    public static final String TAG = "TAG";
    private TextView textViewData;
    FirebaseAuth fAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_history);
        fAuth= FirebaseAuth.getInstance();
        textViewData=findViewById(R.id.data);

        loadNotes();
    }

    public void loadNotes() {
        notebookref.get()
                .addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
                    @Override
                    public void onComplete(@NonNull Task<QuerySnapshot> task) {
                        if (task.isSuccessful()) {
                            String data="";
                            for (QueryDocumentSnapshot document : task.getResult()) {
                                Log.d(TAG, document.getId() + " => " + document.getData());//
                                if (document.contains("name") && document.contains("description") &&
     document.contains("user type") && document.contains("userid")) {

                                    String name = (String) document.get("name");
```

```
                        String type = (String) document.get("user type");
                        String description = (String) document.get("description");
                        String Userid = (String) document.get("userid");
                        String userID = fAuth.getCurrentUser().getUid();
                        Timestamp ts = (Timestamp) document.get("timestamp");
                        //String dateandtime=String.valueOf(ts);
                        String dateandtime=String.valueOf(ts.toDate());
                        //String dateandtime = ts.toString();

                        if(Userid.equals(userID)) {
                            data += "Name: " + name + "\nUser Type: " + type + "\nDescription: " +
        description + "\nDate & Time: " + dateandtime + "\n\n";
                            //data += "Name: " + name + "\nUser Type: " + type + "\nDescription: " +
        description + "\n";
                        }
                        textViewData.setText(data);
                    }
                }
                //textViewData.setText(data);
            } else {
                Log.d(TAG, "Error fetching data: ", task.getException());
            }
        }
    });
  }
}
```

My Pin Code:

```
package com.sathyaseelan.foodhunger;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;

import android.Manifest;
import android.content.pm.PackageManager;
import android.location.Location;
import android.os.Bundle;
import android.util.Log;
import android.widget.Toast;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
```

```
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.firestore.FirebaseFirestore;
import com.google.firebase.firestore.GeoPoint;
import com.google.firebase.firestore.QueryDocumentSnapshot;
import com.google.firebase.firestore.QuerySnapshot;

public class MyPin extends AppCompatActivity implements OnMapReadyCallback,
        GoogleApiClient.ConnectionCallbacks, GoogleApiClient.OnConnectionFailedListener,
        com.google.android.gms.location.LocationListener {

    private GoogleMap mMap;
    GoogleApiClient mGoogleApiClient;
    Location mLastLocation;
    LocationRequest mLocationRequest;
    SupportMapFragment mapFragment;
    private int REQUEST_CODE = 11;
    FirebaseFirestore fStore;
    FirebaseAuth fAuth;
    public static final String TAG = "TAG";
    private FirebaseFirestore cloudstorage;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_my_pin);
        fAuth= FirebaseAuth.getInstance();

        mapFragment = (SupportMapFragment)
         getSupportFragmentManager().findFragmentById(R.id.google_map);
        if (ActivityCompat.checkSelfPermission(this,
         Manifest.permission.ACCESS_FINE_LOCATION) ==
         PackageManager.PERMISSION_GRANTED) {
            mapFragment.getMapAsync(this);
        } else {
            ActivityCompat.requestPermissions(this, new
         String[]{Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_CODE);
        }

    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;
        if (ActivityCompat.checkSelfPermission(this,
         Manifest.permission.ACCESS_FINE_LOCATION) !=
```

```
     PackageManager.PERMISSION_GRANTED &&
     ActivityCompat.checkSelfPermission(this,
     Manifest.permission.ACCESS_COARSE_LOCATION) !=
     PackageManager.PERMISSION_GRANTED) {
       return;
    }
    buildGoogleApiClient();
    mMap.setMyLocationEnabled(true);
}

protected synchronized void buildGoogleApiClient(){
    mGoogleApiClient = new GoogleApiClient.Builder(this)
            .addConnectionCallbacks(this)
            .addOnConnectionFailedListener(this)
            .addApi(LocationServices.API)
            .build();
    mGoogleApiClient.connect();
}

@Override
public void onLocationChanged(@NonNull Location location) {
    mLastLocation = location;
    showLocation();
    LatLng latLng = new LatLng(location.getLatitude(),location.getLongitude());

    MarkerOptions markerOptions1 = new MarkerOptions().position(latLng).title("You are
     here").icon(BitmapDescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_RED));
    //mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
    //mMap.animateCamera(CameraUpdateFactory.zoomTo(15));
    //mMap.addMarker(markerOptions1).showInfoWindow();
    mMap.animateCamera(CameraUpdateFactory.newLatLngZoom(latLng,15));
    mMap.addMarker(markerOptions1).showInfoWindow();
}

public void showLocation() {
    this.cloudstorage = FirebaseFirestore.getInstance();
    cloudstorage.collection("user data")
            .get()
            .addOnCompleteListener(new OnCompleteListener<QuerySnapshot>() {
                @Override
                public void onComplete(@NonNull Task<QuerySnapshot> task) {
                    if (task.isSuccessful()) {
                        for (QueryDocumentSnapshot document : task.getResult()) {
                            Log.d(TAG, document.getId() + " => " + document.getData());//
                            if (document.contains("location") && document.contains("name") &&
     document.contains("description") && document.contains("userid")) {
                                GeoPoint location = (GeoPoint) document.get("location");
                                String title = (String) document.get("name");
                                String type = (String) document.get("type");
                                String description = (String) document.get("description");
                                String Userid = (String) document.get("userid");
```

```
                        String userID = fAuth.getCurrentUser().getUid();

                        if(type.equals("Donor") & Userid.equals(userID)) {
                            Log.d(TAG, userID + " Success " + title);
                            LatLng latLng = new LatLng(location.getLatitude(),
    location.getLongitude());
                            //mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
                            mMap.addMarker(new
    MarkerOptions().position(latLng).title(title+"("+type+")").snippet(description).icon(Bitmap
    DescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_GREEN)));
                        }
                        else if(type.equals("Receiver") & Userid.equals(userID)){
                            Log.d(TAG, String.valueOf(location) + " Success " + title);
                            LatLng latLng = new LatLng(location.getLatitude(),
    location.getLongitude());
                            //mMap.moveCamera(CameraUpdateFactory.newLatLng(latLng));
                            mMap.addMarker(new
    MarkerOptions().position(latLng).title(title+"("+type+")").snippet(description).icon(Bitmap
    DescriptorFactory.defaultMarker(BitmapDescriptorFactory.HUE_BLUE)));
                        }
                    }
                }
            } else {
                Log.d(TAG, "Error fetching data: ", task.getException());
            }
        }
    });

}

@Override
public void onConnected(@Nullable Bundle bundle) {
   mLocationRequest = new LocationRequest();
   mLocationRequest.setPriority(LocationRequest.PRIORITY_HIGH_ACCURACY);
   if (ActivityCompat.checkSelfPermission(this,
    Manifest.permission.ACCESS_FINE_LOCATION) !=
    PackageManager.PERMISSION_GRANTED &&
    ActivityCompat.checkSelfPermission(this,
    Manifest.permission.ACCESS_COARSE_LOCATION) !=
    PackageManager.PERMISSION_GRANTED) {
      return;
   }
   LocationServices.FusedLocationApi.requestLocationUpdates(mGoogleApiClient,
    mLocationRequest, this);

}

@Override
public void onConnectionSuspended(int i) {

}
```

```
    @Override
    public void onConnectionFailed(@NonNull ConnectionResult connectionResult) {

    }

    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull String[] permissions,
        @NonNull int[] grantResults) {
       if (requestCode == REQUEST_CODE){
          if(grantResults.length > 0 && grantResults[0]  ==
        PackageManager.PERMISSION_GRANTED){
             mapFragment.getMapAsync(this);
          }else{
             Toast.makeText(this,"Permission Denied", Toast.LENGTH_SHORT).show();
          }
       }
    }
}
```

Landing Page Code:

```
package com.sathyaseelan.foodhunger;

import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;

import com.google.firebase.auth.FirebaseAuth;

public class landingpage extends AppCompatActivity {

    CardView login,register,about;
    FirebaseAuth fAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_landingpage);

        login = findViewById(R.id.cardLogin);
        register = findViewById(R.id.cardRegister);
        about = findViewById(R.id.cardAboutus);

        fAuth= FirebaseAuth.getInstance();
        if(fAuth.getCurrentUser() !=null){
            Intent intent = new Intent(landingpage.this, MainActivity.class);
```

```
            intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP |
          Intent.FLAG_ACTIVITY_CLEAR_TASK | Intent.FLAG_ACTIVITY_NEW_TASK);
            startActivity(intent);
        }

        login.setOnClickListener(new View.OnClickListener ()
        {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(getApplicationContext(), Logup.class));
            }
        });
        register.setOnClickListener(new View.OnClickListener ()
        {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(getApplicationContext(), Signup.class));
            }
        });
        about.setOnClickListener(new View.OnClickListener ()
        {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(getApplicationContext(), About.class));
            }
        });
    }

}
```

Image Prediction Code:

```
  import androidx.annotation.Nullable;
  import androidx.appcompat.app.AppCompatActivity;

  import android.Manifest;
  import android.app.ProgressDialog;
  import android.content.Intent;
  import android.content.pm.PackageManager;
  import android.graphics.Bitmap;
  import android.media.ThumbnailUtils;
  import android.net.Uri;
  import android.os.Build;
  import android.os.Bundle;
  import android.os.Handler;
  import android.provider.MediaStore;
  import android.view.View;
  import android.widget.Button;
  import android.widget.ImageView;
  import android.widget.TextView;
```

```
import com.sathyaseelan.foodhunger.ml.Model;

import org.tensorflow.lite.DataType;
import org.tensorflow.lite.support.tensorbuffer.TensorBuffer;

import java.io.IOException;
import java.nio.ByteBuffer;
import java.nio.ByteOrder;

public class Imageprediction extends AppCompatActivity {
    TextView result, confidence;
    ImageView imageView;
    Button picture, gal;
    int imageSize = 224;
    int maxPos = 0;
    private ProgressDialog progress;
    static final int REQUEST_IMAGE_CAPTURE = 1;
    ProgressDialog progressBar;
    private int progressBarStatus = 0;
    private Handler progressBarHandler = new Handler();
    private long fileSize = 0;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_imageprediction);

        result = findViewById(R.id.result);
        confidence = findViewById(R.id.confidence);
        imageView = findViewById(R.id.imageView);
        picture = findViewById(R.id.takebtn);
        gal = (Button) findViewById(R.id.buttontt);

        picture.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View view) {
                // Launch camera if we have permission
                if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.M) {
                    if (checkSelfPermission(android.Manifest.permission.CAMERA) ==
PackageManager.PERMISSION_DENIED) {

                        Intent cameraIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
                        startActivityForResult(cameraIntent, 1);
                    } else {

                        requestPermissions(new String[]{Manifest.permission.CAMERA}, 100);
                    }
                }
            }
        });
```

```
        gal.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent cameraIntent = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                startActivityForResult(cameraIntent, 3);
            }
        });
    }

    public void classifyImage(Bitmap image) {
        try {
            Model model = Model.newInstance(getApplicationContext());

            // Creates inputs for reference.
            TensorBuffer inputFeature0 = TensorBuffer.createFixedSize(new int[]{1, 224, 224, 3},
DataType.FLOAT32);
            ByteBuffer byteBuffer = ByteBuffer.allocateDirect(4 * imageSize * imageSize * 3);
            byteBuffer.order(ByteOrder.nativeOrder());

            int[] intValue = new int[imageSize * imageSize];
            image.getPixels(intValue, 0, image.getWidth(), 0, 0, image.getWidth(),
image.getHeight());

            int pixel = 0;
            for (int i = 0; i < imageSize; i++) {
                for (int j = 0; j < imageSize; j++) {
                    int val = intValue[pixel++];
                    byteBuffer.putFloat(((val >> 16) & 0xFF) * (1.f / 255.f));
                    byteBuffer.putFloat(((val >> 8) & 0xFF) * (1.f / 255.f));
                    byteBuffer.putFloat((val & 0xFF) * (1.f / 255.f));

                }
            }

            inputFeature0.loadBuffer(byteBuffer);

            Model.Outputs outputs = model.process(inputFeature0);
            TensorBuffer outputFeatures0 = outputs.getOutputFeature0AsTensorBuffer();

            float[] yolo = outputFeatures0.getFloatArray();


            float maxConfidence = 0;
            for (int i = 0; i < yolo.length; i++) {
                if (yolo[i] > maxConfidence) {
                    maxConfidence = yolo[i];
                    maxPos = i;

                }
            }
```

```
        /*  0 Bad Rice
          1 Clean Rice
          2 Good Roti
          3 Invalid Image */

          String[] classes = {"Fresh Biryani", "Spoiled Biryani","Fresh Bread","Spoiled
Bread","Fresh Noodles","Spoiled Noodles","Invalid Image"};
          // result.setText(classes[maxPos]);

          //   result.setText(classes[maxPos]);
          /* final Progress dialog =
                new ProgressDialog(CancerDetection.this);*/
          progressBar = new ProgressDialog(Imageprediction.this);
          progressBar.setCancelable(true);
          progressBar.setMessage("Image Prediction Processing ...");
          progressBar.setProgressStyle(ProgressDialog.STYLE_HORIZONTAL);
          progressBar.setProgress(0);
          progressBar.setMax(100);
          progressBar.show();
          //reset progress bar and filesize status
          progressBarStatus = 0;
          fileSize = 0;

          new Thread(new Runnable() {
             public void run() {

                result.setText(classes[maxPos]);
                while (progressBarStatus < 100) {
                   // performing operation

                   progressBarStatus = doOperation();
                   try {
                      Thread.sleep(1000);
                   } catch (InterruptedException e) {
                      e.printStackTrace();
                   }
                   // Updating the progress bar
                   progressBarHandler.post(new Runnable() {
                      public void run() {
                         progressBar.setProgress(progressBarStatus);
                      }
                   });
                }
                // performing operation if file is downloaded,

                if (progressBarStatus >= 100) {
                   // sleeping for 1 second after operation completed
                   //
                   try {
```

```
                    Thread.sleep(1000);
                } catch (InterruptedException e) {
                    e.printStackTrace();
                }
                // close the progress bar dialog
                progressBar.dismiss();

            }
        }
    }).start();


     /*  result.setOnClickListener(new View.OnClickListener() {
          @Override
          public void onClick(View v) {
             Intent i=new Intent(ImagePrediction.this,DoctorActivity.class);
             startActivity(i);
          }
       });*/
       model.close();

    } catch (IOException e) {

    }
}

private int doOperation() {


    while (fileSize <= 10000) {
       fileSize++;
       if (fileSize == 1000) {
          return 10;
       } else if (fileSize == 2000) {
          return 20;
       } else if (fileSize == 3000) {
          return 30;
       } else if (fileSize == 4000) {
          return 40; // you can add more else if
       }
       else if (fileSize == 5000) {
          return 50; // you can add more else if
       }
     /* else {
          return 100;
       }*/
    }//end of while
    return 100;
}//end of doOperation
```

```
    @Override
    protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
        if(resultCode == RESULT_OK) {
            if (requestCode == 1) {
                Bitmap image = (Bitmap) data.getExtras().get("data");
                int dimension = Math.min(image.getWidth(), image.getHeight());
                image = ThumbnailUtils.extractThumbnail(image, dimension, dimension);

                imageView.setImageBitmap(image);
                // demoTxt.setVisibility(View.GONE);
                //clickHere.setVisibility(View.VISIBLE);
                confidence.setVisibility(View.VISIBLE);
                result.setVisibility(View.VISIBLE);

                image = Bitmap.createScaledBitmap(image, imageSize, imageSize, false);
                classifyImage(image);

            } else {
                Uri dat = data.getData();
                Bitmap image = null;
                try {
                    image = MediaStore.Images.Media.getBitmap(this.getContentResolver(), dat);
                } catch (IOException e) {
                    e.printStackTrace();
                }
                imageView.setImageBitmap(image);

                image = Bitmap.createScaledBitmap(image, imageSize, imageSize, false);
                classifyImage(image);
            }
        }
        super.onActivityResult(requestCode, resultCode, data);
    }
}
```