

IoT-Based Smart Kitchen Inventory and Auto Shopping Assistant

**Dr. P. N. Shinde¹, Prof. N. R. Janavekar², Vaishnavi Katare³, Dnyaneshwari Kulkarni⁴,
Samruddhi Tilekar⁵**

Head of Department, Electronics and Telecommunication, TSSM's BSCOER, Pune, India¹

Professor, Electronics and Telecommunication, TSSM's BSCOER, Pune, India²

Student, Electronics and Telecommunication, TSSM's BSCOER, Pune, India³⁻⁵

Abstract: With the rapid growth in technology around, life is getting easier and easier. The world is shifting towards smartness and automation with the help of sensor technology, which is playing a huge role. Kitchen is an important space where the ingredients and the grocery items are still managed manually, which may lead to wastage of food, unnecessary purchases of items, making poor expense management. This paper proposes an IoT-Based Smart Kitchen Inventory and Auto Shopping Assistant which helps with the monitoring and management of the kitchen inventory with help of real-time data processing. This system helps from tracking the inventory to restocking the inventory to keeping a track on the expenses. The system uses Esp32 microcontroller, load cell sensors with HX711 module which helps in tracking the inventory system. The data is transmitted over Wi-fi to server backend. The system detects the items on low stock and generates a shopping list along with sending notification. Addition to this, an OCR (Optical Character Recognition)-based receipt scanning feature is added to extract the product and price information from the bill for automatic expense tracking and updating the inventory status. A web application is built to provide real-time status, shopping list and track of expense to the user.

Keywords: Esp32, HX711, Load cell Sensor, OCR, Inventory Monitoring, Internet of Things (IoT), Smart Kitchen, Real-time Monitoring, Optical Character Recognition, Expense tracking.

I. INTRODUCTION

In modern world, rapid growth in technology and Internet of Things (IoT) has increased the usage of smart automation in person's daily life. The concept of 'Smart Homes' has become popular as they help in performing household activities more efficiently and helping in saving one's time and energy. Automated systems help reduce the manual effort, saves time and improves the management of resources using sensors, microcontrollers and different communication technologies. One important area in this 'Smart homes' is the kitchen and automation over here can help overcome problems like manual tracking of items, no track of expense.

Traditional kitchen management usually depends on checking the stock manually. This process is time consuming and not sufficient as it may lead to the wastage of food, over purchasing, random shortages and tracking of expense is mainly difficult. User may forget to check the stock daily which may result in unexpected shortage of stock which may cause inconvenience during the cooking activity time. Managing the receipts and keeping the track of the system manually is a difficult task on daily basis. Hence, this increases the need for a system that can continuously monitor our kitchen inventory and provide real-time data.

A smart kitchen inventory system helps the user track the kitchen stock levels automatically and helps improve the household inventory management. With the help of IoT technology and sensor-based monitoring users receive real-time data of the stock levels. Such system can help reduce the wastage of food items, and avoid unnecessary purchases. System detects low-stock and shopping assistance can simplify the management of household items and overall improve the user convenience.

The proposed project helps in monitoring and managing the kitchen inventory using real-time data. The system uses ESP32 as the main controller, load cell sensors and HX711 modules to measure the quantity of the items in a container. The collected data is transferred to the Flask-based server backend over Wi-fi where the information is stored in SQL database. A web application is built which provides real-time tracking status, generated shopping list, low stock notification and record of expense. OCR-based receipt processing is integrated for automatic updating of the track of

inventory. The system helps in reducing the manual efforts, improve the management of inventory, helps making the management of inventory easier for the user.

The objectives of this project are:

1. To Develop an application detecting low stocks, generating shopping list, tracking of expenses.
2. To detect the quantity of the item present in the respective container and notifying the user when the quantity goes below threshold level.
3. Connect the data from the kitchen and present it through the web application to the user.
4. Tracking of expenses and updating the stock levels with the help of OCR based receipt feature.
5. Provide purchase links from the online grocery platforms such as Zepto, Blinkit and Big Basket.

II. RELATED WORK

The advantage of using a automated system is to make user communication over internet. It not only makes the communication easier but also helps decrease the human intervention and helps manage the task singlehandedly. The following are few works that are based on this field of automation combined with the Internet of Things (IoT):

In ‘Smart Kitchen - A Measurement System’ M. Azran et al. proposed smart tracking of kitchen environment using load cell sensors and IoT technology to measure the kitchen ingredients and monitor the level of inventory in real time. The system sends the data through a wireless communication and provides user with updates on an Android application. This solution helps the user maintain the kitchen inventory and manage the budgeting more effectively [1].

In ‘Design and Implementation of Stock-Aware Shelves using a Pervasive Computing’ S. Anturkar et al. proposed a smart shelf system that monitors the stock and uses load sensing and pervasive computing. The system generated a low stock alert notification to the user which helps in reducing manual efforts of tracking the inventory. The author mainly has focus on smart shelf automation and stock detection over real time using sensors and different wireless communication [2].

In ‘IoT- Based Food Inventory Tracking System for Domestic Applications’ S. Kakade et al. proposed the food inventory tracking for the domestic kitchen appliances. The proposed system uses sensors, wireless communication modules, cloud connectivity along with continuous monitoring of the food inventory levels. This solution helps in the reduction of food wastage and improve the inventory management with the automation [3].

In ‘OCR-Based Receipt Data Extraction System’ A. Katagri et al. mainly focused on the use of the Optical Character Recognition (OCR) technology for extracting the text present in receipts and bills. The system helps process the scanned receipt picture to identify the name of the product, quantity, price for the digital expense management and help keeping a record of the same. OCR technology helps reduce the manual entry of the respective data and helps in improving the tracking of the expense [4].

In ‘Cloud-Based Grocery Management System’ A. Munene et al. proposed a cloud-based grocery management system which is designed for the maintenance of the records of the inventory and monitoring the data real-time. This system uses the cloud storage and database technologies to keep a track in the level of stock and helps in generating notification when the stock hit low. This solution provides improved inventory management [5].

III. PROPOSED SYSTEM

A. OVERVIEW OF SYSTEM

The system is an IoT-based inventory monitoring system developed to reduce the manual effort in kitchen management. System continuously monitors the data of quantity of ingredients using load sensor cells which are further integrated with HX711 module and ESP32 controller. The sensor data is transferred to Flask-based backend server and it is stored in SQLite database.

The system continuously checks the level with a predefined threshold value. When the quantity of any ingredient falls below the threshold values, the system automatically generates a low-stock alert to the user. This helps user understand and maintain the inventory.

The system also provides direct links from online grocery platform such as Zepto, Blinkit, BigBasket of the items that are on low-stock level.

In addition to this, system also integrates OCR technology for receipt scanning and automatic management of the expense.

B. SYSTEM ARCHITECTURE

The system is designed in 2 separate sections. The first part is the hardware section and the second consists of the software which are together integrated and automated. The overall architecture of the system includes following major components:

- ESP32 Microcontroller
- Load cell Sensors
- HX711 Amplifier Module.
- Flask Backend Server
- SQLite Database
- OCR Module
- Web dashboard
- Notification System

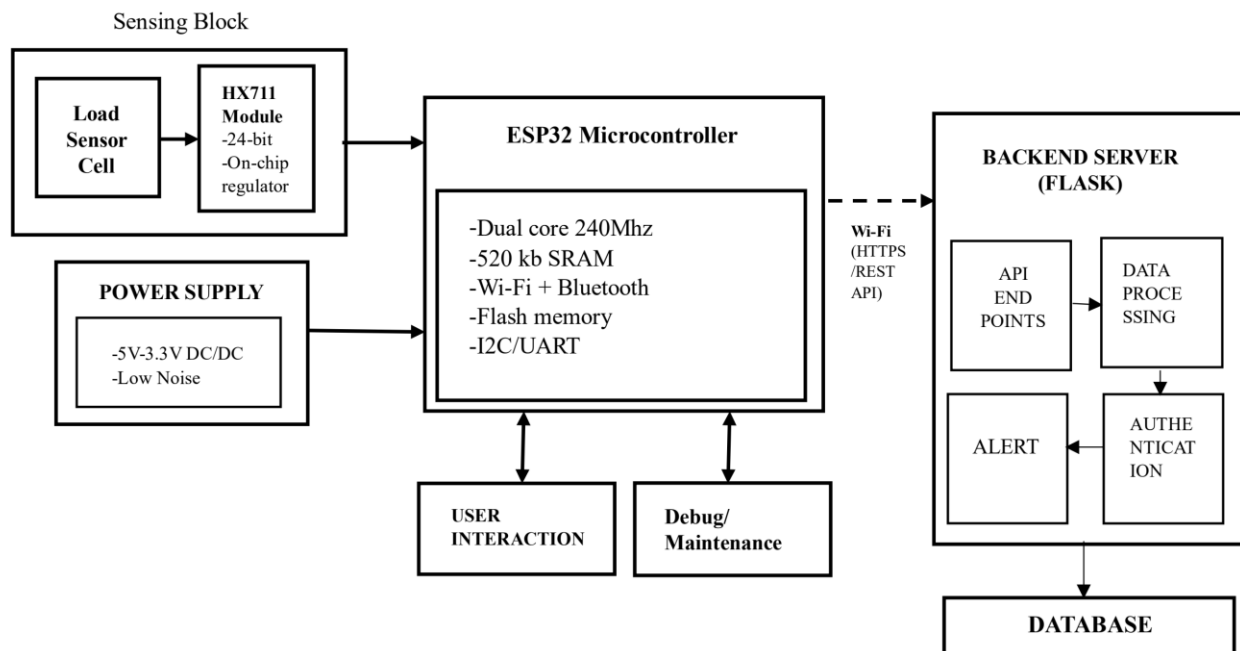


Fig. 1 Block Diagram of Proposed Model

The load cells perform the function of measuring the weight of the containers placed on the platform. Since the load sensor cell generates very small analog signals, the HX711 amplifier module helps to amplify the analog signals to digital values used for processing.

ESP32 controller acts as the central processing and communication unit of the system. It receives the data from HX711 module continuously and transmits the data to flask backend server through Wi-Fi communication.

The Flask backend server receives the data, processes it and then show the required results on Web dashboard / User Interface and then the data is simultaneously stored in the database.

C. WORKING PRINCIPLE

The model works in following steps –

Load cell sensor is placed below the containers bottom; this load cell continuously measures the weight of the containers and generate analog signals which corresponds to the quantity of the stored ingredients.

The HX711 module converts these analog signals into digital signals and these digital signals are transferred to the controller which processes the data for inventory tracking.

The ESP32 transfers the processed data to the backend server through a Wi-Fi communication. Backend server continuously monitors the stock level with a predefined threshold level. System automatically generates notifications when stock turns low, shopping assistance like Zepto, Blinkit, Big Basket is provided to help user purchase required items quickly. Aimed at expense management, the user uploads a scanned grocery slip/receipt/bill into system. The system processes the image and extracts product name, quantities, and prices.

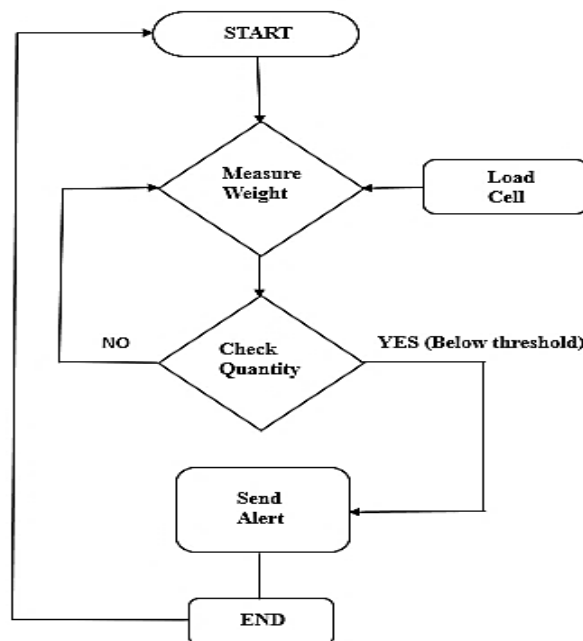


Fig.2 Flow Chart of Proposed Model

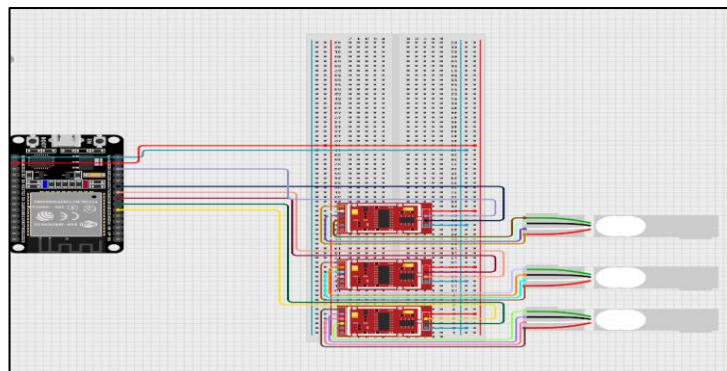
IV. HARDWARE SETUP

Fig. 3 Simulation figure

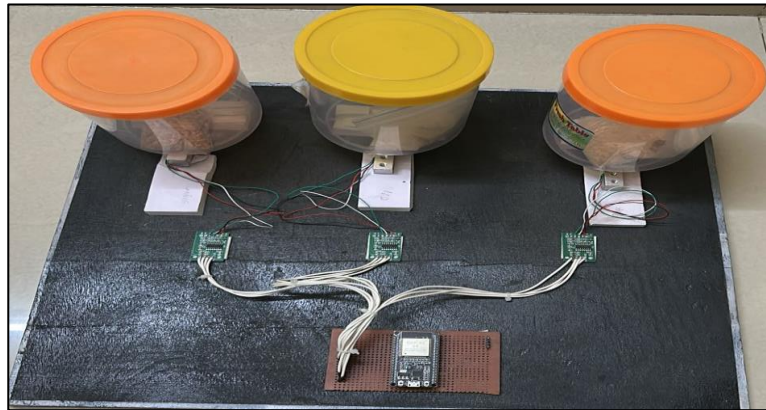


Fig. 4 Hardware Setup

V. SOFTWARE SETUP

1. Arduino IDE

The program is uploaded using Arduino IDE. Below are the steps followed for programming and uploading the code:

- Open Arduino IDE.
- Install ESP32 board package from board manager.
- Connect ESP32 controller with USB cable.
- Select the appropriate ESP32 board from Tools – board menu.
- Select the appropriate COM port from Tools – Port menu.
- Write proper code for sensor interfacing and Wi-Fi connection.
- Click the upload button to upload the code on ESP32.
- Open serial monitors to verify the sensor readings and Wi-Fi connection.

2. Backend Server Setup

The backend is build using Python and Flask framework. Flask handles the inventory data processing, OCR operations, database management and communication of dashboard. The following are the steps used for backend setup:

- Install Python in system.
- Install Flask using pip installer.
- Create Flask files for backend processing.
- Configure database connection.
- Develop APIs for receiving data from ESP32
- Implement threshold checking and notification functions.
- Run Flask server using command prompt.
- Verify server communication with ESP32.

3. Dashboard Setup

Dashboard is developed with the help of HTML, CSS and JavaScript technologies. It is used to display the stock levels, send notifications, OCR data and shopping assistance links.

- Create frontend pages using HTML and CSS.
- Design user interface for monitoring.
- Implement JavaScript functions for data updating data.

- Connect dashboard with Flash backend APIs.
- Display real-time data and alerts.
- Integrate receipt upload for OCR processing.

4. Algorithms for programming

4.1 Algorithm for Inventory Monitoring System

- Step 1: Start.
- Step 2: Initialize ESP32 controller.
- Step 3: Initialize HX711 and load cell sensor.
- Step 4: Connect ESP32 to Wi-Fi network.
- Step 5: Read weight values from load sensors.
- Step 6: Convert analog values to digital values using HX711 module.
- Step 7: Calculate quantity of inventory.
- Step 8: Send inventory data to Flask backend server.
- Step 9: Send inventory data to Database.
- Step 10: Repeat process.
- Step 11: End.

4.2 Algorithm for Threshold Detection and Notification System

- Step 1: Start.
- Step 2: Take inventory data from database.
- Step 3: Compare inventory data with threshold value.
- Step 4: Check if quantity is below threshold value.
- Step 5: If low, generate low-stock alert.
- Step 6: Update on dashboard.
- Step 7: Generate shopping list links.
- Step 8: Continue monitoring.
- Step 9: End.

4.3 Algorithm for OCR bill processing

- Step 1: Start.
- Step 2: Upload receipt image.
- Step 3: Process image for OCR extraction.
- Step 4: Apply OCR processing by Tesseract OCR.
- Step 5: Extract product name, quantity and price.
- Step 6: Update expense record.
- Step 7: Update inventory data.
- Step 8: Display the information.
- Step 9: End.

4.4 Algorithm for Shopping List and Link

- Step 1: Start.
- Step 2: Check quantity continuously.
- Step 3: Detect low-cost items.
- Step 4: Generate shopping list.
- Step 5: Generate product links.
- Step 6: Display links on dashboard.
- Step 7: End.

VI. RESULT ANALYSIS

1. Dashboard monitoring result

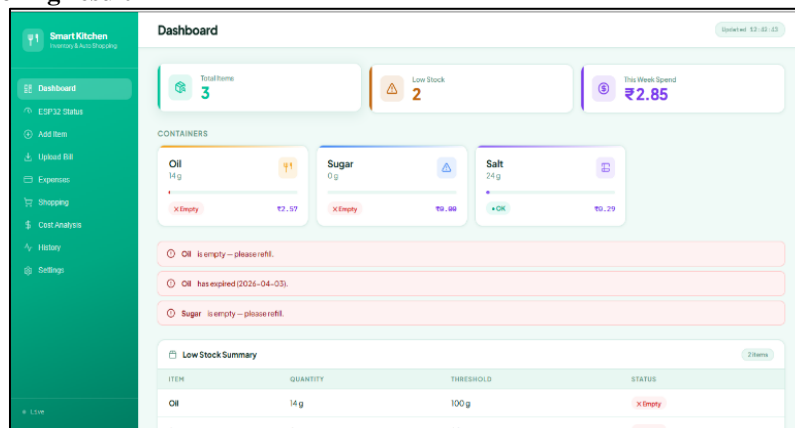


Fig. 6.1 Main Page

It displays the stock level and current inventory quantity for each of three containers. The system automatically generates alerts when the stock is low.

Table 1. Inventory status analysis

Item	Quantity	Threshold	Status
Oil	14g	100g	Urgent
Sugar	0g	20g	Empty
Salt	24g	20g	Normal

2. OCR bill Upload

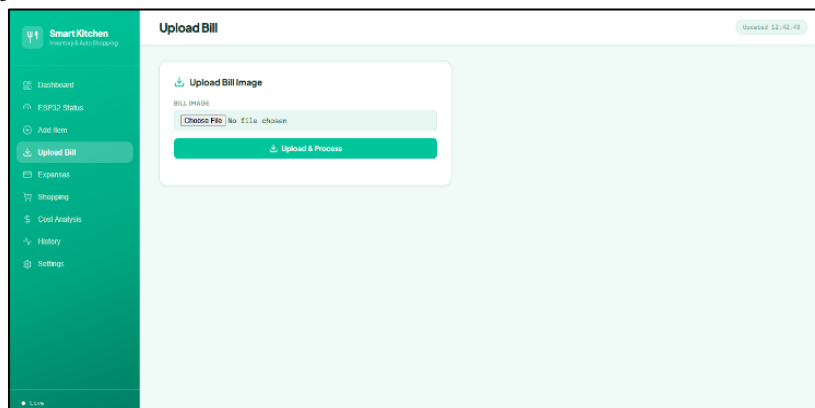


Fig. 6.2 Receipt Page

It allows users to upload image of receipt to extract the required information to track expenses and updating the stock levels.

3. Shopping Assistant

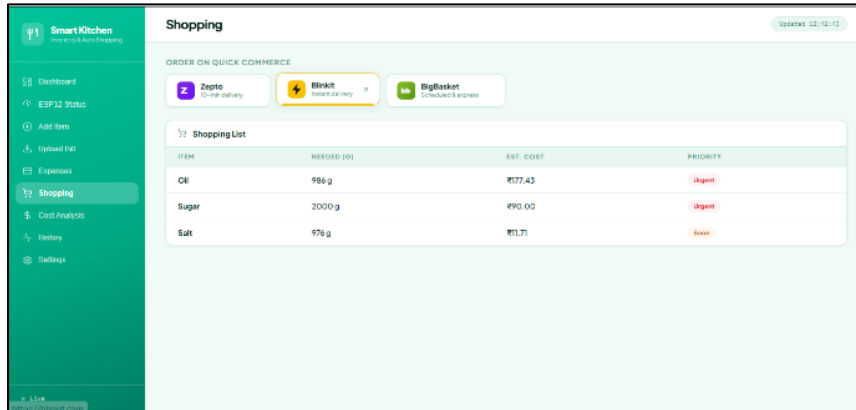


Fig. 6.3 Shopping Page

It generates the low stock items list and provides users with purchase links.

4. Expense Tracking



Fig. 6.4 Expense Tracking Page

It displays the graphical visualization of the expense. It helps users to keep a track on their expense pattern.

5. Settings and Threshold Defining

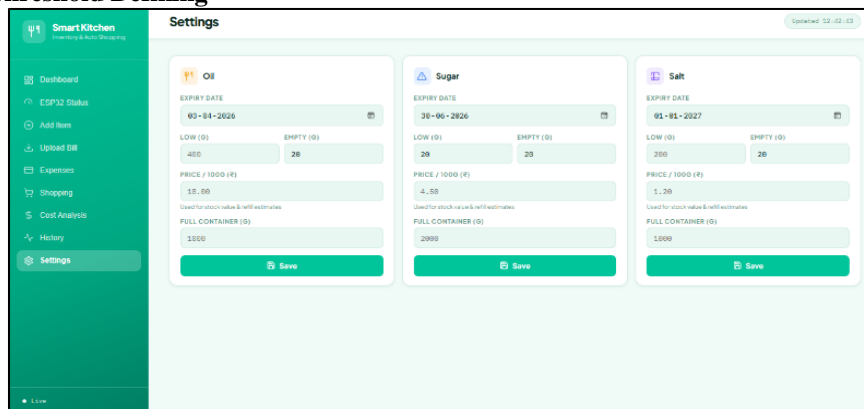


Fig. 6.5 Settings Page

It allows users to set a defined threshold value including expiry dates, limits, pricing of items. It helps in creating flexibility.

Table 2. Threshold parameters

Item	Low Threshold	Empty Threshold	Full Capacity
Oil	100g	10g	1000g
Sugar	20g	10g	500g
Salt	20g	10g	500g

Along with the individual data, the following analysis shows the system proposed proves a real-time inventory monitoring which stands out to be efficient. The System successfully generates alerts and Shopping assistace links within a short time of response.

Table 3. Performance Analysis

Sr. No.	Parameter	Values Observed
1	Sensor Accuracy	± 6 g
2	OCR Extraction Accuracy	89%
3	Data transmission via Wi-Fi Delay	5 sec
4	Low-stock Detection Time	2-3 sec
5	Notification Response Time	4-5 sec
6	OCR Processing time	2-3 sec
7	Power Consumption	1.5-2 W
8	Inventory Monitoring Reliability	93%
9	Shopping Link Generation Time	2 sec

VII. ADVANTAGES OF SYSTEM

The system continuously monitors the quantity of grocery items using load cells.

Reduces manual checking of grocery containers and improves the inventory management.

The system provides low-stock detection using predefined values. This helps maintain required inventory level and prevent the sudden shortages of essential kitchen ingredients.

OCR model helps extract all the data and help user keep a track on the expense. This helps in reducing the manual task of keeping a record of expense.

Shopping assistance feature helps in purchasing the item on a quick note.

The proposed system helps in reducing food wastage by tracking expiry related information.

VIII. CONCLUSION

With the rapid growth in population, urbanization and industrialization the kitchen management has become a global challenge. system provides solutions to the people using it on small scale as well as large scale. The prototype developed improves the convenience, reduces the food wastage and minimizes the unnecessary purchases, thereby reducing the manual efforts and improving the usability. Integration of platforms such as Blinkit, Zepto and BigBasket enhanced the practicality of the system for household system. Finally, the project successfully tried to fulfil all the objectives and hope this research would be of benefit. The systems turn out to be cost efficient. Because of the low cost, effective design and implementation this system became an essential prototype.

We would like to express our sincere gratitude to our project guide, faculty members and Head of department for the constant motivation and support.

We are really thankful for all the guidance provided and directions shown. We are thankful for our department for providing all the necessary facilities and resources required throughout.

REFERENCES

- [1] M. Azran, S. Rahman, and A. Karim, "Smart Kitchen - A Measurement System," *International Journal of Computer Applications (IJCA)*, vol. 179, no. 2, pp. 15–20, 2017.
- [2] S. Anturkar, P. Kulkarni, and R. Joshi, "Design and Implementation of Stock-Aware Shelves using Pervasive Computing," *International Journal of Computer Applications (IJCA)*, vol. 157, no. 9, pp. 10–15, 2017.
- [3] S. Kakade, S. Kadus, and S. A. Itkarkar, "IoT Based Smart Inventory Management System for Kitchen," *International Research Journal on Advanced Engineering and Management (IRJAEM)*, vol. 3, no. 9, pp. 2989–2993, Sept. 2025.
- [4] A. Katagri, S. Patil, and R. Desai, "OCR-Based Receipt Data Extraction System for Automated Expense Management," *International Journal of Computer Applications (IJCA)*, vol. 186, no. 72, pp. 59–65, 2025.
- [5] A. Munene and J. M. Wafula, "Cloud-Based Grocery Management System for Real-Time Inventory Monitoring," *International Journal of Computer Applications (IJCA)*, vol. 180, no. 21, pp. 20–25, 2018.
- [6] P. A. Patil and S. R. Deshmukh, "IoT-Driven Inventory Management Using Load Cells and NodeMCU for Real-Time Stock Monitoring," *International Journal of Scientific Research in Engineering and Management (IJSREM)*, vol. 8, no. 4, pp. 1–6, 2024.
- [7] R. Sharma and P. Gupta, "Smart Inventory Monitoring System Using ESP32 and Cloud Computing," *International Journal of Engineering Research & Technology (IJERT)*, vol. 12, no. 5, pp. 234–239, 2023.
- [8] Tesseract OCR Development Team, "Tesseract Open Source Optical Character Recognition Engine," Version 5.0, Google Open Source, 2023.
- [9] M. R. Khan and S. Ahmed, "Automated Expense Tracking System Using Optical Character Recognition Technology," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 15, no. 2, pp. 112–118, 2024.
- [10] V. Kumar, A. Singh, and R. Patel, "Cloud-Based Smart Grocery Management and Notification System Using IoT," *International Journal of Innovative Technology and Exploring Engineering (IJITEE)*, vol. 11, no. 8, pp. 45–51, 2022.