# SMART CONTAINER USING NODE MCU (ESP8266)

## 18CSE462J – INTRODUCTION TO INTERNET OF THINGS

*Submitted by*


**ANKIT PAUL [RA2111042010060]**

**KARTHIK MSV [RA2111042010058]**

**TAKSH PRAJAPATI [RA2111042010049]**

**SIDDHARTHA SHANKAR RAI [RA2111042010031]**

**DHAWAL SAHU [RA2111042010015]**


*Under the guidance of*

**Dr. K. Arthi**

Associate Professor, Department of Data Science and Business Systems

*In partial fulfilment of the requirements for the degree*

*of*

**BACHELOR OF TECHNOLOGY**

**in**

**COMPUTER SCIENCE ENGINEERING**

**AND BUSINESS SYSTEMS**



**DEPARTMENT OF DATA SCIENCE AND BUSINESS SYSTEMS**

**FACULTY OF ENGINEERING AND TECHNOLOGY**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**(Under Section 3 of UGC Act, 1956)**

**SRM NAGAR, KATTANKULATHUR – 603 203**

**CHENGALPATTU DISTRICT**


**JULY 2024**

SRM INSTITUTE OF SCIENCE&TECHNOLOGY
FACULTY OF ENGINEERING &TECHNOLOGY
S.R.M. NAGAR, KATTANKULATHUR – 603 203

# BONAFIDE CERTIFICATE

Certified that this project report "**SMART CONTAINER USING NODE MCU (ESP8266)**" is the bonafide work of "ANKIT PAUL [RA2111042010060], KARTHIK MSV [RA2111042010058], SIDDHARTHA SHANKAR RAI [RA2111042010031], TAKSH PRAJAPATI [RA2111042010049], DHAWAL SAHU [RA2111042010015] of IV Year/VII Semester B. Tech (CSBS) who carried out the project work under my supervision in SRM Institute of Science and Technology, Kattankulathur during the academic year 2024-2025 (Odd Semester).

SIGNATURE

Dr. K. Arthi
Associate Professor
Department of Data Science and Business Systems

SIGNATURE

Dr. M Lakshmi
Head of Department
Department of Data Science and Business Systems

# ACKNOWLEDGEMENT

We express our humble gratitude to Dr. C. Muthamizhchelvan, Vice Chancellor (I/C), SRM Institute of Science and Technology, for the facilities extended for the project work and his continued support.

We extend our sincere thanks to Dr. Revathi Venkataraman, Professor & Chairperson, School of Computing, SRM Institute of Science and Technology, for her invaluable support.

We wish to thank Dr. M. Lakshmi, Professor & Head, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for her valuable suggestions and encouragement throughout the period of the project work.

We are extremely grateful to our Academic Advisor Dr E. Sasikala, Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for her great support at all the stages of project work.

We register our immeasurable thanks to our Faculty Advisors, Dr. Jeba Sonia, and Dr. Mercy Theresa M, Assistant Professor, Data Science and Business Systems, SRM Institute of Science and Technology, for leading and helping us to complete our course.

Our inexpressible respect and thanks to our guide Dr. K. Arthi, Associate Professor, Department of Data Science and Business Systems, SRM Institute of Science and Technology, for providing us an opportunity to pursue our project under her mentorship. She provided the freedom and support to explore the research topics of our interest. Her passion for solving real problems and making a difference in the world has always been inspiring.

We sincerely thank staff and students of the Data Science and Business Systems Department, SRM Institute of Science and Technology, for their help during my research.

Finally, we would like to thank our parents, our family members and our friends for their unconditional love, constant support, and encouragement.

# ABSTRACT

The Smart Container is an IoT-enabled device designed to assist users in managing their kitchen inventory by providing alerts when items are running low and notifications for upcoming expiration dates. This report outlines the development process, the technologies used, and the potential impact on household management. By integrating sensors and a web/mobile application, the Smart Container aims to enhance the efficiency and convenience of managing food supplies.

The increasing demand for smart home solutions has driven the need for innovative kitchen management tools. The Smart Container addresses common challenges such as food waste, over-purchasing, and expired items. By leveraging Internet of Things (IoT) technology, the Smart Container continuously monitors the quantity of stored items and provides real-time data to users. This proactive approach ensures that users are always aware of their inventory status and can make informed decisions about restocking and consumption.

The device features weight sensors to detect the quantity of items and an interface for users to input expiration dates. When the quantity of an item falls below a predefined threshold, the Smart Container sends an alert to the user's connected device. Additionally, as the expiration date of a stored item approaches, the device changes color and sends a notification to ensure timely consumption or disposal.

The development of the Smart Container involved several stages, including hardware design, software development, and system integration. The hardware components include advanced sensors and microcontrollers, while the software comprises a user-friendly mobile and web application for managing alerts and notifications. The integration of these components ensures seamless communication and functionality.

The potential impact of the Smart Container on household management is significant. It not only reduces food waste and promotes efficient use of resources but also enhances the overall user experience by providing a convenient and automated solution for kitchen management. The Smart Container exemplifies the future of smart home technology, offering a practical and innovative approach to everyday challenge.

# LIST OF FIGURES

# TABLE OF CONTENTS

# Chapter 1: INTRODUCTION

## 1.1 Introduction

The Smart Container is an innovative IoT device designed to monitor the quantity of items stored within and alert users when supplies are running low. Additionally, it features an expiration date tracking system that provides visual and digital notifications as the expiration date approaches. This solution aims to address common issues in kitchen management, such as over-purchasing and food waste, by providing timely alerts and notifications to users.

In today's fast-paced world, many households struggle to keep track of their pantry inventory, leading to either the overstocking of certain items or the accidental consumption of expired products. The Smart Container leverages cutting-edge IoT technology to offer a seamless, automated solution to these problems. By ensuring that users are constantly informed about their inventory status and upcoming expiration dates, the Smart Container promotes better food management practices, reduces waste, and enhances overall kitchen efficiency.

## 1.2 Objectives of the Project

- **To develop an IoT-enabled container that alerts users when the contents are running low:** The Smart Container is equipped with sensitive weight sensors that can detect the quantity of items stored inside. When the quantity falls below a pre-set threshold, the device sends an alert to the user's mobile or web application, ensuring that they are aware of the need to restock.
- **To implement a system that tracks expiration dates and provides timely notifications:** Users can input the expiration dates of stored items via the application. The Smart Container then tracks these dates and provides visual alerts (such as changing LED colors) and digital notifications as the expiration date approaches. This feature helps users to consume items before they expire, thereby reducing food waste.
- **To integrate a web/mobile application for user interaction with the Smart Container:** The Smart Container is supported by a user-friendly web/mobile application that facilitates interaction with the device. Through the application, users can set alert thresholds, input expiration dates, and receive notifications. The application also provides detailed analytics on inventory usage, helping users to manage their kitchen supplies more effectively.

## 1.3 Project Focus

The primary focus of the project is to enhance kitchen management by providing an automated solution for tracking inventory levels and expiration dates of food items. This includes developing a user-friendly interface for managing notifications and ensuring the system's reliability and accuracy in detecting inventory changes and expiration dates.

To achieve this, the project emphasizes the following aspects:

- **Reliability of Sensor Data:** Ensuring that the weight sensors are accurate and reliable is crucial for the effective functioning of the Smart Container. The sensors need to be able to detect small changes in weight and provide consistent data to the microcontroller.
- **User-Friendly Interface:** The web/mobile application must be intuitive and easy to navigate, allowing users to quickly input data and access notifications. A well-designed interface will enhance user experience and ensure that the Smart Container is a practical addition to the kitchen.
- **Seamless Integration:** The hardware and software components of the Smart Container must be seamlessly integrated to provide real-time updates and notifications. This includes ensuring that data is accurately transmitted from the container to the cloud server and then to the user's device.
- **Scalability and Flexibility:** The Smart Container should be scalable to accommodate different types and sizes of containers. Additionally, the system should be flexible enough to adapt to various kitchen environments and user preferences.
- **Security and Privacy:** As the Smart Container involves the collection and transmission of data, it is essential to implement robust security measures to protect user data and ensure privacy.

The Smart Container project aims to create a reliable, user-friendly, and efficient solution for kitchen management, leveraging the power of IoT technology to simplify everyday tasks and promote sustainable living practices.

## 1.4 Organization of Report

This report is organized into the following chapters:
- Chapter 1: Introduction
- Chapter 2: Background Material
- Chapter 3: Methodology
- Chapter 4: Implementation
- Chapter 5: Conclusion & Future Scope
- References

# Chapter 2: BACKGROUND MATERIAL

## 2.1 Conceptual Overview

The Smart Container integrates sensors and IoT technology to provide real-time data on inventory levels and expiration dates. This overview covers the basic principles of IoT, sensor technology, and data transmission. The Internet of Things (IoT) refers to the network of physical devices that communicate and exchange data over the internet. In the context of the Smart Container, IoT enables the container to send data about the inventory levels and expiration dates to a central server, which processes the data and sends notifications to the user.

The IoT ecosystem of the Smart Container involves several key components:

- **Sensors:** These are responsible for collecting data. Weight sensors measure the quantity of items within the container, while date input mechanisms track the expiration dates entered by the user.
- **Microcontroller:** This component processes the data collected by the sensors. It performs necessary calculations and formats the data for transmission.
- **Communication Module:** This module is responsible for sending the processed data to the cloud server. The Smart Container uses Wi-Fi to ensure real-time data transmission.
- **Cloud Server:** The server receives data from the container, stores it, and processes it to generate notifications. This server acts as the central hub for data management and processing.
- **User Interface:** The web/mobile application serves as the interface between the user and the Smart Container. It displays inventory data, expiration dates, and notifications, allowing users to manage their kitchen inventory effectively.

The combination of these components allows the Smart Container to function seamlessly, providing users with up-to-date information and alerts about their kitchen inventory.

## 2.2 Technologies Involved

### *IoT Technology*

IoT involves connecting devices to the internet to share data. The Smart Container uses IoT to transmit inventory data to a cloud-based server for processing and notification. The IoT technology used in the Smart Container includes:

- **Device Connectivity:** Each Smart Container is equipped with a Wi-Fi module that enables it to connect to the internet. This connectivity is crucial for real-time data transmission and communication.
- **Cloud Computing:** The cloud server plays a critical role in the IoT ecosystem. It collects data from multiple Smart Containers, processes it, and sends notifications

to users. Cloud computing ensures scalability and reliability, allowing the system to handle a large number of devices and users.
- **Data Analytics:** The data collected by the Smart Container is analyzed to provide meaningful insights to users. This includes tracking inventory trends, predicting future needs, and optimizing kitchen management.

### *Sensor Technology*

The container uses weight sensors to measure the quantity of items and date input mechanisms to track expiration dates. These sensors provide accurate and timely data for the system to process. The sensor technology includes:

- **Weight Sensors:** These sensors are sensitive enough to detect small changes in weight, ensuring accurate measurement of the quantity of items in the container. They are calibrated to provide precise data, which is crucial for generating reliable notifications.
- **Date Input Mechanisms:** Users can input expiration dates manually using the web/mobile application. This data is then transmitted to the microcontroller and stored in the cloud server. The system tracks these dates and provides alerts as the expiration date approaches.
- **Sensor Integration:** The integration of sensors with the microcontroller and communication module ensures seamless data collection and transmission. This integration is essential for the real-time functionality of the Smart Container.

### *Communication Protocols*

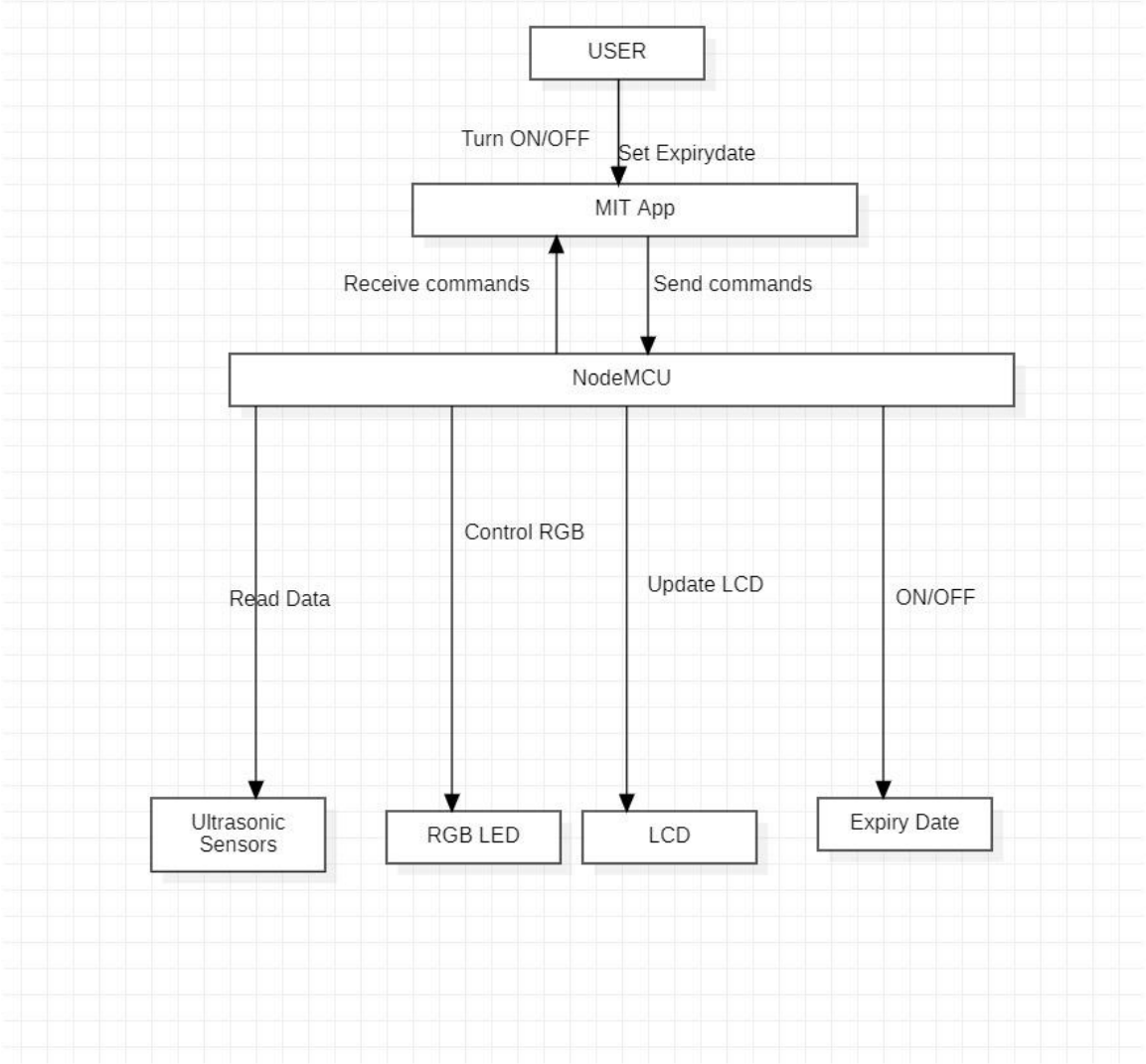
The container uses Wi-Fi for data transmission. Wi-Fi enables real-time communication between the container, the cloud server, and the user's web/mobile application. The communication protocols involved include:

- **Wi-Fi Connectivity:** Wi-Fi is chosen for its widespread availability and reliability. It allows the Smart Container to connect to the internet and transmit data without the need for additional infrastructure.
- **Data Encryption:** To ensure the security and privacy of the data transmitted, the Smart Container uses encryption protocols. This protects the data from unauthorized access and ensures that user information is secure.
- **Real-Time Communication:** The communication protocols are designed to support real-time data transmission, ensuring that users receive timely notifications about their inventory levels and expiration dates. This is crucial for the effectiveness of the Smart Container.

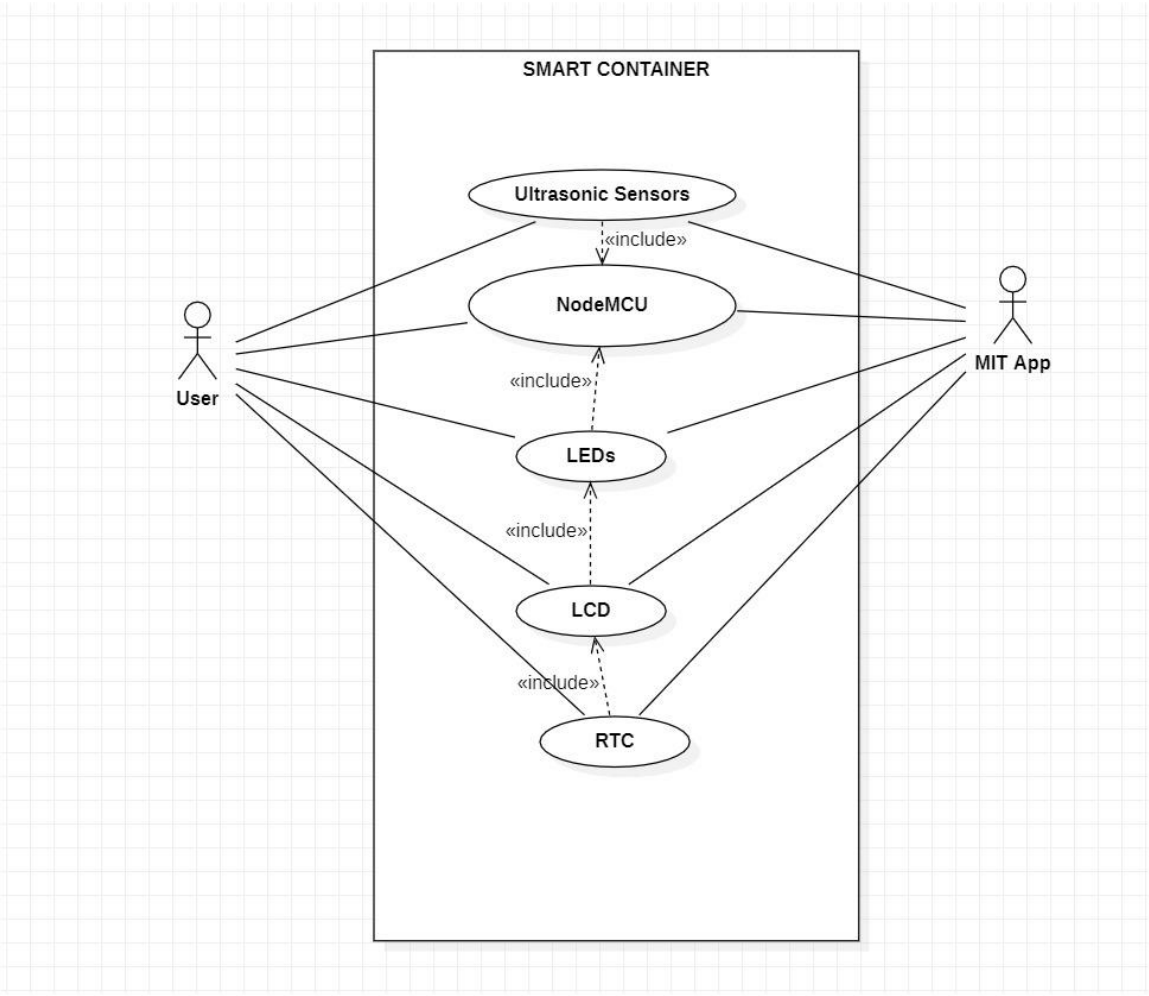
By leveraging these technologies, the Smart Container provides a robust and reliable solution for kitchen management. The integration of IoT, sensor technology, and communication protocols ensures that users have access to accurate and timely information, helping them to manage their kitchen inventory more efficiently and reduce food waste.

## 2.3 UML Diagrams

Data Flow Diagram



Use Case Diagram



# Chapter 3: METHODOLOGY

## 3.1 Detailed Methodology that will be Adopted

This section outlines the step-by-step process used to develop the Smart Container, including hardware design, software development, and system integration. The development process includes:

1. **Requirement Analysis**
    - **Stakeholder Consultation:** Engage with potential users and stakeholders to gather detailed requirements. This includes understanding user needs, identifying pain points in current kitchen management practices, and defining key functionalities of the Smart Container.
    - **Specification Development:** Develop detailed hardware and software specifications based on the gathered requirements. This includes defining the types of sensors needed, the microcontroller specifications, communication protocols, and the features of the web/mobile application.
    - **Feasibility Study:** Conduct a feasibility study to assess the technical and economic viability of the project. This involves evaluating the cost of components, potential manufacturing costs, and the overall project timeline.
2. **Hardware Design**
    - **Component Selection:** Select appropriate sensors, microcontrollers, and communication modules based on the specifications. This involves researching and choosing reliable and cost-effective components that meet the project requirements.
    - **Circuit Design:** Design the electrical circuits that will connect the sensors, microcontroller, and communication modules. This includes creating circuit diagrams and selecting appropriate connectors and power supplies.
    - **Prototyping:** Create a prototype of the hardware to test the design. This involves assembling the components on a breadboard or a custom PCB (Printed Circuit Board) and testing the functionality of the sensors and microcontroller.
    - **Enclosure Design:** Design the physical enclosure of the Smart Container. This includes considering the placement of sensors, the user interface, and the overall aesthetics and ergonomics of the container. The design should ensure that the sensors are positioned correctly for accurate data collection.
3. **Software Development**
    - **Firmware Development:** Develop the firmware that will run on the microcontroller. This includes programming the microcontroller to collect data from the sensors, process the data, and transmit it to the cloud server. The firmware must be optimized for power efficiency and real-time data processing.

- **Web/Mobile Application Development:** Develop the user interface for the Smart Container. This involves designing and coding a web/mobile application that allows users to input data, view inventory levels, and receive notifications. The application should be user-friendly and provide a seamless experience across different devices.
- **Cloud Server Development:** Develop the backend server that will store and process the data collected from the Smart Container. This includes setting up a database to store inventory data, developing APIs for data transmission, and implementing algorithms for data analysis and notification generation.
- **User Interface (UI) Design:** Design the UI of the web/mobile application, ensuring it is intuitive and easy to navigate. This includes creating wireframes, mockups, and interactive prototypes to gather user feedback and refine the design.

4. **System Integration**
   - **Hardware-Software Integration:** Integrate the hardware components with the software system. This involves ensuring that the sensors correctly interface with the microcontroller, and that the data collected is accurately transmitted to the cloud server.
   - **Network Setup:** Configure the communication protocols and network settings to enable seamless data transmission between the Smart Container, cloud server, and user devices. This includes setting up Wi-Fi connectivity and ensuring secure data transmission.
   - **Feature Implementation:** Implement the core features of the Smart Container, including real-time inventory tracking, expiration date notifications, and user interactions through the web/mobile application. This involves integrating the firmware, cloud server, and application to work together harmoniously.
5. **Testing and Validation**
   - **Unit Testing:** Test individual components and modules to ensure they function correctly. This includes testing the sensors, microcontroller, communication modules, and software components separately.
   - **System Testing:** Conduct comprehensive system testing to ensure all components work together as intended. This includes testing the entire system under various conditions to identify and fix any issues.
   - **User Testing:** Involve potential users in the testing process to gather feedback on the functionality and usability of the Smart Container. This includes conducting usability tests and gathering feedback on the user interface and overall experience.
   - **Performance Testing:** Test the system's performance under different conditions, such as varying network speeds and different inventory levels. This helps ensure the system remains reliable and responsive in real-world scenarios.
   - **Validation:** Validate the system against the original requirements and specifications to ensure it meets all objectives. This involves checking that

all functionalities are implemented correctly and that the system operates as expected.

6. **Deployment and Maintenance**
    - **Pilot Deployment:** Deploy the Smart Container in a limited setting to gather real-world data and feedback. This helps identify any final issues and make necessary adjustments before a full-scale launch.
    - **Full Deployment:** Once validated, deploy the Smart Container to a wider audience. This involves setting up production, distribution, and support systems.
    - **Ongoing Maintenance:** Provide ongoing maintenance and support to address any issues that arise post-deployment. This includes releasing software updates, providing customer support, and gathering user feedback for continuous improvement.

By following this detailed methodology, the Smart Container project aims to develop a robust, reliable, and user-friendly solution for kitchen management, leveraging the latest IoT and sensor technologies.

## 3.2 Overall Project Timeline

The project timeline is as follows:

Week 1: Requirement Analysis and Planning
Week 2: Hardware Design and Prototyping
Week 3: Software Development (Firmware)
Week 4: Software Development (Web/Mobile Application)
Week 5: System Integration and Testing
Week 6: Final Testing, Validation, and Documentation

# Chapter 4: IMPLEMENTATION

## 4.1 Modules

**Hardware Module**: The hardware module includes the physical components of the Smart Container, such as weight sensors, microcontrollers, and communication modules. The weight sensors are strategically placed at the base of the container to accurately measure the quantity of items inside. These sensors can detect minute changes in weight, ensuring precise inventory tracking.

The microcontroller acts as the brain of the Smart Container, processing the data received from the weight sensors. It is programmed to analyze the weight data and determine whether the quantity of an item has fallen below a pre-defined threshold. Once the threshold is crossed, the microcontroller triggers an alert mechanism.

The communication module, typically a Wi-Fi module, is responsible for transmitting the processed data to the cloud server. This module ensures real-time data synchronization between the Smart Container and the user's connected devices. Additionally, the hardware module includes an LED indicator system that changes color to signal different statuses, such as low inventory or approaching expiration dates.

**Software Module**: The software module includes the firmware running on the microcontroller and the web/mobile application used by the user. The firmware is a specialized software program embedded in the microcontroller that manages data collection from the sensors, processes the data, and controls the communication module. The firmware is designed to be efficient and reliable, ensuring continuous operation of the Smart Container.

The web/mobile application provides a user-friendly interface for managing notifications and viewing inventory data. The application allows users to set preferences, such as the minimum quantity threshold for alerts and the notification settings for expiration dates. Users can input the expiration dates of new items through the application, which then syncs with the Smart Container. The application also provides detailed analytics and reports on inventory usage, helping users make informed decisions about their shopping and consumption patterns.

## 4.2 Prototype

The prototype of the Smart Container includes a basic design with essential features for testing and validation. The prototype is equipped with weight sensors, a microcontroller, and a Wi-Fi communication module. The design is compact and modular, allowing easy assembly and testing of different components.

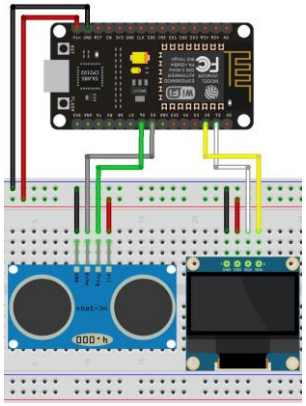
The software includes the basic functionalities for data collection and transmission, as well as a simple user interface for receiving notifications and viewing inventory data. The user interface is designed to be intuitive, with clear visual indicators for different statuses. The application also includes a setup wizard to guide users through the initial configuration process.

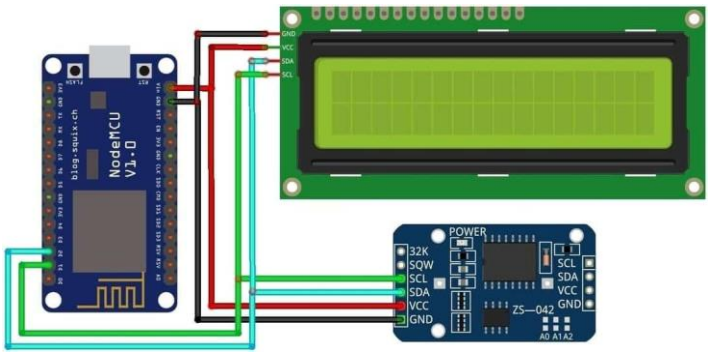
The prototype allows us to test the system's functionality and make necessary adjustments before developing the final product. During the testing phase, various scenarios are simulated to evaluate the accuracy of the weight sensors, the reliability of data transmission, and the responsiveness of the notification system. Feedback from initial users is gathered to identify potential improvements in both hardware and software components.

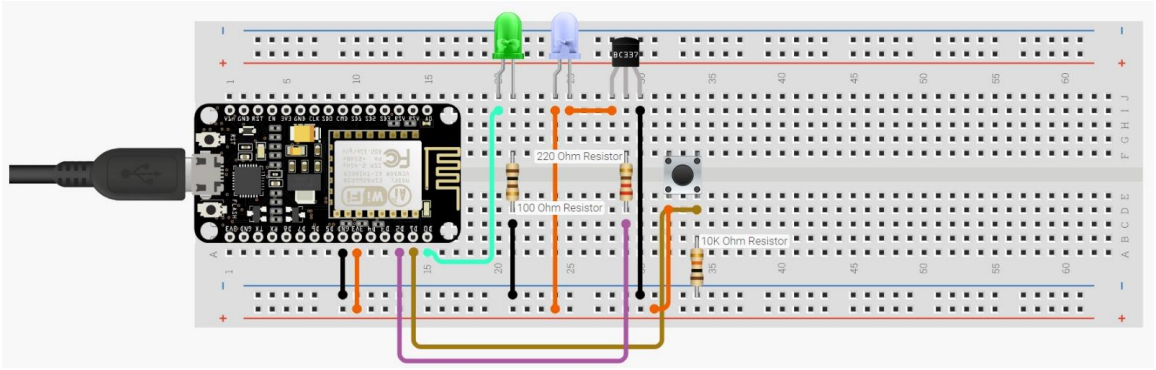
In addition to functional testing, the prototype undergoes stress testing to ensure durability and longevity. This includes subjecting the Smart Container to different environmental conditions, such as changes in temperature and humidity, to ensure it operates reliably in a typical kitchen environment.

The insights gained from the prototype testing phase are critical for refining the final design and ensuring the Smart Container meets the needs and expectations of users. The final product will incorporate the lessons learned, with enhancements in sensor accuracy, user interface design, and overall system integration.

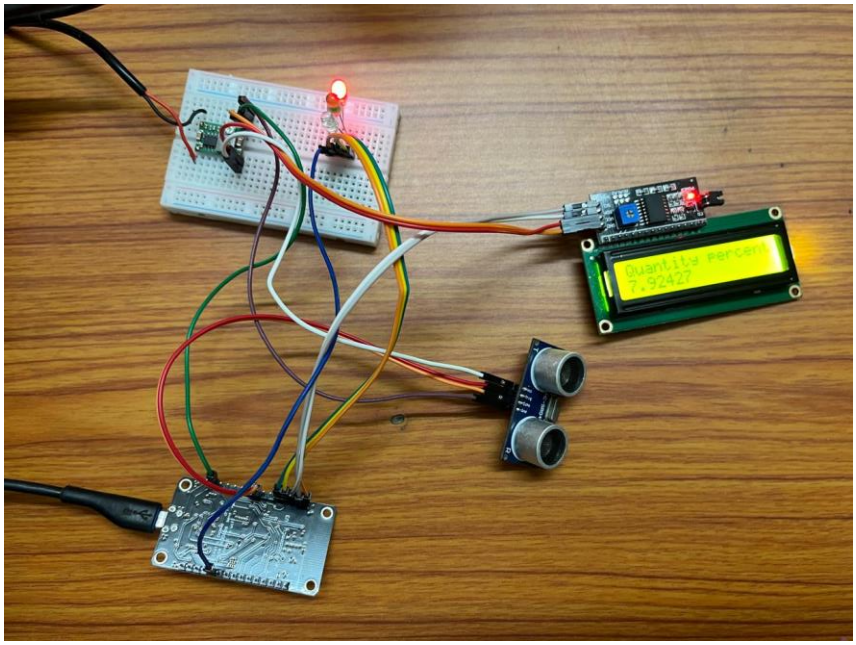
## 4.3 Circuit Diagrams

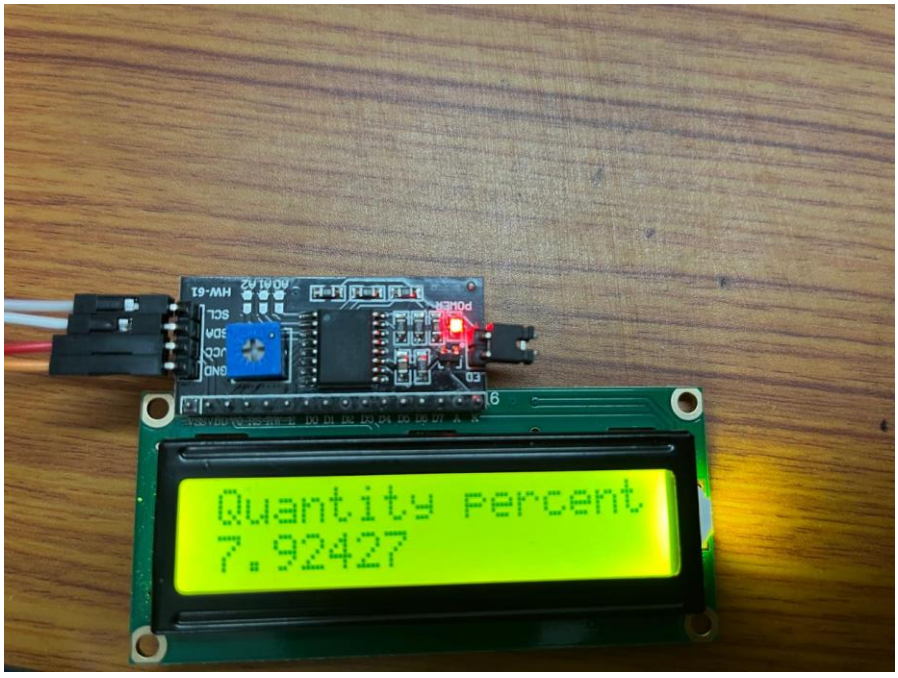



NodeMCU
V1.0
POWER
32K
SQW
SCL
SDA
VCC
GND
ZS-042


220 Ohm Resistor
100 Ohm Resistor
10K Ohm Resistor

## 4.4 Prototype Images



## 4.5 Codes

```
#include <Wire.h>

#include <LiquidCrystal_I2C.h>

#include <ESP8266WiFi.h>

const char WiFiPassword[] = "9004652173";

const char AP_NameChar[] = "EmbeddedGate" ;


WiFiServer server(80);


String header = "HTTP/1.1 200 OK\r\nContent-Type: text/html\r\n\r\n";

String html_1 = "<!DOCTYPE html><html><head><title>LED
Control</title></head><body><div id='main'><h2>LED Control</h2>";

String html_2 = "<form id='F1' action='LEDON'><input class='button' type='submit'
value='LED ON' ></form><br>";

String html_3 = "<form id='F2' action='LEDOFF'><input class='button' type='submit'
value='LED OFF' ></form><br>";

String html_4 = "</div></body></html>";


String request = "";

int LED_Pin = D0;

int counter = 0;

LiquidCrystal_I2C lcd(0x27, 16, 2); // Set the LCD address to 0x27 for a 16 chars and 2
line display

const int trigPin = 12;//D6

const int echoPin = 14;//D5

#define SOUND_VELOCITY 0.034
```

```
long duration;
float distanceCm;
float distanceInch;

void setup()
{
   Serial.begin(115200);
   lcd.init();                    // Initialize the LCD
   lcd.backlight();               // Turn on the backlight
   lcd.clear();  // Starts the serial communication
   pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output
   pinMode(echoPin, INPUT);// Clear the LCD screen
    pinMode(LED_Pin, OUTPUT);
   boolean conn = WiFi.softAP(AP_NameChar, WiFiPassword);
   server.begin();

}

void loop()
{

   // Check if a client has connected
   digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
```

```
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);

// Reads the echoPin, returns the sound wave travel time in microseconds
duration = pulseIn(echoPin, HIGH);

// Calculate the distance
distanceCm = duration * SOUND_VELOCITY/2;



// Prints the distance on the Serial Monitor
Serial.print("Distance (cm): ");
Serial.println(distanceCm);
int x = distanceCm;
delay(1000);
lcd.setCursor(0, 0);              // Set the cursor to the first column and first row
lcd.print("Quantity percentage");// Print some text
lcd.setCursor(0, 1);
lcd.print(distanceCm);
  WiFiClient client = server.available();
  if (!client)  {  return;  }

  // Read the first line of the request
  request = client.readStringUntil('\r');
```

```
    if      ( request.indexOf("LEDON") > 0 )  { digitalWrite(LED_Pin, LOW);  }
    else if  ( request.indexOf("LEDOFF") > 0 ) { digitalWrite(LED_Pin, HIGH);   }


    client.flush();


    client.print( header );
    client.print( html_1 );
    client.print( html_2 );
    client.print( html_3 );
    client.print( html_4);


    delay(5);
  // The client will actually be disconnected when the function returns and 'client' object is detroyed
  // Clears the trigPin


}
```

# Chapter 5: CONCLUSION & FUTURE SCOPE

## 5.1 Conclusions

The Smart Container project successfully demonstrates the potential of IoT technology in enhancing kitchen management. By providing real-time alerts for low inventory levels and upcoming expiration dates, the Smart Container helps users manage their kitchen supplies more efficiently, reducing food waste and improving overall convenience. The project achieved its primary objectives, including the development of a functional prototype and the integration of hardware and software components. Future improvements and additional features can further enhance the system's usability and effectiveness.

The project has highlighted several key outcomes and benefits:

- **Enhanced Inventory Management:** The Smart Container ensures that users are always aware of their inventory levels, preventing over-purchasing and minimizing instances of running out of essential items. This contributes to better planning and more organized kitchen management.
- **Reduction in Food Waste:** By providing timely notifications about upcoming expiration dates, the Smart Container encourages users to consume food items before they spoil, thus significantly reducing food waste. This is a critical step towards more sustainable living practices.
- **User Convenience:** The integration of a user-friendly web/mobile application allows users to effortlessly monitor and manage their kitchen inventory from anywhere. This added convenience simplifies the task of inventory management and fits seamlessly into the user’s daily routine.
- **Scalability and Adaptability:** The modular design of the Smart Container makes it scalable and adaptable to different kitchen environments. Users can customize the system according to their needs, whether they require multiple containers or different sensor configurations.
- **Technological Innovation:** The project showcases the innovative application of IoT technology and sensor integration in everyday household tasks. This sets a precedent for future developments in smart home devices, paving the way for more intelligent and interconnected home environments.

## 5.2 Future Scope of Work

1. **Integration with Voice Assistants**: Integrating the Smart Container with popular voice assistants such as Amazon Alexa and Google Assistant can significantly enhance user convenience by allowing hands-free operation. This integration can enable users to interact with the Smart Container using simple voice commands, making it easier to check inventory levels, set expiration date alerts, and receive notifications. For instance, users can ask their voice assistant to list items that are running low or expiring soon, add items to their shopping list, or even reorder supplies automatically through voice commands. This feature can make the Smart Container more accessible and user-friendly, especially for individuals with disabilities or those who prefer voice interaction over traditional interfaces.

2. **Enhanced User Interface**: Developing a more sophisticated user interface for the web/mobile application can provide users with a richer and more intuitive experience. This includes features such as:

- **Inventory Management:** A comprehensive dashboard that displays current inventory levels, categorized by item type, along with historical data and usage trends. Users can easily add, remove, or update items and track their consumption patterns.
- **Shopping List Generation:** An intelligent shopping list feature that automatically generates a shopping list based on inventory levels and upcoming expiration dates. Users can customize the list, set priorities, and even share it with family members or sync it with their preferred online grocery service.
- **Detailed Analytics:** Advanced analytics and visualizations that provide insights into consumption habits, spending patterns, and potential areas for reducing waste. This can help users make informed decisions about their purchasing and usage behaviors.

3. **Expansion to Monitor Other Items**: Extending the functionality of the Smart Container to monitor other types of items, such as medicines and toiletries, to provide a comprehensive household management solution.

# REFERENCES


**Books and Journals:**

- Ashton, K. (2009). "That 'Internet of Things' Thing." RFID Journal.
- Gubbi, J., Buyya, R., Marusic, S., & Palaniswami, M. (2013). "Internet of Things (IoT): A vision, architectural elements, and future directions." Future Generation Computer Systems, 29(7), 1645-1660.
- Holler, J., Tsiatsis, V., Mulligan, C., Avesand, S., Karnouskos, S., & Boyle, D. (2014). "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence." Academic Press.
- Li, S., Da Xu, L., & Zhao, S. (2015). "The internet of things: a survey." Information Systems Frontiers, 17(2), 243-259.
- Madakam, S., Ramaswamy, R., & Tripathi, S. (2015). "Internet of Things (IoT): A literature review." Journal of Computer and Communications, 3(05), 164.

**Web Articles and Reports:**

- "How IoT is Revolutionizing the Food Industry." (2019). Food Safety Magazine. Retrieved from https://www.foodsafetymagazine.com/magazine-archive1/december-2018january-2019/how-iot-is-revolutionizing-the-food-industry/
- "IoT in Smart Kitchen Appliances: Benefits and Applications." (2020). IoT For All. Retrieved from https://www.iotforall.com/iot-in-smart-kitchen-appliances-benefits-and-applications
- "Smart Containers: How IoT is Shaping the Future of Shipping." (2021). IoT World Today. Retrieved from https://www.iotworldtoday.com/2021/03/12/smart-containers-how-iot-is-shaping-the-future-of-shipping/

**Technical Documentation:**

- Arduino. (2021). "Arduino Uno Rev3." Retrieved from https://store.arduino.cc/usa/arduino-uno-rev3
- Espressif Systems. (2020). "ESP8266EX Datasheet." Retrieved from https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf
- Texas Instruments. (2019). "HDC1080: Low Power, High Accuracy Digital Humidity Sensor with Temperature Sensor." Retrieved from https://www.ti.com/lit/ds/symlink/hdc1080.pdf