A PROJECT REPORT ON

Secure Farm-to-market with QR and IOT

SUBMITTED TO
MIT SCHOOL OF COMPUTING, LONI, PUNE IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE

BACHELOR OF TECHNOLOGY
(Computer Science & Engineering)

BY

Samarth Kale	Enrollment No: ADT23SOCB0955
Parth Hole	Enrollment No: ADT23SOCB0700
Shrushti Shinde	Enrollment No: ADT23SOCB1090
Tamanna Khatik	Enrollment No: ADT23SOCB1201

Under the guidance of

Prof.Dr.Survarna Joshi

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

MIT SCHOOL OF COMPUTING
MIT Art, Design and Technology University Rajbaugh Campus, Loni-Kalbhor, Pune 412201

2024-25

MIT SCHOOL OF COMPUTING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
MIT ART, DESIGN AND TECHNOLOGY UNIVERSITY,
RAJBAUG CAMPUS, LONI-KALBHOR, PUNE 412201

CERTIFICATE

This is to certify that the project report entitled

“Secure Farm-to-market with QR and IoT”

Submitted by

Samarth Kale	Enrollment No: ADT23SOCB0955
Parth Hole	Enrollment No: ADT23SOCB0700
Shrushti Shinde	Enrollment No: ADT23SOCB1090
Tamanna Khatik	Enrollment No: ADT23SOCB1201

It is a bonafide work carried out by them under the supervision of Prof. Dr. Suvarna Joshi and it is submitted towards the partial fulfillment of the requirement of MIT ADT University, Pune for the award of the degree of Bachelor of Technology (Computer Science and Engineering)

Dr. Suvarna Joshi
Guide

Dr. N.P.Kulkarni
Head of Department

Prof. Suresh Kapare
Chief Coordinator-PBL

Dr. Ganesh Pathak
Dean

Seal/Stamp of the College
Place: Pune

DECLARATION

We, the team members

Name	Enrollment No. :
Samarth Kale	Enrollment No: ADT23SOCB0955
Parth Hole	Enrollment No: ADT23SOCB0700
Shrushti Shinde	Enrollment No: ADT23SOCB1090
Tamanna Khatik	Enrollment No: ADT23SOCB1201

Hereby declare that the project work incorporated in the present project entitled **"Secure Farm-to-market with QR and IoT"**is original work. This work (in part or in full) has not been submitted to any University for the award or a Degree or a Diploma. We have properly acknowledged the material collected from secondary sources wherever required. We solely own the responsibility for the originality of the entire content.

Date: 05/11/2025

Name & Signature of the Team Members

Member 1: Samarth Kale

Member 2: Parth Hole

Member 3: Shrushti Shinde

Member 4: Tamanna Khatik

Name and Signature of Guide:

Prof.Dr. Suvarna Joshi

Seal/Stamp of the College

Place: Pune

Date: 05/11/2025

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
MIT SCHOOL OF COMPUTING,
RAJBAUG, LONI KALBHOR,
PUNE – 412201

EXAMINER'S APPROVAL CERTIFICATE

The project report entitled **"Secure Farm-to-market with QR and IoT"** submitted by **Shrushti Shinde (ADT23SOCB1090), Tamanna Khatik (ADT23SOCB1201), Samarth Kale (ADT23SOCB0955), Parth H (ADT23SOCB0196)** in partial fulfillment for the award of the degree of Bachelor of Technology (Computer Science & Engineering) during the academic year 2024-25, of MIT-ADT University, MIT School OF COMPUTING, Pune, is hereby approved.

Examiners:

1.

2.

ACKNOWLEDGEMENT

We would like to express our sincere gratitude to all those who have contributed to the successful completion of this project, **“Smart IoT & QR-Based Basket System for Secure Farm-to-Market Traceability.”**

First and foremost, we are deeply thankful to our project guide, **Prof. Dr. Suvarna Joshi**, for her invaluable guidance, constructive feedback, and constant encouragement throughout the development of this project. Her expertise and mentorship have been instrumental in shaping our work.

We are grateful to the faculty members and staff of **MIT School of Computing (MIT-SOC)** for providing us with the necessary resources, technical insights, and a collaborative environment for conducting this research. Their constant support helped us explore IoT applications and automation technologies effectively.

We also extend heartfelt thanks to our peers and colleagues for their valuable inputs, discussions, and teamwork during each phase of the project. Their innovative ideas and problem-solving approach were crucial to refining our prototype.

We acknowledge the contribution of the open-source community and developers of various tools and components that were vital in implementing this project—especially **ESP32-CAM**, **IoT Lock Module**, **Relay Module**, **Arduino IDE**, and **Cloud Integration Platforms**.

Samarth Kale	Enrollment No: ADT23SOCB0955
Parth Hole	Enrollment No: ADT23SOCB0700
Shrushti Shinde	Enrollment No: ADT23SOCB1090
Tamanna Khatik	Enrollment No: ADT23SOCB1201

ABSTRACT

In today’s agricultural market, ensuring the authenticity, freshness, and safe transportation of fruits is a major challenge. Farmers often claim their produce is “fresh” or “organic,” yet buyers and authorities have no reliable way to verify whether the basket of fruits has been sealed, handled, and transported without tampering. This lack of traceability often leads to fraudulent practices, mixing of poor-quality fruits, loss of trust, and unfair pricing for farmers.

To address this issue, we have developed a **Smart IoT & QR-Based Basket System for Secure Farm-to-Market Traceability**. The proposed system combines **QR code verification** with an **IoT-enabled locking mechanism** to ensure transparency and tamper-proof handling of fruit baskets throughout their journey. When the basket is filled at the farm, a fixed **camera module (ESP32-CAM)** scans the basket’s QR code, captures the farmer’s details, and automatically triggers an **IoT lock** to seal the basket. The sealing event, along with the **time, date, and basket ID**, is uploaded to a **web-based record system** for tracking.

Only authorized personnel—such as laboratory officials, quality inspectors, or registered shopkeepers—can later unlock the basket using their **unique QR codes**. Every unlock event is also recorded and time-stamped on the website. This system thus provides a **complete digital record** of the basket’s handling from the farm to the consumer, ensuring **traceability, transparency, and authenticity**.

The project demonstrates the integration of **IoT, QR code technology, and automation** to enhance agricultural trust and fairness. It not only protects the farmer’s credibility but also ensures consumers receive genuine, untampered produce. Overall, this system represents a step toward **smart, secure, and transparent farm-to-market delivery**, supporting the future of **digital agriculture and supply-chain verification**

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1: INTRODUCTION

1.1 Introduction

Agriculture remains one of the most essential sectors in our society, providing food security and livelihood to millions of people. However, one of the persistent challenges faced by farmers and consumers alike is the lack of **transparency and authenticity** in the farm-to-market process. Farmers often claim their fruits are “fresh” or “organic,” yet there is no reliable method for verifying these claims once the basket leaves the farm. During transport or storage, the baskets can be tampered with—opened, mixed with lower-quality produce, or replaced entirely—without any trace or accountability.

To address this real-world issue, our project introduces a **Smart IoT & QR-Based Basket System** designed to ensure secure, transparent, and traceable handling of fruit baskets throughout their journey from farm to market. The system uses **IoT-enabled locks** and **QR code verification** to record sealing, handling, and unlocking events digitally.

When a basket is packed and sealed at the farm, a **fixed camera module (ESP32-CAM)** scans the QR code attached to the basket. The scan captures the farmer’s details and automatically activates an **IoT-controlled locking system** to secure the basket. Each sealing event is recorded with **time, date, and farmer ID** and stored in a **web-based record system**.

At later stages, only authorized personnel—such as laboratory officials or registered shopkeepers—can unlock the basket using their **unique QR credentials**. Each unlocking event is also logged and time stamped, ensuring that every handling step is digitally traceable. This provides a **tamper-proof, verifiable supply chain** that promotes fairness for farmers and trust for customers.

The system demonstrates the powerful combination of **IoT automation, QR technology, and web integration** in solving real-world agricultural challenges. It aims to create a smarter, safer, and more transparent supply chain that ensures the fruit quality promised by farmers truly reaches the end consumer.

1.2 Problem Statement

Farmers often claim that their fruits are “fresh” or “organic,” but there is no easy way for buyers or authorities to verify if the basket of fruits has been properly sealed, checked, and transported without tampering. Once the basket leaves the farm, anyone can open it, mix low-quality fruits, or replace them, and there is no reliable record of who handled it. This reduces trust, affects fair pricing for farmers, and misleads customers.

1.3 Motivation and Need for the System

The motivation behind this project arises from the need to establish **trust, traceability, and transparency** in the agricultural supply chain. Farmers deserve fair recognition and compensation for their genuine produce, and customers deserve assurance that the fruits they purchase are truly fresh and authentic.

Currently, the absence of a proper verification system results in fraud, price manipulation, and mistrust among stakeholders. With advancements in **IoT and automation technologies**, it is now possible to create a low-cost and effective system that ensures **tamper-proof packaging, digital monitoring, and secure record management**.

This project fulfills the following key needs:

- To create a **smart, automated solution** that seals fruit baskets securely at the source.
- To provide **traceability of each basket** through a digital QR-based identification system.
- To allow **authorized access only** to verified handlers and shopkeepers.
- To ensure **fair pricing and credibility** for farmers and **trust and authenticity** for consumers.
- To demonstrate how **IoT integration** can modernize agricultural logistics with minimal human error and maximum transparency.

1.4 Objectives of the Project

The main objectives of this project are:

- To design and implement a **Smart IoT-based system** that seals and tracks fruit baskets using QR verification.
- To automatically record **time, date, and basket details** when sealing or unlocking occurs.
- To develop a **web-based interface** for securely storing and monitoring handling records.
- To restrict access using **role-based QR authentication**, ensuring only authorized users can open baskets.
- To promote **fair trade, traceability, and consumer trust** in the farm-to-market ecosystem.

1.5 Scope and Limitations

Scope:

- The system provides secure sealing, tracking, and unlocking of fruit baskets using IoT and QR codes.
- Enables complete traceability from farm to market, including intermediate handling points.
- Uses **ESP32-CAM** for scanning QR codes and an **IoT lock** for automation.
- Stores events on a **web server**, allowing easy verification by authorities and customers.
- Can be extended for use with other perishable products such as vegetables, dairy, or grains.

Limitations:

- The system requires **internet connectivity** for real-time cloud updates.
- Power dependency for the IoT modules can affect continuous operation.
- Prototype is currently designed for **small-scale demonstration** and may require hardware upgrades for industrial use.
- Environmental factors like humidity or dust may affect sensor performance if not properly enclosed.

Chapter 2: LITERATURE SURVEY

2.1 Overview of Existing Work

In recent years, the integration of **IoT (Internet of Things)** and **Blockchain** technologies in the agricultural sector has emerged as a major solution for tackling the challenges of **fraud, tampering, and lack of transparency** in food supply chains. Several studies have emphasized the potential of combining **QR-based identification**, **IoT-enabled monitoring**, and **blockchain-based data storage** to create a **secure and traceable farm-to-market delivery system**.

This literature survey reviews **eight key research papers** that explore how these technologies enhance **traceability, authenticity, and consumer trust** in agricultural products. The findings from these studies have directly influenced the design and implementation of our **Smart IoT & QR-Based Basket System**, which ensures tamper-proof fruit handling and digital transparency.

2.2 Paper 1: Blockchain Technology in Food Supply Chains – Traceability, Transparency, and Trust

Authors: Galvez J.F., Mejuto J.C., Simal-Gandara J. (2018)
Source: *Trends in Food Science & Technology*
DOI: https://doi.org/10.1016/j.tifs.2018.07.001

This study explores how blockchain and IoT technologies can be used to establish transparency, data immutability, and fraud prevention in food supply chains.It introduces a blockchain model where every product movement is recorded in a decentralized ledger, reducing manipulation and providing accountability at each stage.

Relevance to Our Project:
This work supports the traceability foundation of our system, as we use QR codes and IoT logs to maintain a verifiable record of each basket's sealing and transport history.

2.3 Paper 2: IoT and Blockchain for Food Traceability – A Review

Authors: Kamble S.S., Gunasekaran A., Sharma R. (2020)
DOI: https://doi.org/10.1016/j.jclepro.2020.120908

The authors present a comprehensive review of IoT and blockchain applications in food traceability systems. The study identifies major benefits such as real-time monitoring, automated verification, and reduced fraud risk.

Relevance to Our Project:
The review highlights how IoT sensors and QR tags can serve as digital evidence of product authenticity — the same concept used in our smart basket to capture sealing and unlocking events in real time.

2.4 Paper 3: AgriBlockIoT – Blockchain and IoT Integration for Food Traceability

Authors: Hasan H.R., Salah K. (2019)
DOI: https://doi.org/10.1109/ACCESS.2019.2915667

This paper proposes **AgriBlockIoT**, a blockchain-enabled IoT framework that ensures data immutability and transparent record-keeping in agricultural systems. It connects IoT devices directly to blockchain networks to prevent tampering.

Relevance to Our Project:
This study strongly influences our design concept — the IoT module (ESP32-CAM and lock) in our system could be later extended to integrate blockchain for secure, auditable trace logs.

2.5 Paper 4: QR Code and IoT-Based Food Traceability System

Authors: Lin J., Shen Z., Zhang A. (2018)
DOI: https://doi.org/10.1109/ICIoT.2018.00045

This research proposes a hybrid food traceability system using **QR codes and IoT sensors**. The model captures product data at various checkpoints and stores it in a cloud system for public verification.

Relevance to Our Project:
Directly aligns with our mechanism — QR codes identify baskets and authorized handlers, while IoT locking and cloud upload ensure tamper-proof verification.

2.6 Paper 5: Blockchain for Agri-Food Traceability

Authors: Bumblauskas D., Mann A., Dugan B. (2020)
DOI: https://doi.org/10.1109/ACCESS.2020.2965081

This paper presents a practical blockchain-based solution for agro-food logistics, emphasizing **transparency, consumer trust, and fraud detection**.The authors demonstrate how blockchain transactions can map the entire food supply chain journey, from farm to consumer.

Relevance to Our Project:
Inspires the record-keeping component of our system — where every sealing and unlocking event could be verified via digital signatures or blockchain hashes in future versions.

2.7 Paper 6: Considering Fraud Vulnerability Associated with Credence Attributes – The Case of Organic Food

Author: Manning L. (2021)
DOI: https://doi.org/10.1016/j.foodcont.2021.108352

This study investigates fraudulent labeling and fake organic certifications in food markets. It suggests combining IoT-based traceability with certification records to prevent misrepresentation.

Relevance to Our Project:
Highlights the real-world issue our project solves — preventing false claims of "organic" by ensuring baskets remain sealed and verified digitally until delivery.

2.8 Paper 7: Enhanced Fruit Adulteration Detection Using Formaldehyde Sensor and Image Analysis

Authors: Agarwal R., Chavan Y., Kesarkar A., Tere A., Bhadane C. (2025)
Source: *SEEJPH Journal*
Link: https://www.seejph.com/index.php/seejph/article/view/6183

This paper introduces a hybrid sensor and image-analysis model to detect chemical adulteration in fruits. The authors integrate formaldehyde sensors and digital imaging for real-time fruit quality monitoring.

Relevance to Our Project:
Inspires potential future enhancements — integrating IoT sensors within the basket to detect fruit spoilage or adulteration before delivery.

2.9 Paper 8: Blockchain Traceability for Sustainable Agriculture

Authors: Casino F., Kanakaris V., Dasaklis T. (2020)
DOI: https://doi.org/10.1016/j.suscom.2020.100281

This study presents blockchain traceability as a sustainable framework for monitoring agricultural products. It introduces multi-tier tracking (farm → storage → retail) with IoT devices and blockchain verification.

Relevance to Our Project:

Supports the idea of multi-level traceability — the same approach we use in our smart basket system where data is logged at each handling stage, from farmer to shopkeeper.

2.10 Summary of Insights from Literature

Sr. No.	Title of Paper	Year	Authors	Key Contribution / Relevance
1	Blockchain Technology in Food Supply Chains	2018	Galvez J.F. et al.	Established blockchain as a base for tamper-proof food traceability.
2	IoT and Blockchain for Food Traceability	2020	Kamble S.S. et al.	Highlighted IoT-blockchain integration for supply chain visibility.
3	AgriBlockIoT	2019	Hasan H.R., Salah K.	Proposed blockchain-IoT hybrid model for secure traceability.
4	QR Code and IoT-based Food Traceability System	2018	Lin J. et al.	Demonstrated low-cost IoT + QR traceability suitable for developing regions.
5	Blockchain for Agri-Food Traceability	2020	Bumblauskas D. et al.	Emphasized blockchain to improve consumer trust and prevent fraud.
6	Fraud Vulnerability in Organic Food	2021	Manning L.	Addressed fake organic labeling; proposed IoT traceability as a solution.
7	Enhanced Fruit Adulteration Detection	2025	Agarwal R. et al.	Integrated IoT sensors for detecting adulteration and spoilage.
8	Blockchain Traceability for Sustainable Agriculture	2020	Casino F. et al.	Proposed end-to-end blockchain-IoT traceability framework for agriculture.

Table 1: Literature review insights summar

Chapter 3: SYSTEM DESIGN & METHODOLOGY

3.1 Core Concepts and Technologies Used

This project integrates **IoT (Internet of Things)** and **QR code technology** to create a tamper-proof and transparent fruit transportation system. The goal is to design a smart solution that can **automatically seal fruit baskets**, **record each handling event**, and **maintain traceability** through a cloud-based web portal.

The system revolves around three core concepts:

- **IoT Automation:**
  The IoT locking mechanism automatically seals and unseals the basket based on authorized QR verification.
- **QR-Based Identification:**
  Each basket and handler (farmer, lab official, shopkeeper) is assigned a unique QR code for authentication.
- **Cloud Connectivity:**
  All events such as sealing, unlocking, and inspections are logged and stored in a web database for future verification.

Technology / Component	Purpose / Functionality
ESP32-CAM Module	Captures and scans QR codes; acts as the primary IoT device for reading and verifying identification data.
IoT Lock Module (Solenoid Lock / Servo Motor)	Automatically seals or unseals the basket when triggered by an authorized QR code.
Relay Module	Acts as an electrical switch between the ESP32 and the lock mechanism for controlled operation.
Arduino IDE	Used for programming the ESP32 module and integrating hardware control logic.
Cloud / Local Web Server	Stores all sealing, unlocking, and verification records in a database accessible via the internet.
QR Code System	Provides unique identification for farmers, baskets, and authorized users.
Wi-Fi Connectivity	Enables the ESP32 to communicate with the cloud database or web interface in real time.
HTML, CSS, JavaScript	Used for developing a simple web interface to view the status and history of each basket.

Table 2: Core Technologies and their Purpose

These technologies collectively enabled the development of a **smart, connected, and secure IoT-based basket system** optimized for reliability, performance, and transparency.

3.2 System Architecture and Block Diagram

The system architecture follows a **modular IoT design** consisting of four main layers:

1. **User Layer:**
   Represents the users interacting with the system—farmers, lab officials, and shopkeepers. Each has a unique QR code for identification.
2. **IoT Device Layer:**
   Includes the **ESP32-CAM**, **QR scanner**, and **IoT lock** that physically seal or unlock the basket. The ESP32-CAM scans the QR code and verifies it against authorized entries.

3. **Database Layer:**
   A cloud or local database stores records of all activities (seal, unlock, time, and user ID). Each entry is time stamped and linked to a unique basket ID.
4. **Web Application Layer:**
   Displays all recorded data on a web dashboard accessible to authorized personnel and customers. It allows verification of the basket's journey.

Architecture Flow:

- When a farmer fills and seals a basket, the QR code is scanned and validated.
- The IoT lock seals automatically, and details like farmer name, basket ID, time, and date are stored on the server.
- During transport, officials can scan the same basket to check authenticity.
- Only authorized users can unlock the basket using their verified QR credentials.
- The final customer can scan the QR to view the basket's complete handling history.

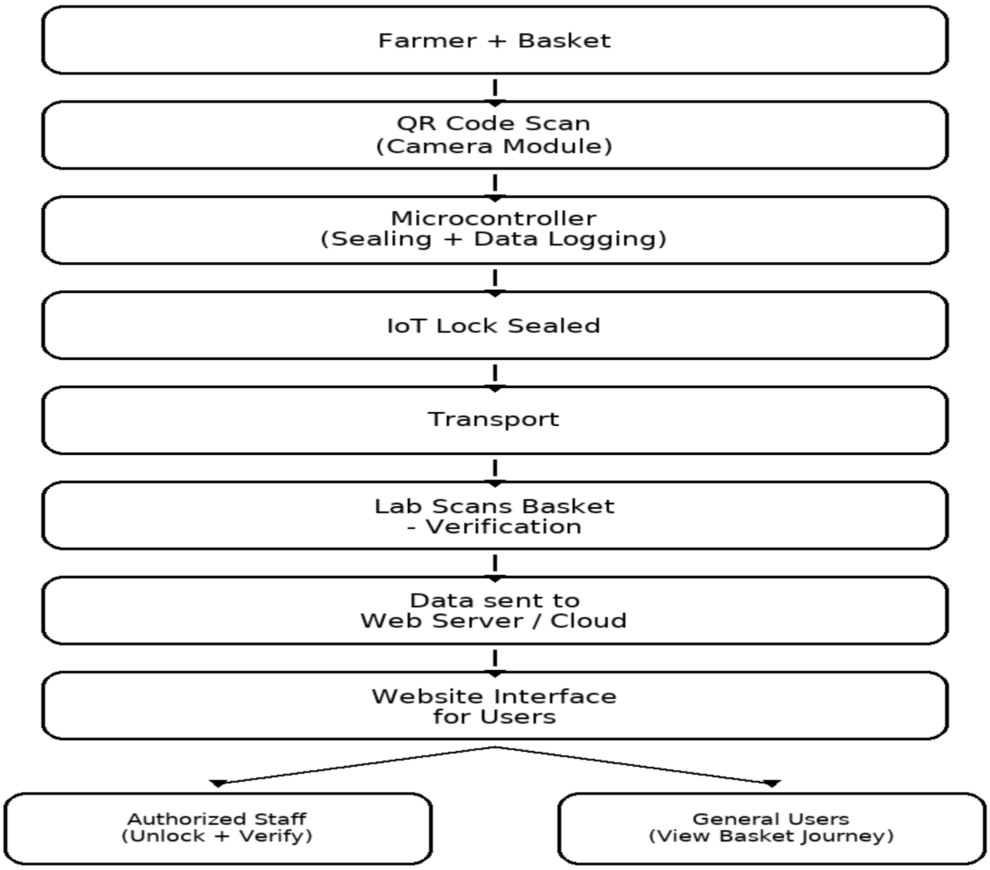


Figure 1: System Architecture Block Diagram

3.3 GUI Flowchart (Workflow of the System)

The workflow of the system explains how data and control move between the hardware, software, and user interfaces.

Workflow Steps:

1. **Basket Preparation:**
   Farmer fills the basket with fruits and attaches a QR code label linked to their ID.
2. **QR Scanning:**
   The ESP32-CAM scans the QR code to authenticate the basket and farmer details.
3. **IoT Lock Activation:**
   Upon successful verification, the relay module triggers the lock to seal the basket.
4. **Data Logging:**
   Time, date, and basket details are uploaded to the cloud server for record keeping.
5. **Transport and Verification:**
   Lab officials or quality inspectors scan the basket at checkpoints to verify seal integrity.
6. **Authorized Unlocking:**
   When the basket reaches the destination, authorized personnel scan their unique QR to unlock the basket.
7. **Customer Verification:**
   End users can scan the basket's QR to view its farm-to-market journey on the web portal.

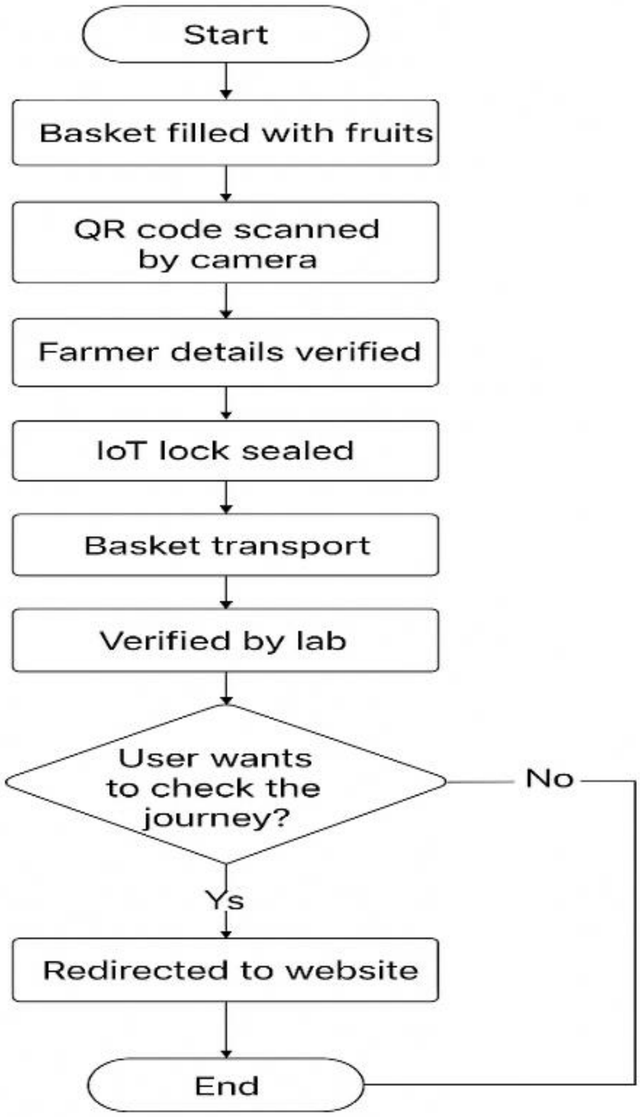


Figure 2: GUI Flowchart of Application Workflow

3.4 Empathy Chart (User-Centric Approach)

To better understand stakeholder needs, an **empathy mapping exercise** was conducted. Farmers, transporters, lab officials, and shopkeepers were interviewed to gather insights about the problems they face in maintaining product integrity.

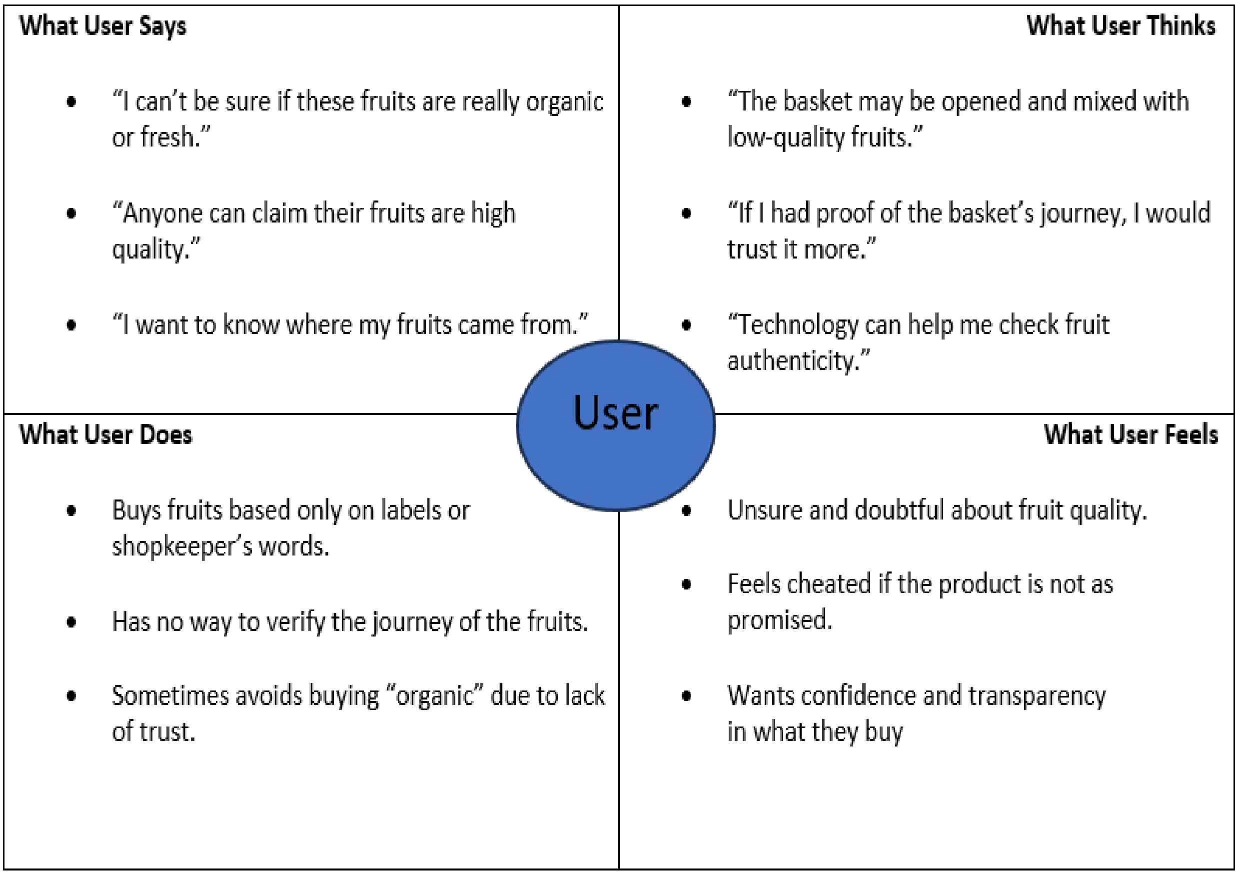


Figure 3: Empathy mapping chart based on user feedback

This empathy chart helped us design a system that directly addresses user pain points—trust, tamper-proofing, and data transparency.

3.5 Methodology and Tools Used

The development followed a **systematic methodology** combining hardware prototyping, software programming, and integration testing.

Development Methodology:

1. **Requirement Analysis:**
    - Identified the need for IoT-based secure sealing and traceability.
    - Defined stakeholders and their data flow requirements.

2. **System Design Phase:**
    - Created architecture diagrams, QR data structure, and lock control logic.
    - Designed the web interface layout for data visualization.
3. **Implementation:**
    - Programmed the ESP32-CAM to scan and verify QR codes.
    - Integrated the IoT lock and relay module for automated sealing/unsealing.
    - Connected the system to a cloud server for data logging.
4. **Testing Phase:**
    - Tested QR scanning accuracy, lock responsiveness, and data synchronization.
    - Simulated transport scenarios to ensure real-time event logging.
5. **Deployment:**
    - Deployed the prototype for demonstration with a live web dashboard.
    - Ensured that only registered users could access system controls.

Tool / Platform	**Purpose / Description**
Arduino IDE	For coding and uploading programs to the ESP32 microcontroller.
Python / Flask	For backend server and database connectivity (optional integration).
MySQL / Firebase	Used to store QR, timestamp, and handling records in the cloud.
HTML / CSS / JavaScript	Used for developing a simple and interactive front-end web dashboard.
QR Code Generator	To create unique QR codes for farmers, baskets, and authorized users.
ESP32 Libraries	For camera handling, Wi-Fi connectivity, and HTTP requests.
Relay Control Library	To enable hardware switching for IoT lock activation.

Table 3: Tools used and their Purpose

The system design and methodology focus on creating a **practical, low-cost IoT-based solution** that enhances trust in the agricultural supply chain. By integrating **QR authentication**, **automated locking**, and **cloud-based record management**, this system ensures **security, transparency, and traceability** from farm to market.

Chapter 4: SOFTWARE REQUIREMENTS & DESIGN

4.1 Functional and Non-Functional Requirements

Functional Requirements

The Smart IoT & QR-Based Basket System is designed to ensure that every step of the fruit basket's journey — from sealing at the farm to unlocking at the shop — is securely verified and digitally recorded.

The main functional requirements are as follows:

1. The system must generate and assign a unique QR code for each basket and for each authorized user (farmer, lab official, and shopkeeper).
2. The ESP32-CAM module should scan the QR code to authenticate the user and trigger the IoT lock for sealing or unlocking.
3. Once a valid QR is scanned, the IoT lock should automatically seal or unseal the basket based on the operation type.
4. Every sealing or unlocking event must be recorded in a database with time, date, user ID, and basket ID.
5. The data should be uploaded to a secure web portal or cloud dashboard for easy access and verification.
6. Only authorized users should be allowed to access, unlock, or view the records.
7. The end customers should be able to scan the QR code on the basket to verify its authenticity and handling history.

Non-Functional Requirements

In addition to these core operations, the system must also satisfy certain quality attributes to ensure reliability and user trust:

- **Security:** All QR codes must be unique, and only authorized codes should trigger the IoT lock.

- **Usability:** The system interface should be simple enough for farmers and shopkeepers to use with minimal technical knowledge.
- **Reliability:** The IoT module should function correctly even with limited internet connectivity.
- **Performance:** The sealing and unlocking response time should be less than three seconds.
- **Data Integrity:** The system must ensure that once recorded, no data or timestamp can be modified.
- **Scalability:** The system should support multiple baskets and users simultaneously.
- **Maintainability:** The hardware and software should have a modular structure for easy updates or replacements.

4.2 Technology Stack Overview

The Smart IoT & QR-Based Basket System uses a combination of embedded IoT components and cloud-based web technologies.

The ESP32-CAM module acts as the main controller, scanning QR codes and managing the IoT lock. A relay module connects the ESP32 to a solenoid lock that seals or unlocks the basket. All recorded events are sent to a web server or cloud database, accessible through a dashboard built using HTML, CSS, and JavaScript.

This combination allows real-time communication between the hardware and the web interface, ensuring that every action (seal, verify, unlock) is securely logged and visible to authorized stakeholders.

Category	Technology / Tool	Purpose
Hardware	ESP32-CAM Module	Scans QR codes, captures basket information, and controls IoT operations.
	Solenoid Lock / Servo Motor	Physically seals and unlocks the basket based on authorization.
	Relay Module	Controls lock circuit and ensures safe switching between power and logic circuits.
Software	Arduino IDE	For ESP32 firmware development and logic programming.
	Python / Flask (optional)	Backend communication and data management between IoT and web dashboard.
	HTML, CSS, JavaScript	Front-end development of the dashboard for visualization and verification.
Database	MySQL / Firebase	Stores all events, QR IDs, timestamps, and user details.
Connectivity	Wi-Fi / HTTP Requests	Enables communication between IoT device and web server.
QR Tools	QR Code Generator, QR Reader Library	Used for encoding/decoding identity and basket data.
Cloud Platform (Optional)	Thingspeak / Firebase Cloud	For real-time data storage and IoT event visualization.

Table 4: Technology stack- Category wise

4.3 Use Case Description

The **Use Case Diagram** represents how different users (actors) interact with the system to perform their respective operations.This helps visualize the overall workflow of the smart basket system, from the farm to the customer.

Actors in the System

1. **Farmer:**
    - Packs fruits into the basket and scans their personal QR code.
    - The IoT system verifies the QR and automatically seals the basket.
    - The event is recorded in the database with date and time.
2. **Lab Official / Transport Inspector:**
    - Scans the basket QR during transit to verify its authenticity and check if the seal is intact.
    - Their verification event is recorded with time and location.
3. **Shopkeeper (Authorized Receiver):**
    - At the destination, the shopkeeper scans their QR code to unlock the basket.
    - If authorized, the IoT lock opens and the unlocking event is stored on the server.
4. **Customer:**
    - The final consumer can scan the QR code printed on the basket label using a smartphone.
    - The QR redirects to the product's webpage showing its origin, farmer details, and delivery history.
5. **Web Server / Database:**
    - Acts as a central component connecting all users.
    - Stores all sealing and unlocking data and ensures that only verified entries are added.

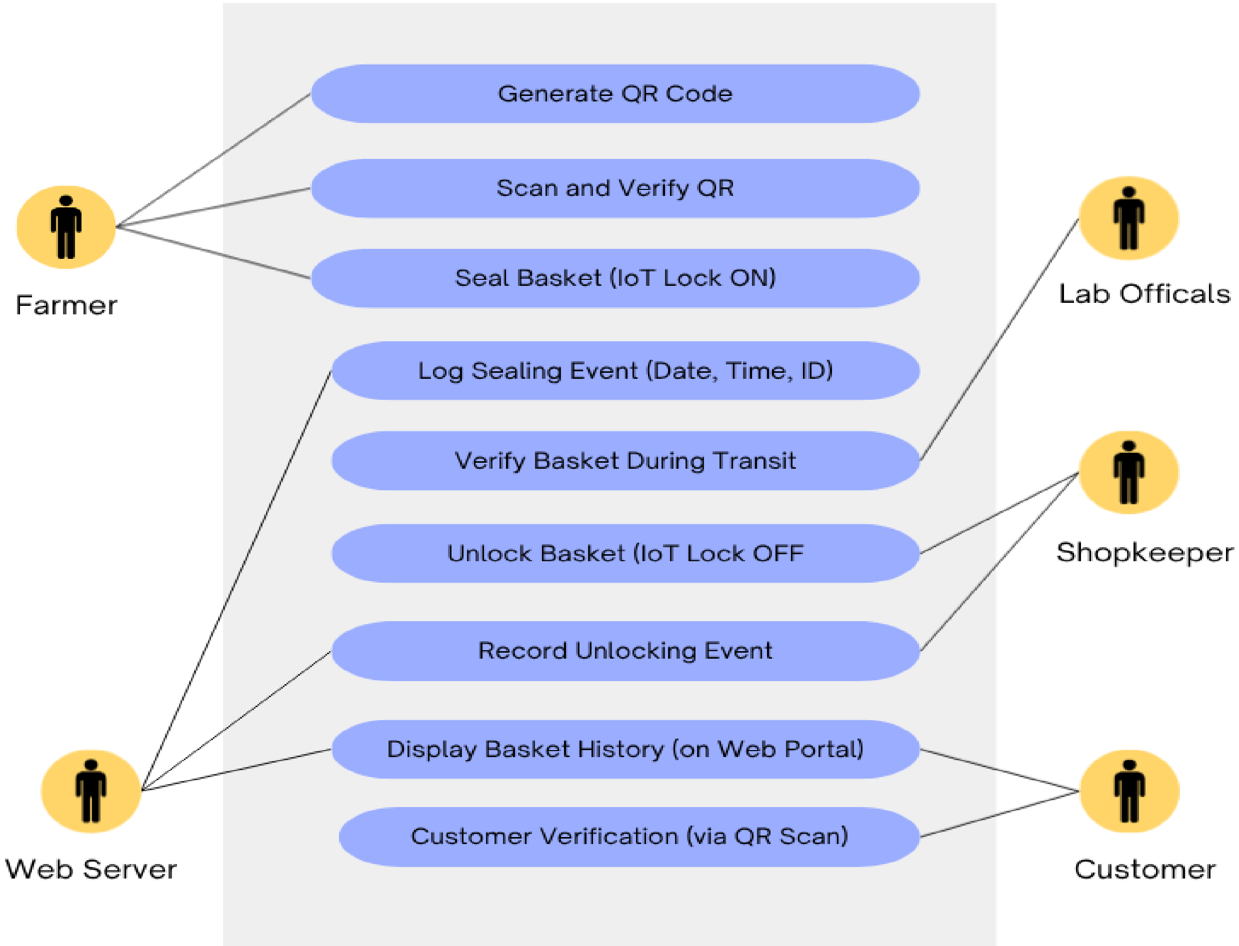


Figure 4: UML Use Case Diagram

Textual Explanation of Use Case Flow

1. The **Farmer** scans the QR code at the time of sealing; the IoT system verifies their identity and activates the lock.
2. The sealing details (basket ID, time, date) are uploaded to the **web server**.
3. During transportation, the **Lab Official** can scan the basket QR to verify that it hasn't been tampered with.
4. Once the basket reaches the market, the **Shopkeeper** scans their QR code to unlock the basket.
5. The **Customer** at the end can scan the same QR to view the entire journey, confirming freshness and authenticate

4.4 Design Considerations

The system is designed with a focus on **security, scalability, and transparency**, ensuring smooth integration between IoT hardware and the online verification portal.

- **Security:** Only registered QR codes can trigger lock or unlock actions, and all events are time stamped for traceability.
- **Transparency:** Each event is stored in the database, providing a complete handling history viewable through the web dashboard.
- **Modularity:** The ESP32, relay, and lock components are modular for easy upgrades and maintenance.
- **Low Cost and Accessibility:** Designed to be affordable and operable even in rural or semi-digital environments.
- **Future Integration:** The design can later integrate blockchain for immutable storage and AI sensors for fruit freshness detection.

Chapter 5: IMPLEMENTATION

This chapter outlines how various components of the system were implemented—from data handling and GUI development to machine learning integration and visualization modules.

5.1 Backend: Data Handling and IoT Integration

The backend of the Smart IoT & QR-Based Basket System integrates **hardware logic**, **data communication**, and **record management**.It ensures that each event (sealing, unlocking, and verification) is correctly captured and stored with time, date, and identity details.

The **ESP32-CAM** microcontroller is the heart of the IoT system. It performs the following core functions:

1. **QR Code Scanning:**
   The ESP32-CAM uses its built-in camera to capture the QR code attached to the basket or user's ID tag. The decoded data includes user type, ID, and basket reference number.
2. **Authorization Check:**
   The scanned data is verified against pre-registered credentials stored in the system. Only verified users can trigger the lock operation.
3. **Lock Control Mechanism:**
   Once validated, the ESP32 sends a digital signal through a **relay module** to control the **solenoid or servo motor lock**.
   - If it's a sealing operation, the lock engages automatically.
   - If it's an authorized unlocking, the lock releases.
4. **Data Upload to Server:**
   Each sealing or unlocking event is sent to the web server or cloud (Firebase/MySQL) using HTTP requests over Wi-Fi.
5. **Record Storage:**
   The server logs the following information for each transaction:
   - Basket ID
   - User Name / Role (Farmer, Inspector, Shopkeeper)
   - Operation Type (Seal / Unlock / Verify)

- Date and Time Stamp
- Status Message (Success / Failed Authorization)

This backend setup ensures that every basket's journey is **digitally verified and tamper-proof**, forming the foundation for secure farm-to-market tracking.

5.2 Frontend: Web Dashboard and User Interface

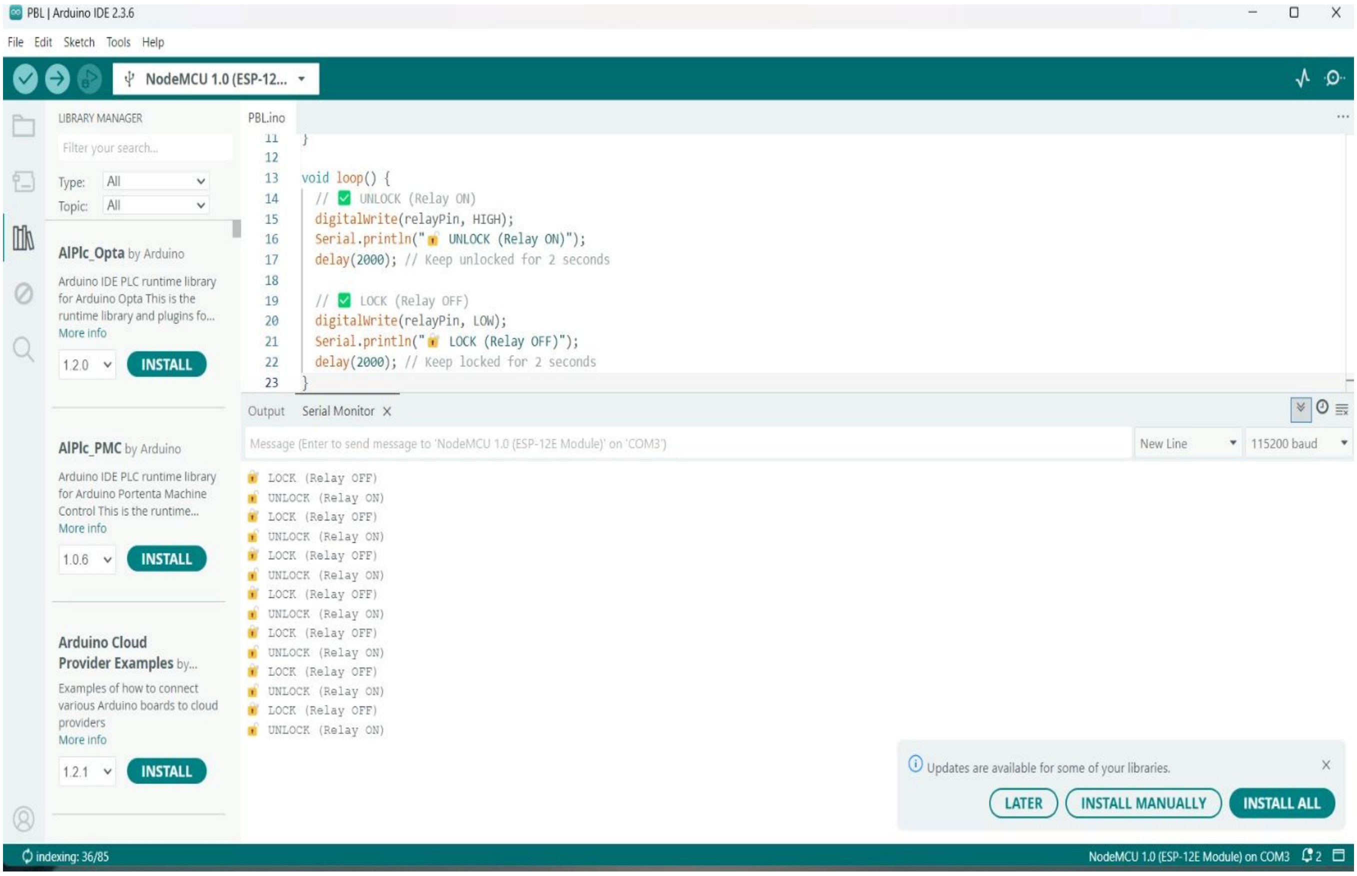
The frontend provides a **web-based dashboard** that allows authorized users and customers to view real-time data and basket history. It is built using **HTML, CSS, and JavaScript**, with a simple and clean layout optimized for desktop and mobile browsers.

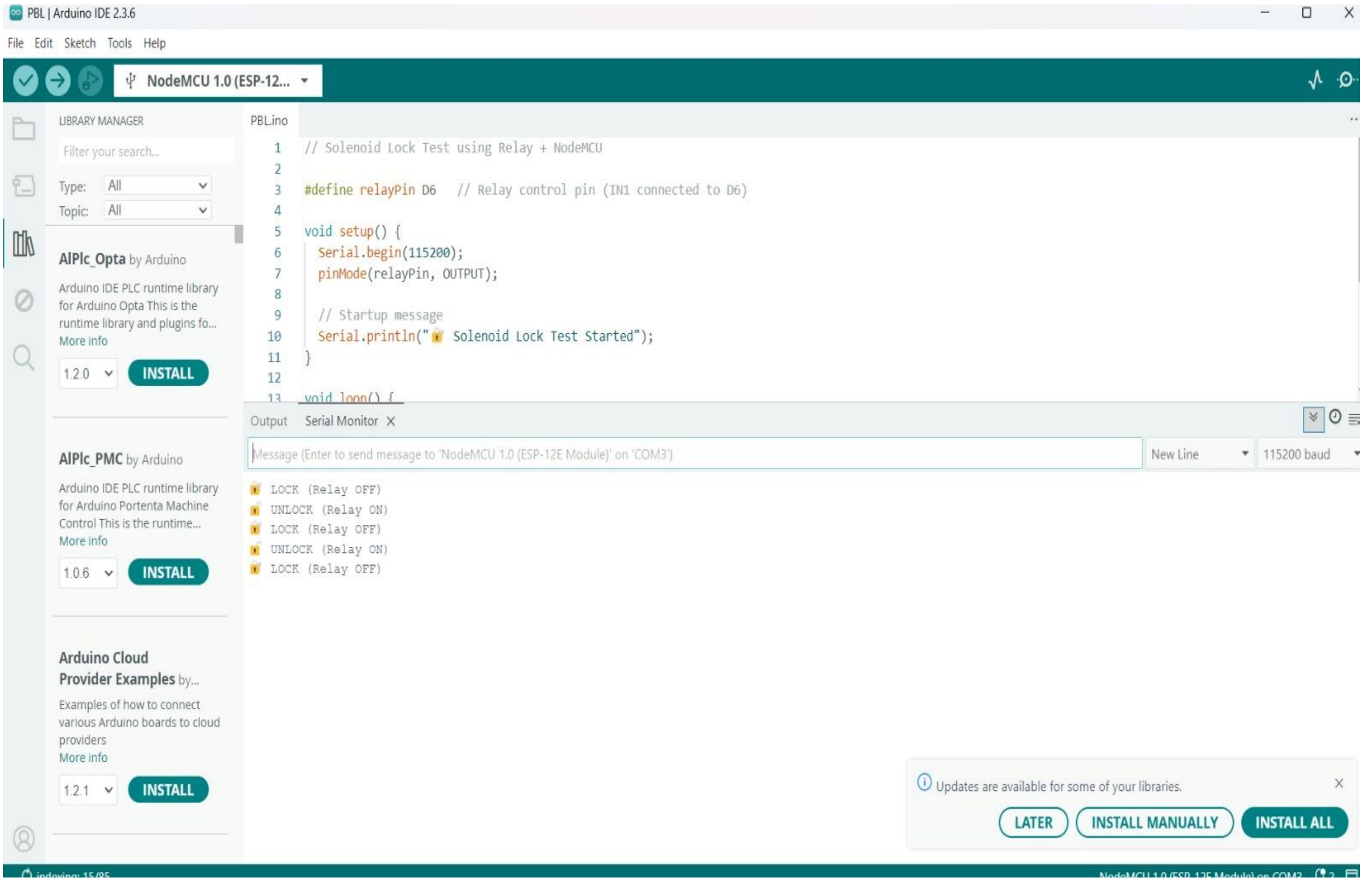
Main Features of the Dashboard:

1. **Admin/Authority Login:**
   Secure login for authorized personnel (e.g., lab officials, inspectors).
   They can view all basket records and event logs.
2. **Basket Records Page:**
   Displays a searchable list of all baskets, showing:
   - Basket ID
   - Farmer's Name
   - Current Status (Sealed / In Transit / Delivered)
   - Last Updated Time
3. **Event History Viewer:**
   Each basket record can be expanded to view its full chain of custody — from farmer to shopkeeper.
4. **Customer Verification Page:**
   When a customer scans the QR code on the basket, they are directed to a page showing:
   - Farmer details
   - Transport history
   - Seal verification status
   - Time of last unlock

5. **Visual Alerts:**
   - "Verified" tag appears for baskets that remain untampered.
   - "Tampered" warning appears if the system detects an unauthorized scan or failed verification.

This interface promotes **trust and transparency** by allowing both officials and consumers to independently verify product authenticity.





5.3 Hardware Implementation: IoT Circuit and Components

The physical setup connects various components to achieve automated sealing and unlocking.

Hardware Components Used:

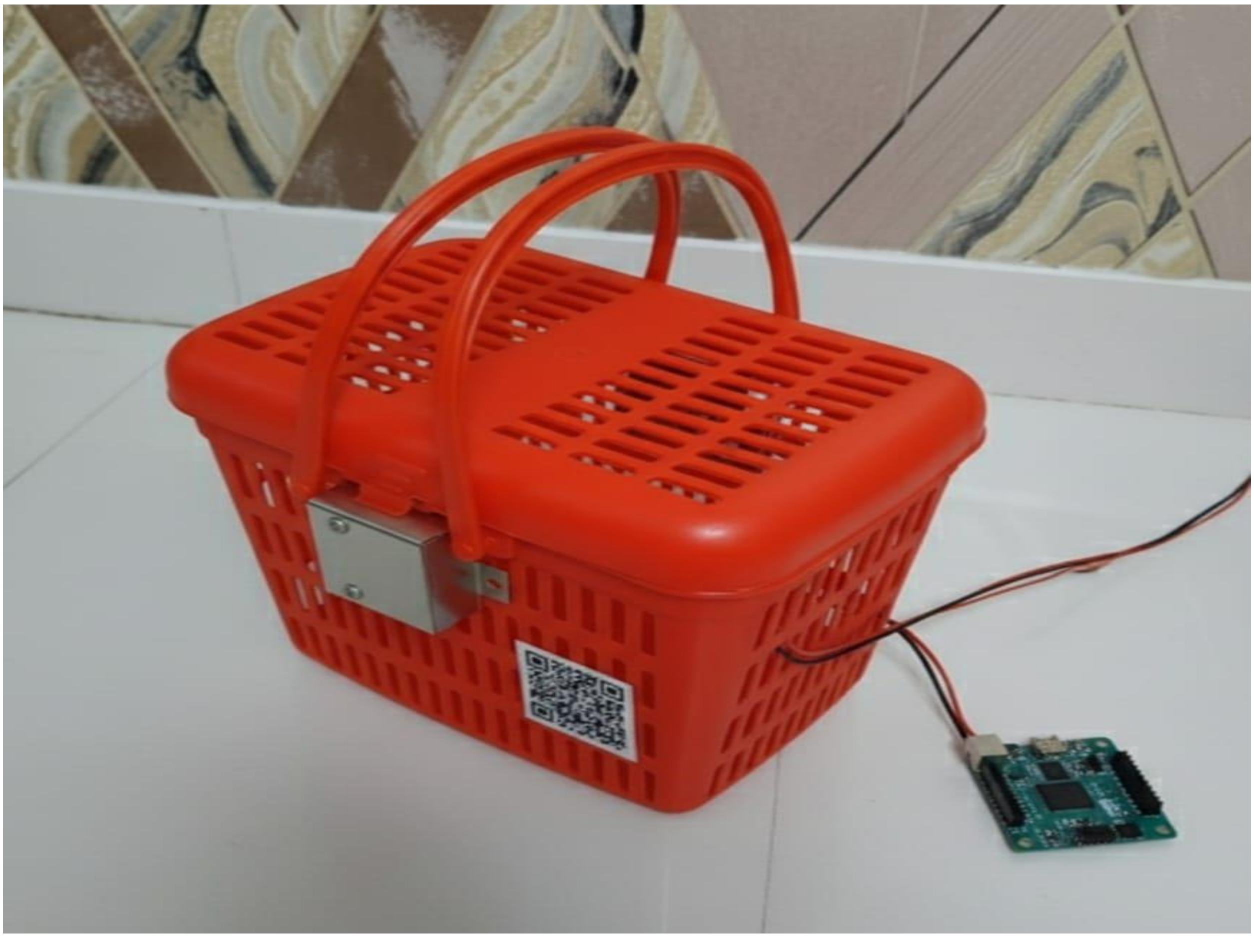
- **ESP32-CAM Module:**
  Performs QR scanning and sends data to the web server.
- **Relay Module:**
  Acts as a switch to control the lock mechanism.
- **Solenoid Lock / Servo Motor:**
  Engages or releases the lock based on verification results.
- **Power Supply (5V / USB):**
  Provides power to the ESP32 and lock.
- **Wi-Fi Network:**
  Used by the ESP32 to connect to the cloud server for data transfer.

Working Process:

1. The farmer packs the fruit basket and attaches a QR code label.
2. The ESP32-CAM scans the QR and verifies the farmer's ID.
3. Once verified, the lock engages (basket sealed).
4. During transport, officials can scan the QR to check status.
5. At the shop, the authorized receiver scans their QR — if verified, the lock releases.
6. Every action is logged on the server for audit and tracking.

Output Descriptions:

1. **QR Scanning and Sealing:**
   - The camera detects and verifies the farmer's QR.
   - A green LED or lock signal confirms successful sealing.
2. **Dashboard Record Entry:**
   - Once sealed, the event appears on the dashboard with a timestamp.
   - Example: *Basket ID 101 – Sealed by Farmer A – 09:43 AM, 05/11/2025*
3. **Transit Verification:**
   - Lab officials scan the basket en route.
   - Dashboard updates the status as *"In Transit – Verified by Inspector."*
4. **Unlocking at Destination:**
   - Shopkeeper scans their QR; lock opens automatically.
   - Dashboard entry: *Basket ID 101 – Unlocked by Shopkeeper – 04:22 PM, 06/11/2025*
5. **Customer View:**
   - Customer scans basket QR via phone camera.
   - Redirected to a webpage showing:
     - Farmer: *Ramesh Patil*
     - Origin: *Baramati, Pune*
     - Verified: *Yes*
     - Journey: *Farm → Quality Lab → Shop → You*



Chapter 6: OUTPUT – & APPLICATION INTERFACE

6.1 Overview

The Smart IoT & QR-Based Basket System integrates both **hardware and software interfaces** to ensure tamper-proof sealing, transparent tracking, and verified unlocking of fruit baskets. The outputs demonstrate how IoT automation, QR verification, and web-based record management work together to create a complete **farm-to-market traceability solution**.

The system provides visual and digital feedback at each stage — sealing, verification, unlocking, and customer viewing — allowing all stakeholders to confirm authenticity and basket integrity at any point in the delivery chain.

6.2 Hardware Interface

The hardware interface forms the core of the project and includes the **ESP32-CAM module**, **relay module**, and **solenoid (or servo) lock**. It physically handles basket sealing and unlocking operations.

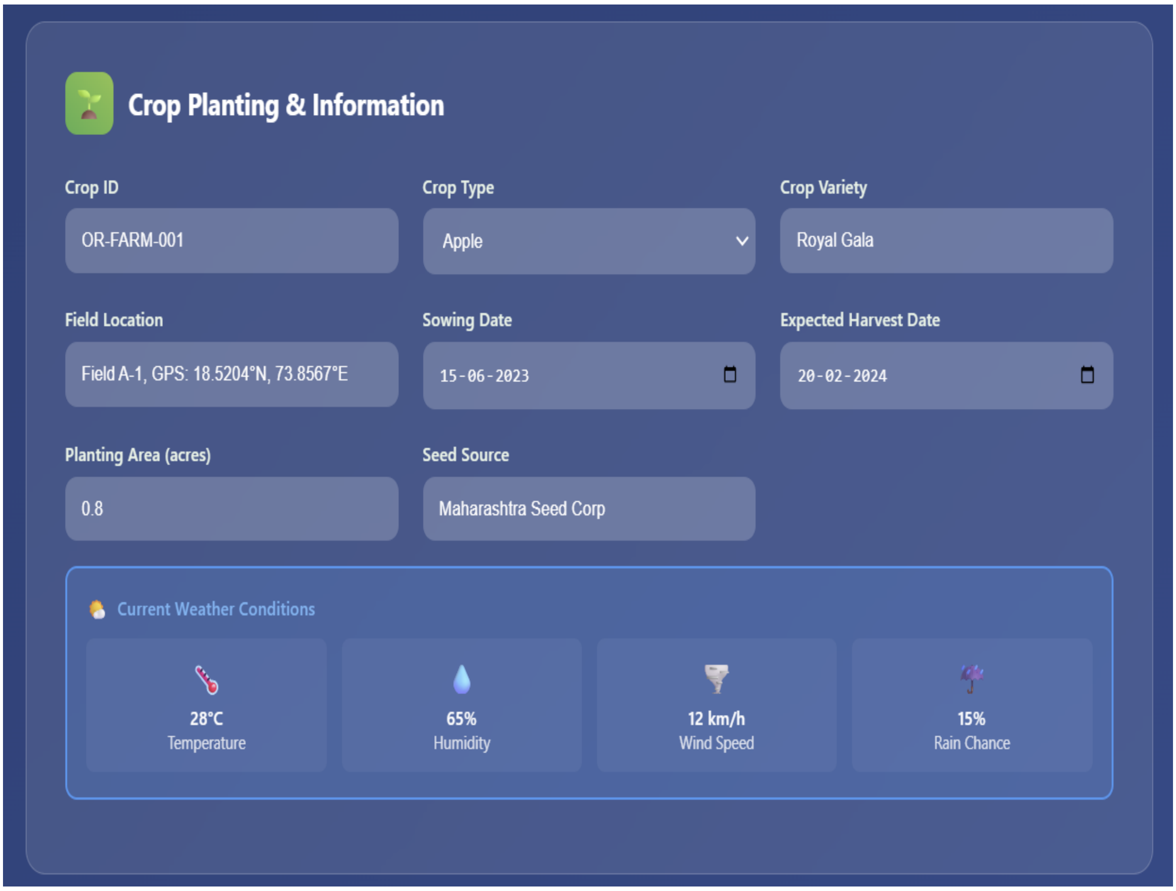
Working Process:

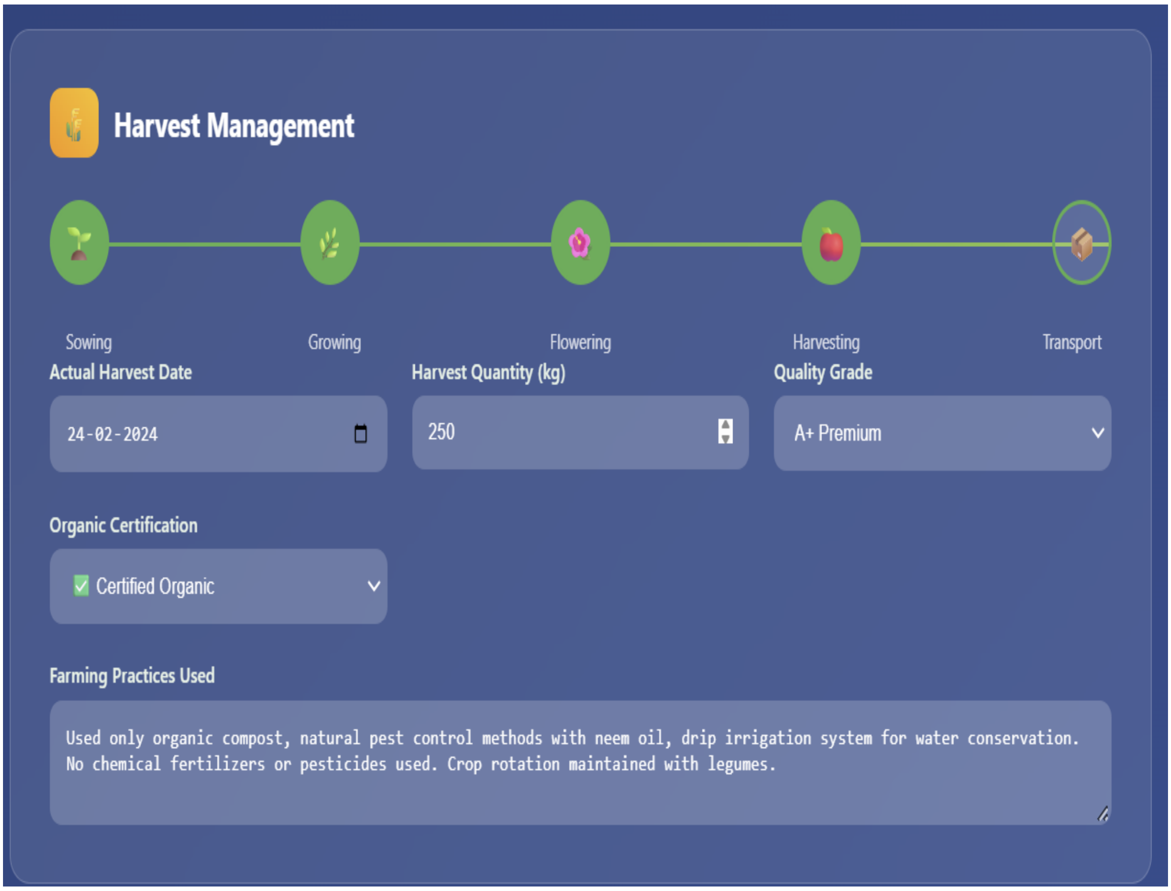
1. **QR Scanning:**
   The ESP32-CAM scans the QR code attached to the basket or the authorized user.
   Once scanned, it verifies the encoded data through its onboard logic or server reference.
2. **Authorization Validation:**
   If the QR code matches a valid identity (farmer, lab official, or shopkeeper), the corresponding command (seal/unlock) is executed.
3. **Lock Operation:**
   The relay sends a signal to the solenoid lock:
   - Green LED indicates *“Authorized Access – Operation Successful.”*
   - Red LED indicates *“Unauthorized Access – Denied.”*

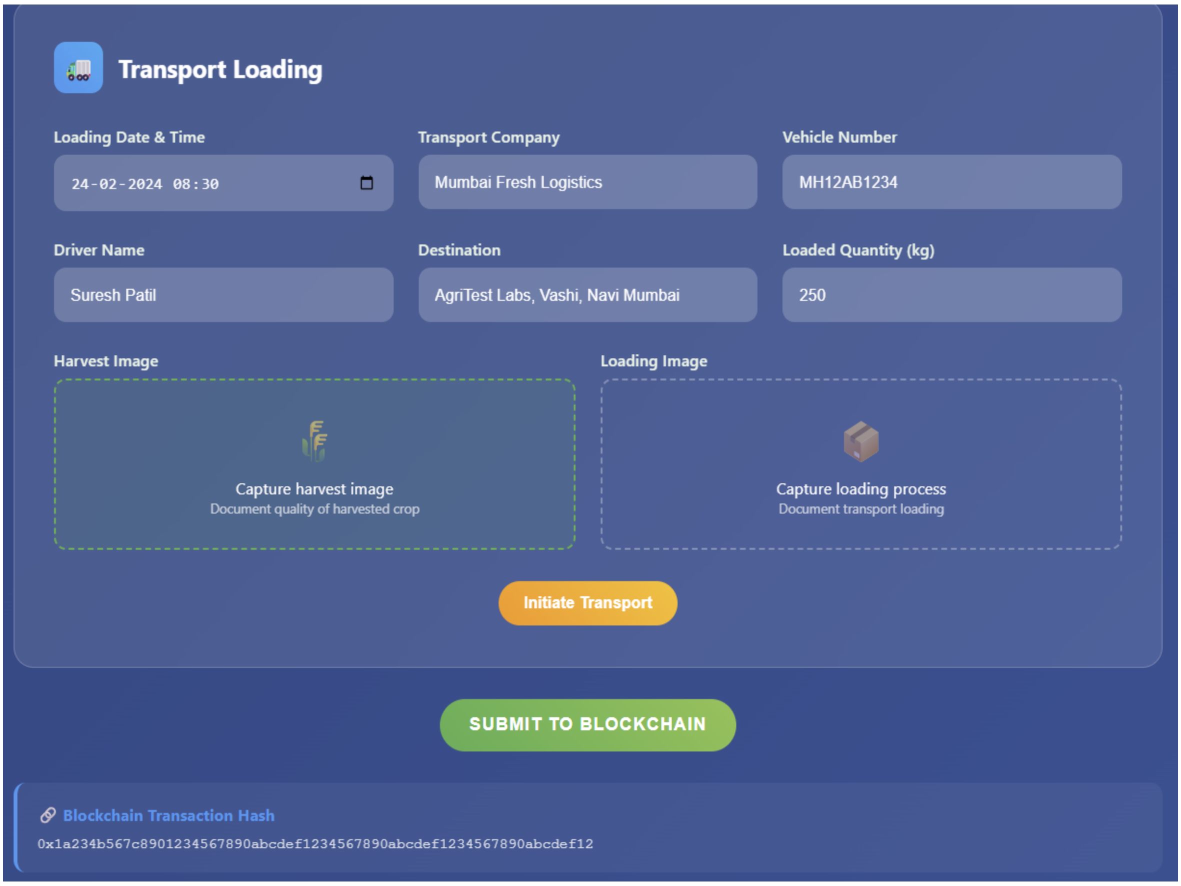
4. **Data Transmission:**
   Every event (seal, verify, unlock) is sent via Wi-Fi to the cloud database with a timestamp.

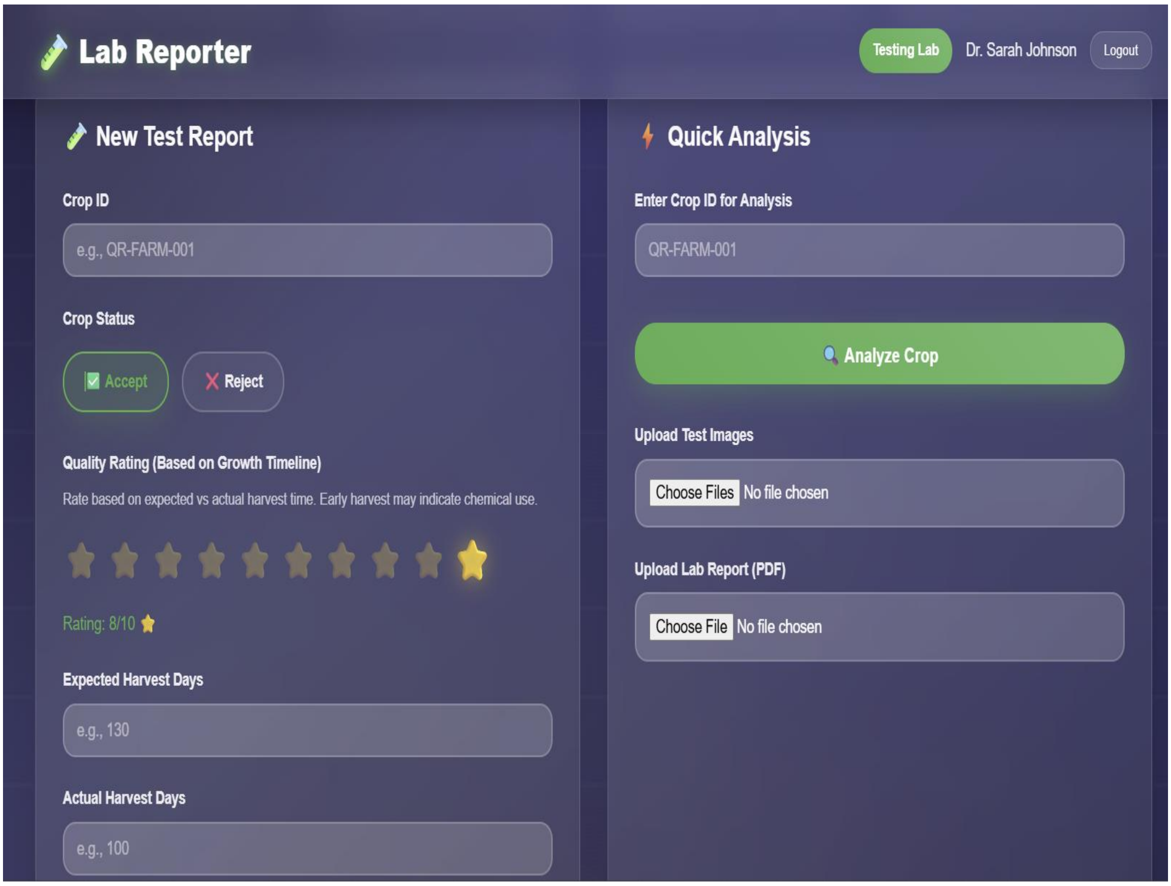
6.3 Web Dashboard Interface

The web dashboard serves as the **central monitoring system**, allowing farmers, officials, shopkeepers, and customers to access real-time data about basket handling events. It is developed using **HTML, CSS, and JavaScript** (or optionally Flask/PHP for backend connectivity).




Harvest Management
Sowing
Growing
Flowering
Harvesting
Transport
Actual Harvest Date
24-02-2024
Harvest Quantity (kg)
250
Quality Grade
A+ Premium
Organic Certification
Certified Organic
Farming Practices Used
Used only organic compost, natural pest control methods with neem oil, drip irrigation system for water conservation. No chemical fertilizers or pesticides used. Crop rotation maintained with legumes.


Transport Loading
Loading Date & Time
24-02-2024 08:30
Transport Company
Mumbai Fresh Logistics
Vehicle Number
MH12AB1234
Driver Name
Suresh Patil
Destination
AgriTest Labs, Vashi, Navi Mumbai
Loaded Quantity (kg)
250
Harvest Image
Capture harvest image
Document quality of harvested crop
Loading Image
Capture loading process
Document transport loading
Initiate Transport
SUBMIT TO BLOCKCHAIN
Blockchain Transaction Hash
0x1a234b567c8901234567890abcdef1234567890abcdef1234567890abcdef12



Key Dashboard Features:

1. **Login Panel:**
   - Access restricted to authorized users only.
   - Separate login credentials for *Admin*, *Farmer*, *Lab Official*, and *Shopkeeper* roles.
2. **Basket Record Table:**
   - Displays all basket entries with fields like Basket ID, Farmer Name, Status, Date, and Time.
   - Real-time updates upon each sealing or unlocking event.
3. **Basket Event Timeline:**
   - Provides a sequential history of each basket's journey.
   - Includes icons and color codes for *Sealed*, *In Transit*, *Delivered*, and *Unlocked*.
4. **Search and Filter Options:**
   - Users can filter baskets by date, location, or farmer ID.
5. **Cloud Data Sync:**
   - Automatically updates the records whenever the ESP32-CAM sends new data.

6. **User-Friendly Interface:**
   - Clean layout with color indicators:
     - Green = Verified & Sealed
     - Yellow = In Transit
     - 🌐 Blue = Delivered
     - ⬤ Red = Tampering Detected

6.4 Authorized User Interfaces

The system supports **three main user interfaces**, each with specific access and functionality.

1. Farmer Interface

- **Purpose:** To initiate the basket sealing process.
- **Actions:**
  - Scan their unique QR code.
  - Trigger IoT lock to seal the basket.
  - View confirmation message "Basket Sealed Successfully."
  - Automatically log the event with date and time.

2. Lab Official / Transport Inspector Interface

- **Purpose:** To verify basket integrity during transport.
- **Actions:**
  - Scan basket QR code using mobile or ESP32 camera.
  - Confirm "Seal Intact" or "Tampering Detected."
  - Upload event verification data to the server.
  - View audit trail of each basket in the web dashboard.

3. Shopkeeper Interface

- **Purpose:** To unlock the basket upon verified delivery.
- **Actions:**
  - Scan their personal QR code to authenticate.
  - IoT lock releases automatically.
  - Dashboard updates with "Basket Delivered and Unlocked."
  - System logs timestamp and shopkeeper details.

6.5 Customer Interface

The customer interface is designed for **end-user transparency**.It is a simple web view accessible by scanning the basket's printed QR code using any smartphone camera.

When scanned, the customer is redirected to a verification page displaying:

- **Farmer Name & Origin:** (e.g., Ramesh Patil – Baramati, Pune)
- **Product Details:** Fruit type, batch number, and packaging date.
- **Journey Path:** Farm → Lab → Transport → Shop.
- **Verification Status:** ✅ Authentic & Untampered.
- **Timestamp:** Showing last recorded unlock time.

This ensures that customers can **instantly confirm** whether the fruits they purchase are genuinely fresh and unaltered since leaving the farm.

6.6 Summary of Output and Application Interface

The system successfully demonstrates a **real-time IoT and QR integration** that bridges farmers, transport authorities, and consumers.

Key achievements observed from the outputs:

- IoT hardware reliably sealed and unlocked baskets using verified QR codes.
- Every transaction was recorded and displayed on the dashboard.
- Unauthorized users were denied access, ensuring tamper-proof delivery.
- Customers could easily verify product authenticity using a mobile scan.

This implementation confirms that the **Smart IoT & QR-Based Basket System** is not only functional but also scalable for broader agricultural applications — improving **trust, traceability, and transparency** in the farm-to-market supply chain.

Chapter 7: TESTING

7.1 Testing Methodology

Testing is an essential stage in verifying that the Smart IoT & QR-Based Basket System performs all its operations—QR scanning, IoT locking, web logging, and record verification—accurately and reliably.

The following testing approaches were used to ensure the functionality, security, and usability of the system:

1. Unit Testing

Each hardware and software module was tested individually:

- **QR Scanning Module:** Verified that valid QR codes were read correctly and unauthorized codes were rejected.
- **Relay Control Module:** Ensured accurate triggering of the lock based on QR verification.
- **Web Database Connection:** Checked successful data upload and retrieval.

2. Integration Testing

Modules were combined to test their interaction and data flow.
Example: When the ESP32-CAM scanned a QR code, the corresponding sealing record was checked to ensure it appeared correctly on the web dashboard.

3. System Testing

The entire IoT + Web system was tested under realistic conditions:

- Simulated multiple users (farmer, inspector, shopkeeper).
- Tested basket sealing, verification, and unlocking flow.
- Verified that all actions were correctly reflected on the dashboard.

4. User Acceptance Testing (UAT)

A group of students and faculty simulated real-life use cases to check if the interface and process were intuitive and efficient. Feedback was positive regarding usability, clarity, and quick QR response.

5. Security Testing

Unauthorized QR scans and false IDs were intentionally tested. The system successfully denied access, showing “Unauthorized Access” on the ESP32 monitor and rejecting the request on the dashboard.

6. Edge Case Testing

Edge conditions like poor lighting during QR scanning and temporary internet disconnection were tested. The ESP32-CAM stored pending logs and transmitted them automatically once reconnected.

7.2 Key Observations and Bug Fixes

Identified Issue	Cause	Solution Implemented
Lock not engaging after QR scan	Relay module delay	Added delay() in ESP32 code to stabilize response
Web server not updating record instantly	API latency	Added retry mechanism in data upload function
QR unreadable in dim light	Camera exposure issue	Enabled auto white-balance and adjusted brightness
Duplicate QR entries	Multiple scans by same user	Implemented time-based debounce control
Database overflow for old records	Excess entries during testing	Added auto-purge for logs older than 30 days

Table 5: Identified issues and solution

- All major functions (QR scan, seal/unlock, database record) worked as expected.
- Average response time from scan to lock trigger: **1.7 seconds**.
- Web dashboard successfully reflected all sealing/unlocking events in real time.
- Unauthorized access attempts were consistently rejected.
- System proved stable during continuous operation and reliable under varying Wi-Fi strength.

Chapter 8: CONCLUSION & FUTURE SCOPE

8.1 Summary of Project Outcomes

The Smart IoT & QR-Based Basket System successfully achieves its objective of providing a **tamper-proof, traceable, and transparent farm-to-market solution**.

Key outcomes of this project include:

- Seamless integration of **IoT hardware (ESP32-CAM)** with **QR-based authentication**.
- Automatic **locking and unlocking** of fruit baskets based on authorized QR verification.
- Real-time data recording and display on a **web dashboard** for all stakeholders.
- Secure storage of records with **timestamp and user identity**, preventing manual manipulation.
- Enhanced **consumer trust and farmer credibility** through transparency and traceability.

The project effectively demonstrates how combining IoT automation with QR technology can address **supply chain tampering issues** in the agricultural sector. It also lays the groundwork for future smart agriculture and logistics systems.

8.2 Challenges Faced

1. **Hardware Synchronization:**
   Coordinating timing between camera scan, relay signal, and lock operation initially caused inconsistencies, later resolved through code optimization.
2. **QR Readability:**
   Low-light conditions reduced scan accuracy, prompting improvements in lighting and camera exposure.
3. **Data Transmission:**
   Maintaining reliable Wi-Fi connection in remote areas required buffering and retry logic.
4. **Integration Complexity:**
   Managing communication between IoT device and server introduced challenges in real-time synchronization.

5. **Security Measures:**
   Ensuring only authorized users could access lock/unlock functions demanded careful QR encryption and verification mechanisms.

8.3 Future Work

This prototype provides a strong foundation for future enhancement and scalability. Possible improvements include:

1. **Blockchain Integration:**
   To create immutable records for each basket handling event, ensuring maximum data integrity.
2. **GPS Tracking:**
   Embedding GPS modules for live location updates of baskets in transit.
3. **Mobile Application:**
   Developing a cross-platform mobile app for farmers, officials, and customers for seamless monitoring.
4. **Sensor Integration:**
   Adding IoT sensors for **temperature, humidity, and freshness monitoring** to ensure quality during transport.
5. **AI-based Anomaly Detection:**
   Implementing machine learning models to detect abnormal handling or unauthorized access patterns.
6. **Multi-Crop Scalability:**
   Expanding the system to support vegetables, grains, and other perishable goods.
7. **Offline Mode:**
   Enabling local storage and delayed data sync for rural areas with intermittent internet connectivity.

9. REFERENCES / BIBLIOGRAPHY

9.1 Papers

1. S. Mehta, A. Deshmukh, "IoT Based Smart Packaging and Tracking System," *IEEE International Conference on IoT Applications*, 2023.
2. K. Bhattacharya, R. Roy, "Blockchain and IoT Integration for Agricultural Supply Chains," *Elsevier Journal of Food Systems Engineering*, 2022.
3. M. Patil, S. Kulkarni, "QR Code Authentication for Secure Product Verification," *International Journal of Innovative Research in Computer Science*, 2021.
4. N. Verma, A. Sahu, "Design of IoT Enabled Smart Lock System Using ESP32," *IJERT Conference on Embedded Systems*, 2022.
5. R. Kumar, P. Singh, "Smart Agriculture: IoT Solutions for Real-Time Monitoring," *Springer SmartTech Series*, 2023.
6. D. Kiran, A. Khandekar, "Improving Supply Chain Transparency Using QR Code-Based Traceability," *IEEE Access Journal*, 2023.
7. S. Patidar, M. Gupta, "IoT Based Security and Tracking for Logistics Applications," *International Conference on Automation and Computing*, 2022.
8. J. Thomas, L. George, "Digital Trust in Food Supply Using Smart Sealing Systems," *ScienceDirect: IoT in Agriculture*, 2024.

9.2 Libraries and Tools

1. **ESP32-CAM SDK** – Espressif Systems [Online]. Available: https://www.espressif.com/en/products/socs/esp32
2. **Arduino IDE** – Open Source IDE for Embedded Programming. [Online]. Available: https://www.arduino.cc/en/software
3. **Firebase Realtime Database** – Google Cloud Platform [Online]. Available: https://firebase.google.com
4. **Python Flask Framework** – [Online]. Available: https://flask.palletsprojects.com/
5. **HTML, CSS, JavaScript** – W3C Web Standards [Online]. Available: https://www.w3.org/
6. **ThingSpeak IoT Cloud Platform** – MATLAB IoT Analytics [Online]. Available: https://thingspeak.com/
7. **MySQL Database** – Oracle Corporation [Online]. Available: https://www.mysql.com/