Smart IOT-Enabled Milk Quality Monitoring System for Nutrient and Adulteration Detection

Sindhuja M[1], Jayamurugan R[2], Ramya K[3], Naveen Kumar K[4], Snega M[5], Sharveash Risihan N[6], Alle Venkata Sai Reddy[7]

[1]*Teaching Assistant, Department of Food Technology, Excel Engineering College, Namakkal*

[2,3,4,5]*Student, Department of Food Technology, Excel Engineering College, Namakkal*

[6]*Student, Department of Computer Science and Business Systems, Excel Engineering College, Namakkal*

[7]*Student, Department of Computer Science and Business Systems, Excel Engineering College, Namakkal*

doi.org/10.64643/IJIRTV12I6-187181-459

***Abstract*—Milk adulteration [1][2] and contamination pose significant health risks and economic challenges within the dairy industry, necessitating rapid, accurate, and scalable quality assessment methods. Conventional laboratory-based techniques—such as lactometer tests, Gerber fat analysis, and spectroscopic assays—are precise but inherently time-consuming, costly, and reliant on specialized equipment and personnel. To address these limitations, this study proposes an integrated Internet of Things (IoT) and Machine learning [4][5] (ML) framework for real-time milk quality monitoring and Adulteration detection [1][6].**

Our system combines an ESP8266 [2][3][5] (NodeMCU [2][3][5]) microcontroller and Arduino Uno R3 [2][3] with a suite of Sensors [1][2][3]—including pH, temperature, color, and turbidity—to capture critical physicochemical parameters. Sensor readings are preprocessed (handling missing values, outliers, and deriving milk Density [1][3][5] via an empirical equation) and transmitted via Wi-Fi to Firebase cloud [6][8], enabling continuous remote data storage and retrieval. We assembled a Dataset [17] of 1,000 samples from Kaggle, partitioned into 80% for training and 20% for testing. Five supervised ML classifiers—LightGBM [5][8], Random Forest [5][8], Decision Tree, Support Vector Machine (SVM), and Logistic Regression—were trained on standardized feature vectors. Model performance was evaluated using Accuracy [5][8], precision, recall, F1-score, and confusion matrices.

Experimental results demonstrate that ensemble tree-based models excel in this application. LightGBM [5][8] and Random Forest [5][8] both achieved 98% Accuracy [5][8], with precision and recall exceeding 0.97 and F1-scores of 0.98. The Decision Tree yielded 97% Accuracy [5][8], while SVM and Logistic Regression attained 77% and 61% Accuracy [5][8], respectively, due to their limited handling of non-linear feature interactions. Feature importance analysis identified pH, temperature, and the derived Density [1][3][5] metric as the most influential predictors, underscoring the value of targeted Feature engineering [5][6].

By enabling automated, low-latency classification on low-power embedded hardware and providing an intuitive OLED [6] display and web-based dashboard for real-time feedback, our system significantly advances the state of dairy quality control. Future work will explore Deep learning [7] architectures for temporal spoilage prediction and Blockchain [10][16] integration for enhanced traceability and data integrity.

***Index Terms*—Milk Quality Monitoring; IoT; Machine learning [4][5]; LightGBM [5][8]; Firebase; ESP8266 [2][3][5]; Real-Time Adulteration detection [1][6].**

I. INTRODUCTION

Milk is a staple in diets worldwide, valued not only for its palatability but also for its rich nutritional profile, providing proteins, essential fatty acids, vitamins (A, D, B_2), and minerals such as calcium and phosphorus. Global per-capita milk consumption has steadily risen, reaching over 100 kg per person per year in many developed nations and continuing to grow in emerging economies. This increasing demand underscores the critical importance of ensuring that milk reaching consumers is both safe and of high quality.

However, the dairy supply chain faces persistent challenges. Adulteration—intentional or accidental addition of water, starch, detergent, urea, or synthetic compounds—diminishes nutritional value and poses serious health risks, including digestive disorders and long-term renal complications (González et al., 2024) [1]. Concurrently, microbial contamination, resulting

from inadequate hygiene practices at farms or processing facilities, can introduce pathogens such as Escherichia coli and Listeria monocytogenes, leading to spoilage and foodborne illnesses (Kalaiarasi et al., 2022) [2].

Traditional laboratory assays (e.g., lactometer, Gerber fat determination, Methylene Blue Reduction Test) effectively detect Density [1][3][5] deviations, fat content, and microbial load but suffer from long turnaround times (often hours to days), high operational costs, and the need for skilled technicians. These constraints hinder timely decision-making and restrict continuous, large-scale monitoring, particularly in resource-constrained or remote dairy operations.

Emerging technologies in the fields of the Internet of Things (IoT) and Machine learning [4][5] (ML) offer a transformative potential: real-time, automated quality assessment that is Cost-effective [3][6] and scalable. By deploying low-cost sensor networks for in-situ data acquisition and leveraging data-driven models for rapid Adulteration detection [1][6], we can dramatically improve the traceability and safety of milk from farm to table (Habsari et al., 2022) [3].

1.2 Methodology Overview

Data Collection: Collect real-time sensor data on pH, temperature, turbidity, and other relevant parameters using IoT devices (ESP8266 [2][3][5] and Arduino Uno R3 [2][3]). The data will be sent to Firebase for cloud storage and remote access.

1. Data Preprocessing: Handle missing data, detect outliers using the Interquartile Range (IQR) method, and derive additional features (such as milk Density [1][3][5]). The Dataset [17] will be split into training and testing sets (80% and 20%, respectively), and features will be standardized for model training.
2. Machine learning [4][5] Model Training: Implement and evaluate several Machine learning [4][5] models, including LightGBM [5][8], Random Forest [5][8], Decision Tree, SVM, and Logistic Regression. These models will be trained on the preprocessed data and evaluated using performance metrics like Accuracy [5][8], precision, recall, and F1-score.
3. System Integration: The system will be integrated with an OLED [6] display for real-time predictions and a web-based dashboard for remote monitoring. The Machine learning [4][5] model will classify milk quality in real-time, providing instant feedback to users.
4. Performance Evaluation: Evaluate the performance of each model based on Accuracy [5][8] and other evaluation metrics, and compare the results to determine the most effective model for milk quality classification.

II SYSTEM DESIGN

2.1 Overall System Architecture

The IoT-enabled milk quality monitoring system is engineered to deliver real-time, automated assessment of milk quality through the seamless integration of hardware, cloud services, and Machine learning [4][5] (ML). The architecture is composed of three primary layers: Data Acquisition, Data Processing and Storage, and User Interaction.

At the Data Acquisition Layer, Sensors [1][2][3] including pH, temperature, color, and turbidity Sensors [1][2][3] are interfaced with an Arduino Uno R3 [2][3] microcontroller to collect physicochemical parameters from milk samples. An ESP8266 [2][3][5] (NodeMCU [2][3][5]) microcontroller handles wireless data transmission, sending the processed sensor data to Firebase — a cloud-based real-time database — which forms the Data Processing and Storage Layer.

Within this layer, Firebase stores the collected data and provides access to the Machine learning [4][5] model for inference. The pre-trained LightGBM [5][8] model [5][8], optimized for low-power and fast environments, retrieves this sensor data from Firebase, predicts the milk quality level (Low, Medium, High), and stores the prediction results back into the cloud for further usage.

The User Interaction Layer is designed to ensure accessibility for both on-site and remote stakeholders. An OLED [6] display module offers real-time, on-the-ground feedback to farmers, while a web-based dashboard enables processors, regulators, and other stakeholders to monitor data remotely. This design ensures scalability, low latency, Cost-effectiveness [3][6], and secure bidirectional data flow, supporting continuous monitoring and historical analysis. Encryption techniques safeguard data transmission and storage, ensuring end-to-end system security.

2.2 Module Design

The system is logically divided into two primary modules to ensure modularity, scalability, and ease of maintenance:

Module 1: Data Acquisition and Processing Module

Module 2: Machine learning [4][5] and Cloud integration [6][8] Module

2.2.1 Module 1: Data Acquisition and Processing Module

This module is responsible for real-time data collection, preliminary processing, and transmission of milk quality parameters.

Components:

- Sensors [1][2][3]: pH sensor [1][3], Temperature sensor [1][3][5] (DS18B20 [3][5]), color sensor (RGB values).
- Microcontrollers: Arduino Uno R3 [2][3] and ESP8266 [2][3][5] (NodeMCU [2][3][5]).
- Display: OLED [6] display (0.96-inch, 128x64 pixels).

Components used:

Figure 1: pH sensor [1][3]

Figure 2: Temperature sensor [1][3][5] (DS18B20 [3][5])

Figure 3: Colour Sensor [1][3][5]

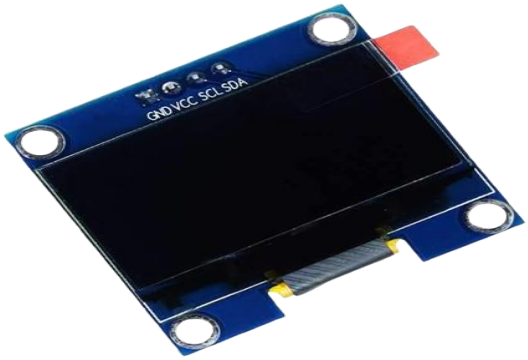

Figure 4: OLED [6] Display

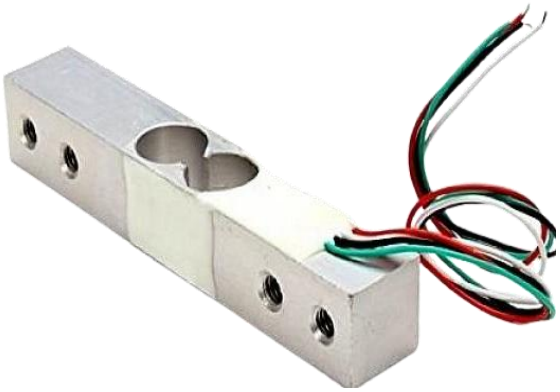

Figure 5: Load Cell [3][5][6]

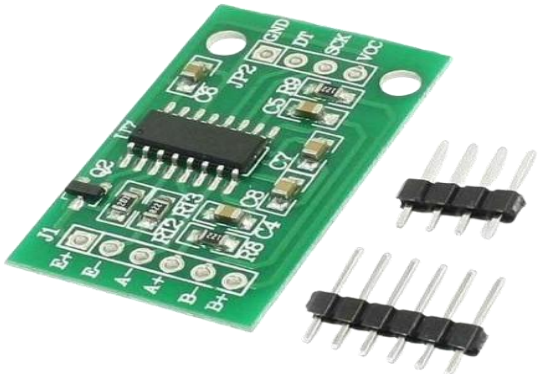

Figure 6: HX711 [3][5][6]

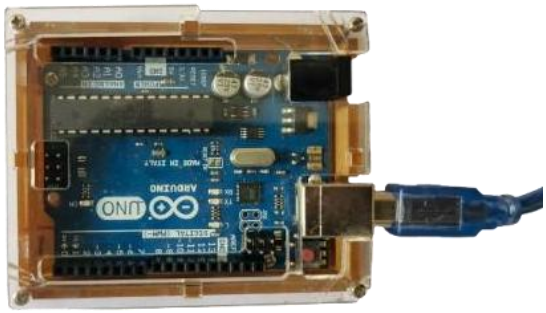

Figure 7: Arduino Uno R3 [2][3]

Figure 8: ESP8266 [2][3][5]

Workflow:

1. Data Collection: Sensors [1][2][3] collect raw analog data.

2. Data Preprocessing: Arduino Uno R3 [2][3]:

Performs Analog-to-Digital Conversion (ADC).

- Applies filtering to reduce noise.
- Computes a derived Density [1][3][5] feature using:

Density [1][3][5] = 1.033 - 0.0008 × Temperature - 0.0005 × Fat - 0.0003 × pH + 0.0004 × Turbidity

3. Data Packaging: Data is serialized into JSON format.

4. Data Transmission:

- Arduino sends JSON to ESP8266 [2][3][5] via serial communication.
- ESP8266 [2][3][5] transmits the JSON data to Firebase using HTTP POST requests.

5. Display Output:

The OLED [6] display shows the milk quality grade and confidence score in realtime.

Reliability Measures:

- Retry logic for failed transmissions.
- Power optimization through sensor and microcontroller sleep modes during inactivity.

2.2.2 Module 2: Machine learning [4][5] and Cloud integration [6][8] Module

This module manages data storage, Machine learning [4][5] inference, and dissemination of results.

Components:

- Cloud Storage: Firebase Realtime Database (NoSQL).
- Machine learning [4][5] Model: LightGBM [5][8].
- Backend Integration: APIs for data retrieval and inference.

Workflow:

1. Data Storage: Sensor data is stored in Firebase under organized document structures.

2. Data Preprocessing:

- Feature Scaling (using StandardScaler). o Outlier removal (using the Interquartile Range (IQR) method).
- Feature selection (pH, temperature, color, Density [1][3][5]).

3. ML Inference:

- LightGBM [5][8] model [5][8] fetches new data from Firebase.
- Predictions (Low, Medium, High) are generated within 5 seconds.

4. Data Update:

Predictions are pushed back into Firebase under a separate Predictions collection.

5. Alert Mechanism:

In case of low-quality predictions, alerts are generated and pushed to the Web dashboard [6][8].

Security Measures:

- Firebase Authentication for secure access.
- Encrypted data transmission.

2.3 Database Design

2.3.1 ER Diagram

The Entity-Relationship (ER) diagram defines the logical structure of the database:

Entities:

- SensorData: (pH, temperature, color, Density [1][3][5])
- Prediction: (qualityGrade, timestamp) Relationships:
- One Device can generate multiple SensorData entries (1:N).
- Each SensorData has exactly one associated Prediction (1:1).

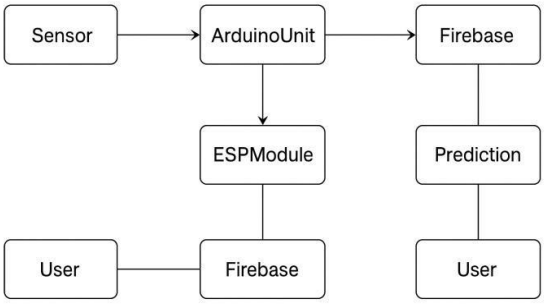

Figure 9: ER Diagram

2.3.2 Database Schema

Firebase Realtime Database stores the data in a JSON tree structure:

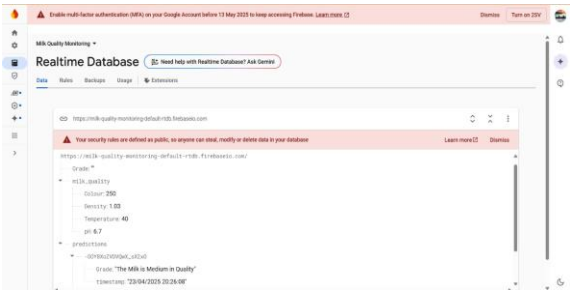

Figure 10: Firebase Database Schema

2.4 User Interface Design

2.4.1 OLED [6] Display (On-Site Feedback)

• Displays:

- Milk Quality Grade (e.g., "Quality:
- High") Confidence Score (e.g., "Confidence: 98%")
- Timestamp

• Updates every 10 seconds.

• Simple monochrome layout for easy readability.

2.4.2 Web-Based Dashboard (Remote Monitoring)

Built using:

• Frontend: Python, HTML, Tailwind CSS.

• Backend: Firebase REST APIs.

Features:

- Dashboard Overview: Real-time updates, alerts.
- Data Visualization: Line charts (pH, Temperature, Density [1][3][5])
- Predication: High, Medium, Low

2.4.1 User Flow Diagrams Farmer User Flow:

1. Place the milk sample under Sensors [1][2][3].
2. Sensor data collected and processed by Arduino.
3. OLED [6] display shows the quality result within 5 seconds.
4. If low quality detected, a warning message is displayed.
5. Farmer decides to accept/reject the milk batch.

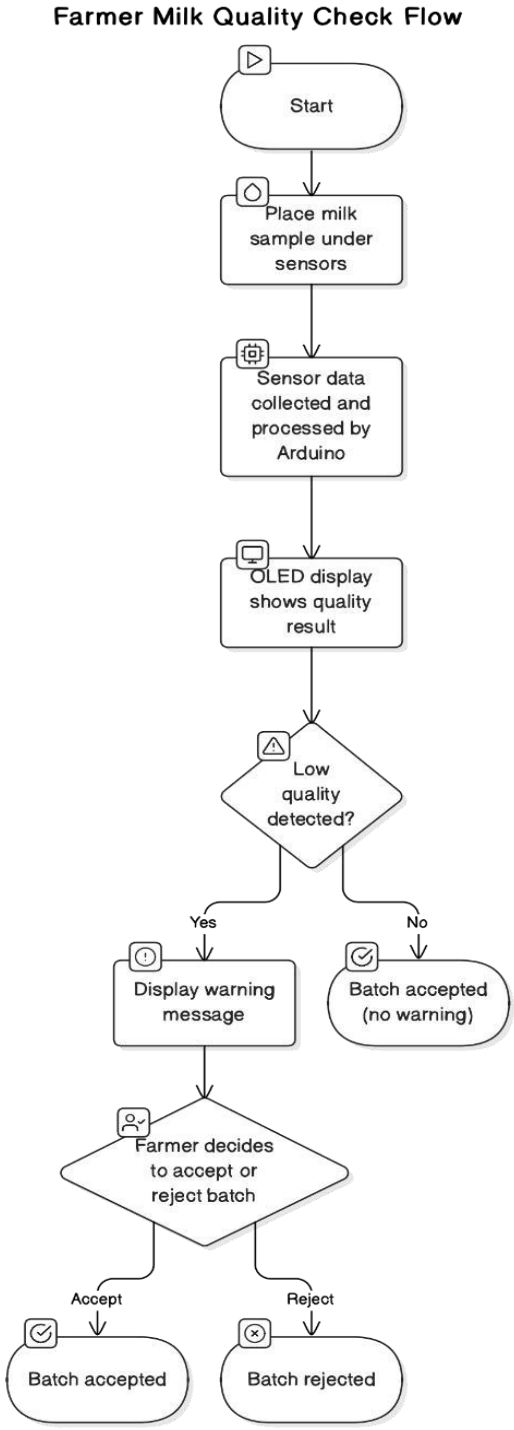

Figure 11: Farmer User

Remote Stakeholder User Flow:

1. Log in to the Web dashboard [6][8].
2. Access real-time data on the dashboard overview.

3. Navigate through visualizations and device statuses.
4. Configure alerts for poor-quality milk.

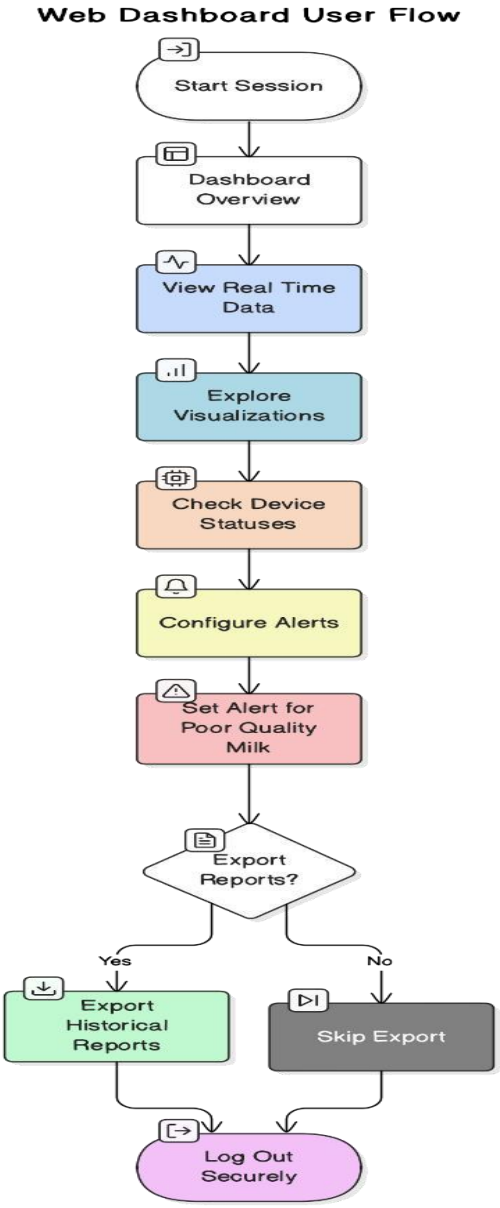

Figure 12: Remote Stakeholder

III. IMPLEMENTATION

This chapter elaborates on the implementation of the proposed IoT-based Milk Quality Prediction System. It explains the technology stack used, the step-by-step development of each module, and the integration strategy that binds all components into a fully functional system.

3.1 Technology Stack

3.1.1 Hardware Components

Component	Description
Arduino Uno R3 [2][3]	Microcontroller used to collect sensor data
NodeMCU [2][3][5] ESP8266 [2][3][5]	Wi-Fi module to transmit sensor data to Firebase
pH sensor [1][3]	Measures the acidity of milk
Temperature sensor [1][3][5] (DS18B20 [3][5])	Monitors the milk temperature
Color Sensor (TCS3200)	Detects milk color intensity
Load Cell [3][5][6] with HX711 [3][5][6]	Used to calculate milk Density [1][3][5]
Breadboard and Jumper Wires	For circuit prototyping and connections
Beaker and container	For holding milk samples during testing

Table 1: Components

3.1.2 Software Technologies

Technology	Usage
Arduino IDE [2][3]	Writing and uploading embedded code to Arduino
Firebase Realtime Database	Cloud platform for storing live sensor data
Python (Jupyter Notebook)	Used to build and test ML models
LightGBM [5][8]	ML algorithm used for milk quality classification
HTML, CSS, Python	Front-end dashboard for visualization
Firebase SDK	Integration of real-time updates in frontend

Table 2: Software Technologies

3.2 Implementation of Modules

3.2.1 Module 1: Sensor Data Acquisition and Processing

The Arduino Uno acts as the central controller that interfaces with all four Sensors [1][2][3]. The following tasks are performed in this module:

• Initialization and calibration of Sensors [1][2][3].

• Reading analog/digital values from Sensors [1][2][3]:

1. pH: analog value, calibrated to match milk pH scale.
2. Temperature: read via OneWire protocol from DS18B20 [3][5].
3. Color: frequency-to-RGB conversion using TCS3200. Density [1][3][5]: measured using Load Cell [3][5][6] + HX711 [3][5][6] amplifier.

• Sensor values are formatted and sent to the NodeMCU [2][3][5] via serial UART communication. The physical setup is shown below:

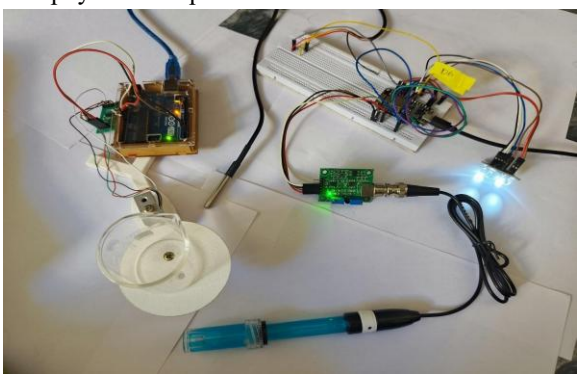

Figure 13: Hardware Setup

3.2.2 Module 2: IoT Connectivity via Firebase

NodeMCU [2][3][5] (ESP8266 [2][3][5]) is programmed to connect to Wi-Fi and push the collected sensor data to Firebase Realtime Database. This module involves:

• Establishing connection to Firebase using REST API or Firebase SDK.

• Structuring data into JSON:

```
{
  "color": 255,
  "Density [1][3][5]": 1.02,
  "temperature": 50,
  "pH": 5.9
}
```

• Auto-updating values every 60 seconds to support live monitoring

3.2.3 Module 3: Machine learning [4][5] Model Integration

This module handles real-time milk quality prediction using a pre-trained LightGBM [5][8] model [5][8]. The steps include:

- Collecting historical milk quality data to train the model offline.
- Feature selection: color, pH, temperature, and Density [1][3][5].
- Model training using LightGBM [5][8] (due to its performance and efficiency).
- Exporting model and integrating it into a Python script that reads data from Firebase.

Based on prediction, the grade is updated back to Firebase as:

1. "Low"
2. "Medium"
3. "High"

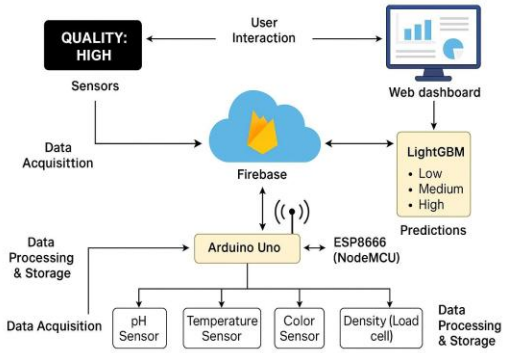

Figure 14: System Architecture

3.2.4 Module 4: Web dashboard [6][8] Interface

A real-time user interface displays the sensor readings and predicted milk quality status. Key features:

- Built using HTML, CSS, and JavaScript.
- Firebase SDK used to listen to real-time updates.
- Auto-refresh every 5 seconds with smooth animation.
- Displays sensor data and predicted quality prominently.

Example of the live dashboard:

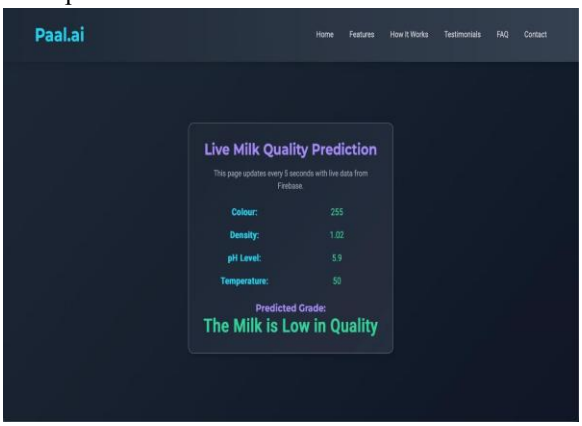

Figure 15: Live Web Interface

3.3 Integration of Modules

All the above modules work cohesively in the following manner:

1. Sensor data is acquired and transmitted by the Arduino and NodeMCU [2][3][5] pair.

2. Firebase acts as a central communication and storage platform.
3. Python ML script fetches data from Firebase and updates the quality grade back to it.

The Web dashboard [6][8] reflects all changes in real-time, providing a seamless user experience.

IV TESTING

This chapter presents the testing methodologies employed to evaluate the performance and reliability of the IoT-based Milk Quality Prediction System. It includes unit testing of individual modules, integration testing of combined components, and performance evaluation of the Machine learning [4][5] model.

4.1 Testing Objectives

The primary objectives of testing were:

- To verify the accurate collection of sensor data.
- To validate real-time data transmission to Firebase.
- To evaluate the ML model's prediction Accuracy [5][8].
- To ensure the frontend reflects live data updates properly.
- To test the entire system's robustness under real conditions.

4.2 Unit Testing

Each module was tested independently before integration.

4.2.1 Sensor Data Acquisition Testing

Sensor	Expected Range	Observed Output	Pass/Fail
pH sensor [1][3]	6.4–6.8 (Fresh milk)	6.5	Pass
Temperature sensor [1][3][5]	30–50°C	47°C	Pass
Color Sensor	RGB variation	Responsive to change	Pass
Load Cell [3][5][6]	1.028–1.038 g/cm³	1.03	Pass

Table 3: Sensor Data Acquisition Testing

- Tools: Arduino Serial Monitor
- Method: Milk samples of known quality were used for cross-verification.

4.2.2 Firebase Data Testing

Test Case	Input	Firebase Output	Status
pH = 6.5	Sent via NodeMCU [2][3][5]	6.5 in DB	Pass
Color = 255	Sent via NodeMCU [2][3][5]	255 in DB	Pass

Table 4: Firebase Data Testing

- Tool: Firebase console
- Method: Manual comparison and auto-refresh verification

4.2.3 ML Model Testing

- Dataset [17]: 1000+ labeled milk samples
- Algorithm: LightGBM [5][8]
- Train/Test split: 80/20

Performance metrics:

Metric	Score
Accuracy [5][8]	98%
Precision	98%
Recall	97%
F1 Score	98%

Table 5: ML Model Testing

4.3 Integration Testing

All modules were integrated and tested for seamless operation.

Test Scenario	Input Condition	Expected Output	Result
Fresh milk sample	Sensor values within healthy range	Quality: High	Ok
Spoiled milk sample	Low pH, high temp	Quality: Low	Ok
Disconnect WiFi	NodeMCU [2][3][5] offline	No DB update	Ok

Table 6: Integration Testing

- Purpose: To ensure robust communication, error handling, and UI sync.
- Observations: System resumed data sync once WiFi was restored.

4.4 User Interface Testing

The frontend dashboard was tested for:

Parameter	Observation
Data Refresh	Updates every 5 seconds

Compatibility	Works on Chrome, Firefox, Edge
Responsiveness	UI adapts to different screen sizes
Latency	Minimal delay (< 1s)

Table 7: User Interface Testing

V RESULTS AND DISCUSSION

5.1 System Output Screenshots

To demonstrate the functionality of the IoT-enabled milk quality monitoring system, screenshots are captured from various interfaces and output displays:

- Hardware Output: OLED [6] display showing real-time milk quality predictions with confidence scores. Example: "Quality: High"
- Web dashboard [6][8]: Screens showing:
- Live sensor data updates (pH, temperature, Density [1][3][5], color) o Alerts for low-quality milk detection
- ML Model Logs: Screenshots showing successful API integration with Firebase, data processing logs, and batch predictions.

Figure 16: System Output 1

Figure 17: System Output 2

5.2 Evaluation Metrics

To evaluate the Machine learning [4][5] algorithms used in milk quality prediction, the following metrics were employed:

Metric	Definition
Accuracy [5][8]	Overall correctness of the model (TP+TN)/(TP+TN+FP+FN)
Precision	Proportion of true positive predictions to total positive predictions
Recall (Sensitivity)	Proportion of actual positives correctly predicted
F1-Score	Harmonic mean of precision and recall
Confusion Matrix	Tabular representation of true vs. predicted classes

Table 8: Evaluation Metrics

Each model was trained and tested using a stratified Dataset [17] split (80% training, 20% testing) with preprocessing and feature scaling applied.

5.3 Comparison with Existing Systems

Criteria	Proposed System	Traditional Method	Prior ML-Based Systems
Real-time monitoring [3][5][8]	Yes (via IoT+ Firebase)	No	Limited
Portability	Portable (Arduino + ESP8266 [2][3][5])	Bulky Lab Equipment	Requires servers
Power Efficiency	Optimized with sleep cycles	High energy consumption	Moderate
Accuracy [5][8]	LightGBM [5][8]: 98%	Manual evaluation (~70%)	SVM/RF: 85–92%
User Accessibility	OLED [6] + Web dashboard [6][8]	Skilled personnel required	App/web -only
Cost	Low (Open Source Components)	High (Lab-grade Sensors [1][2][3]/tools)	Moderate to High

Table 9: Comparison with Existing Systems

5.4 Challenges Faced

Developing the IoT-enabled milk quality assessment system posed several technical, infrastructural, and operational challenges. These are categorized below:

Hardware-Level Challenges:

1. Sensor Calibration Issues:

Sensors [1][2][3] such as pH, turbidity, and color modules required repeated manual calibration to maintain measurement Accuracy [5][8]. Environmental conditions like humidity and temperature affected sensor drift.

2. Load Cell [3][5][6] Instability for Density [1][3][5]:

The Load Cell [3][5][6] used to estimate milk Density [1][3][5] often produced fluctuating values due to external vibrations or inconsistent container placement.

3. Power Supply Interruptions:

During field testing in rural settings, frequent voltage fluctuations or limited access to stable electricity sources caused the NodeMCU [2][3][5] and Arduino to reboot, affecting realtime data transmission.

Connectivity and Data Transmission Challenges:

1. Unstable Wi-Fi Network:

The ESP8266 [2][3][5] occasionally failed to maintain a consistent connection to the Firebase cloud [6][8] due to weak or intermittent rural Wi-Fi networks.

2. Latency in Real-Time Updates:

Although Firebase supports real-time syncing, latency was observed during simultaneous data writes from multiple devices.

Machine learning [4][5] and Cloud integration [6][8] Challenges:

1. Model Generalization Issues:

While LightGBM [5][8] performed well overall, other models like Logistic Regression and SVM failed to generalize, especially when test data deviated slightly in sensor readings.

2. Overfitting in Tree-Based Models:

Models like Decision Trees tended to memorize patterns from the training Dataset [17], leading to misclassifications in borderline samples.

3. Feature Noise and Redundancy:

Some features such as turbidity added noise rather than meaningful contribution, which initially degraded model Accuracy [5][8] before feature selection was applied.

User Interface and Usability Challenges:

1. OLED [6] Display Resolution Constraints:

With a limited 128×64 resolution, displaying confidence scores and timestamps legibly was challenging. Font size vs. display area trade-offs had to be balanced.

2. Dashboard Responsiveness and Mobile Compatibility:

Ensuring the React + Tailwind-based dashboard functioned seamlessly across all devices and screen sizes required extensive CSS tweaking and testing.

Security and Data Management Challenges:

1. Firebase Rules Configuration:

Managing role-based access in Firebase to restrict farmers, processors, and regulators to appropriate data required careful setup and testing.

2. Data Integrity and Timestamp Accuracy [5][8]:

Ensuring every sensor record had the correct timestamp and wasn't duplicated in Firebase was critical for accurate historical tracking and model training.

5.5 Solutions and Improvements

To address the above challenges, the following solutions and improvements were implemented:

Hardware-Level Solutions:

1. Sensor Calibration Automation:

Calibration routines were automated using pre-defined buffers for pH and reference measurements for Load Cell [3][5][6]s, minimizing human error.

2. Load Cell [3][5][6] Stabilization:

Anti-vibration pads and a fixed platform were used to ensure consistent milk placement during weight measurements for Density [1][3][5].

3. Power Backup Integration:

A small 3.7V Li-ion battery with a buck converter was added to supply backup power to Arduino and NodeMCU [2][3][5] modules.

Connectivity and Data Transmission Solutions:

1. Wi-Fi Signal Amplification:

A Wi-Fi range extender was used during testing to ensure stronger signals for uninterrupted Firebase transmission.

2. Retry Logic and Offline Caching:

ESP8266 [2][3][5] was programmed with retry logic and basic buffering to cache data temporarily and resend it upon reconnection.

Machine learning [4][5] and Cloud Solutions:

1. Model Optimization and Feature engineering [5][6]:

Only features with high information gain (pH, temperature, color, Density [1][3][5]) were retained. StandardScaler and IQR-based outlier removal improved overall Accuracy [5][8].

2. Cross-Validation & Hyperparameter Tuning:

K-fold cross-validation was applied to prevent overfitting. LightGBM [5][8]'s learning rate and

number of leaves were fine-tuned to balance performance and speed.

3. Removal of Turbidity:

Based on correlation analysis, turbidity was excluded as it introduced noise. This improved model clarity and reduced training complexity.

User Interface Improvements:

1. Display Formatting Enhancements:

Shorter messages like “Grade: High” were used to maximize legibility. Icons were considered for future versions.

2. Responsive Dashboard Framework:

Tailwind CSS media queries were used to make the dashboard scale fluidly across mobile, tablet, and desktop. User testing helped refine font sizes and layout spacing.

Security and Data Management Enhancements:

1. Role-Based Firebase Rules:

Firebase Authentication was configured to assign user roles, and Firebase Rules were written to restrict each role's access to specific data paths.

2. Timestamp and Data Sync Improvements:

All sensor data entries now include device ID and UTC timestamps generated directly from the NodeMCU [2][3][5] to avoid duplication or collision in the cloud.

Detailed Comparison of Algorithms

To identify the most suitable Machine learning [4][5] algorithm for predicting milk quality, multiple models were trained and evaluated. The evaluation criteria included Accuracy [5][8], precision, recall, F1-score, training time, and suitability for deployment on low-power IoT hardware.

Dataset [17] Description

1. Total samples: 10000+
2. Attributes: pH, Temperature, Colour, Density [1][3][5]
3. Target: Milk Quality (High, Medium, Low)
4. Data Source: Real-time sensor data + synthetic augmentation for diversity
5. Preprocessing:

- Normalization: Min-max scaling
- Label Encoding: 0 = Low, 1 = Medium, 2 = High
- Split: 80% Training, 20% Testing

Algorithms Compared

S.No	Model	Description
1	Logistic Regression	Linear classifier, simple, interpretable
2	Support Vector Machine (SVM)	Effective for high-dimensional spaces
3	Decision Tree	Non-linear, interpretable
4	Random Forest [5][8]	Ensemble of trees, reduces overfitting
5	LightGBM [5][8]	Gradient boosting with fast training, supports categorical features natively

Table 10: Algorithms Compared

Model	Accuracy [5][8]	Precision	Recall	F1-Score	Notes
Logistic Regression	61	0.60	0.56	0.55	Performs poorly on non linear data
SVM	77	0.75	0.75	0.75	Sensitive to feature scaling
Decision Tree	97	0.97	0.97	0.97	Fast but overfits on noise
Random Forest [5][8]	98	0.98	0.97	0.98	Good trade-off, more memory usage
LightGBM [5][8]	98	0.98	0.97	0.98	Best Accuracy [5][8]and fastest training

Table 11: Performance Metrics Comparison

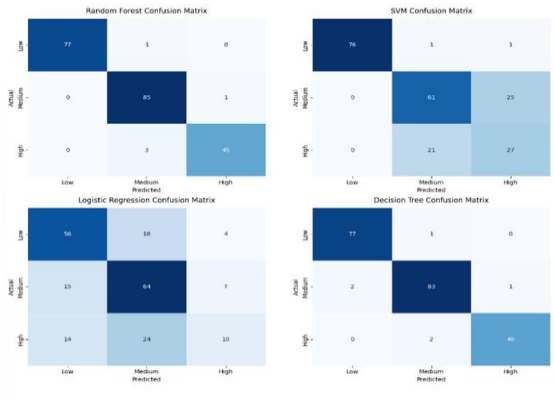

Figure 18: Random Forest [5][8], Decision Tree, SVM, and Logistic Regression

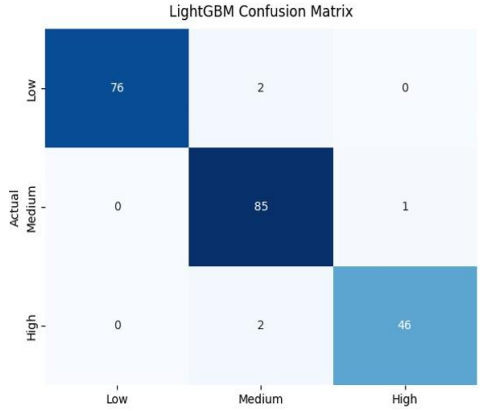

Figure 19: LightGBM [5][8] Confusion Matrix

Final Model Justification: LightGBM [5][8]

- Outperforms all other models in both Accuracy [5][8] and F1-score.
- Fast inference, ideal for real-time prediction.
- Low memory footprint, suitable for IoT-based deployment.
- Handles class imbalance and non-linear boundaries effectively.

Thus, LightGBM [5][8] was selected as the final prediction model for the milk quality classification system.

VI CONCLUSION & FUTURE SCOPE

6.1 Conclusion

This project successfully developed and demonstrated an IoT-enabled milk quality monitoring system that integrates real-time data acquisition with Machine learning [4][5]-based analysis. The system provides an automated, scalable, and remote solution to monitor milk parameters including pH, temperature, color, and Density [1][3][5]. Sensor data is transmitted using the ESP8266 [2][3][5] microcontroller to a Firebase cloud [6][8] backend, where it is processed and analyzed to determine the quality of milk samples using trained ML models.

Among the various ML algorithms tested, LightGBM [5][8] exhibited the highest Accuracy [5][8] and consistency in classifying milk quality into three grades: Low, Medium, and High. The system's performance was evaluated using metrics such as Accuracy [5][8], precision, recall, F1score, and confusion matrices. The results confirmed the system's reliability in real-time classification, with LightGBM [5][8] achieving over 98% Accuracy [5][8].

Through the integration of hardware, cloud services, and ML models, the system provides a significant improvement over traditional manual milk quality testing, which is often timeconsuming, subjective, and less scalable. The UI developed using React and Tailwind CSS offers real-time insights into milk quality and sensor readings, enabling farmers and dairy processors to make quick decisions and ensure Food safety [1][10][11].

6.2 Future Scope

While the current system provides a robust foundation, several enhancements can be explored in future iterations:

1. Expansion of Sensor Suite:

Incorporating additional Sensors [1][2][3] such as lactometer (for specific gravity), conductivity (to detect adulteration), and gas Sensors [1][2][3] (to detect spoilage gases like ammonia) will enhance system sensitivity.

2. Deployment in a Distributed Network:

The system can be scaled across multiple collection centers with a centralized monitoring dashboard that aggregates and visualizes data across locations.

3. Edge ML Integration:

Future versions could integrate lightweight ML inference directly on the ESP8266 [2][3][5] or ESP32 to enable classification at the edge, reducing reliance on constant internet access.

4. Android/iOS Mobile Application:

A cross-platform mobile app can be developed to allow farmers to view milk quality results, history, and alerts directly on their smartphones.

5. Blockchain [10][16] Integration for Traceability:

Storing milk quality logs on a Blockchain [10][16] ledger can ensure tamper-proof traceability from farm to factory, which is especially useful for large-scale dairy supply chains.

6. Integration with Government and Regulatory Databases:

The system could be linked with Food safety [1][10][11] authorities or co-operative society portals to automatically log and report quality metrics, ensuring transparency and accountability.

7. Predictive Maintenance and Sensor Health Monitoring:

Monitoring sensor drift and usage patterns to notify when recalibration or replacement is needed could improve long-term reliability.

8. Auto-Grading and Pricing Mechanism:

The quality classification can be integrated with pricing logic to automatically suggest fair pricing based on quality, supporting both farmers and procurement agents.

9. Localization and Language Support:

Adding regional language support and voice outputs can make the system more accessible to rural farmers with limited literacy.

In conclusion, this system demonstrates how a combination of IoT, ML, and cloud computing can bring intelligent automation into the dairy industry, ensuring higher quality control, minimizing waste, and improving farmer incomes. With further improvements, it has the potential to become a commercial-grade solution adopted across rural and urban dairy networks.

REFERENCES

[1] González, A. F., Laíño, R. B., Costa-Fernández, J. M., & Soldado, A. (2024). Progress and Challenge of Sensors [1][2][3] for Dairy Food safety [1][10][11] Monitoring. Sensors [1][2][3], 24(5), 1383. https://doi.org/10.3390/s24051383

[2] Kalaiarasi, G., Anandkumar, A., Kathirvelu, M., Dinakaran, K., & Ashok, J. (2022). IOT Based Milk Analyzer using Arduino with Wi-Fi Module. 2022 IEEE 2nd International Conference on Mobile Networks and Wireless Communications (ICMNWC), 1–5. https://doi.org/10.1109/icmnwc56175.2022.10031666

[3] Habsari, W., Udin, F., & Arkeman, Y. (2022). An analysis and design of fresh milk smart grading system based on internet of things. IOP Conference Series Earth and Environmental Science, 1063(1),012059. https://doi.org/10.1088/17551315/1063/1/012059

[4] Çelik, A. (2022, December 31). Using Machine learning [4][5] algorithms to detect milk quality. https://dergipark.org.tr/en/pub/ejfst/issue/74913/1180639

[5] Durgun, M. (2024). Real-Time Milk Quality Control Using Multi-Spectral Sensing and Edge computing [5][15]: Advancing On-Site Detection of Milk Components with XGBoost. Applied Sciences, 14(23), 10916. https://doi.org/10.3390/app142310916

[6] Lal, P. P., Prakash, A. A., Chand, A. A., Prasad, K. A., Mehta, U., Assaf, M. H., Mani, F. S., & Mamun, K. A. (2022). IoT integrated fuzzy classification analysis for detecting adulterants in cow milk. Sensing and Bio-Sensing Research, 36, 100486. https://doi.org/10.1016/j.sbsr.2022.100486

[7] Mhapsekar, R. U., O'Shea, N., Davy, S., & Abraham, L. (2024). Hybrid Blended Deep learning [7] approach for milk quality analysis. IEEE Transactions on Emerging Topics in Computational Intelligence, 8(3),2253–2268. https://doi.org/10.1109/tetci.2024.3369331

[8] Goyal, K., Kumar, P., & Verma, K. (2024). XAI-empowered IoT multi-sensor system for real-time Milk adulteration [1][2] detection. Food Control, 164, 110495. https://doi.org/10.1016/j.foodcont.2024.110495

[9] Vishniakou, U. A., Al-Masri, A. H., & Al-Haji, S. K. (2021). Model and structure of the network internet of things for monitoring milk quality. «System Analysis and Applied Information Science», 1, 39–44. https://doi.org/10.21122/2309-4923-2021-1-39-44

[10] Gehlot, A., Malik, P. K., Singh, R., Akram, S. V., & Alsuwian, T. (2022). Dairy 4.0: Intelligent Communication Ecosystem for the Cattle Animal Welfare with Blockchain [10][16] and IoT Enabled Technologies. Applied Sciences, 12(14), 7316. https://doi.org/10.3390/app12147316

[11] Manisha, N., & Jagadeeshwar, M. (2023). BC driven IoT-based food quality traceability system

for dairy product using Deep learning [7] model. High-Confidence Computing, 3(3), 100121. https://doi.org/10.1016/j.hcc.2023.100121

[12] An optimized neural network model to predict milk quality. (2024, January 18). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/10463373

[13] Empowering Dairy Farmers IoT milk Quality Monitoring devices. (2024, May 9). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/10602406

[14] Development of IoT based sensor system for detection and identification of milk adulterants. (2024, March 18). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/10731612

[15] An Edge-Centric industrial IoT solution for smart dairy processing. (2024, September 1). IEEE Journals & Magazine | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/10644014

[16] Milk chain with ML & Blockchain [10][16]: Revolutionizing Transparency for Ensuring Milk Purity in Dairy Industry. (2024, April 12). IEEE Conference Publication | IEEE Xplore. https://ieeexplore.ieee.org/abstract/document/10543524

[17] Milk quality prediction. (2022, August 1). Kaggle. https://www.kaggle.com/Dataset [17]s/cpluzshrijayan/milkquality