



IoT-Based Chemical Detection System for Milk Quality Monitoring: A Comprehensive Review of Sensor Technologies, IoT Architecture, Machine Learning and Connected Quality Assessment

Om Barbude, Prof. Abhishek P. Bangar, Prof. Sachin B. Bhosale, Dr. A. A. Khatri

Department of Computer Engineering
Jaihind College of Engineering, Kuran, Pune, Maharashtra, India

Abstract. Milk can go wrong at almost any point between the cow and the consumer. Natural variation in composition, bacterial growth, rough handling and deliberate adulteration all leave their mark on the final product. Laboratory methods such as spectroscopy, chromatography and microbiological testing are still the reference, but most farms and small collection centres cannot run them for every can of milk. Internet of Things (IoT) sensing is often suggested as a cheap first check that fills this gap. In this paper we review the verified literature on IoT-based milk quality monitoring. We look at four connected topics: what quality indicators and adulteration involve, which sensors (pH, electrical conductivity, temperature, turbidity, optical and gas) are available, how an ESP32-class controller and a cloud dashboard can collect and share the readings, and how machine learning can combine several weak signals into one screening decision. The studies we found show that working prototypes are possible, but they say little about long-term field use. Cross-sensitivity, calibration drift, differences between milk batches, unreliable reference labels and cost remain open. A sensor reading can flag a sample, but it cannot name a chemical, so laboratory confirmation is still needed. We also compare the representative studies, propose a conceptual six-layer architecture with the sensors chosen for a student prototype, and list research gaps in order of priority. No new experimental results are reported.

Keywords: Milk Quality, Chemical Detection, Internet of Things, Adulteration Screening, Sensors, ESP32, Machine Learning, Sensor Fusion

I. Introduction

Milk is nutritious, but it spoils quickly. How good it is when it reaches the consumer depends on the health of the animal, hygiene during milking, how fast it is cooled, and how it is transported and stored. It travels from the farmer through collection centres and processors before anyone drinks it, and at each step something can go wrong, either by accident or on purpose. Adding water to increase volume is the classic example [1], [2].

Large dairy plants have trained staff and laboratories for routine checks. A small producer or collection centre usually does not. A connected device at the collection point could record readings and send them to a dashboard, so that staff can decide which samples deserve a proper laboratory test [1], [3]. Three things make this realistic today: inexpensive sensors for physical and chemical signals, microcontrollers such as the ESP32 that pass the data on, and machine-learning models that merge several readings into one screening result [1]–[3].



We need to be careful with the words “chemical detection”. A cheap pH, conductivity or turbidity probe measures a general property of the milk. It does not recognise a particular adulterant, and the same reading can come from normal variation, temperature, leftover cleaning liquid or a tired sensor. For this reason we treat the systems in this paper as screening and decision-support tools, not as replacements for laboratory analysis. The useful question is not whether one number looks normal, but whether the sample behaves as expected or needs another look.

1. Aim and Scope

Our aim is to go through the main parts of an IoT-based milk screening system and to sketch an architecture that a student team could build as a prototype and then validate in a controlled way. We cover sensors, data acquisition, communication, software, machine learning, calibration, alerts and quality assurance. The focus is adulteration screening and general quality indicators. We do not claim that the system finds every adulterant, diagnoses animal disease, or judges microbial safety from a single electronic reading.

2. Contributions and Organisation

The paper makes four contributions: a structured narrative review of verified studies, a comparison of representative work, a list of research gaps, and a conceptual sample-to-decision architecture with signal-processing, evaluation and implementation guidance. Section II explains how we selected the literature, Section III reviews it, Section IV compares the studies, Section V discusses open problems, Section VI presents the proposed design and research directions, and Section VII concludes.

II. Review Methodology and Literature Selection Criteria

1. Search Strategy

We searched for combinations of the terms milk quality, adulteration, chemical detection, IoT, ESP32, sensor fusion, electronic nose, spectroscopy and machine learning. Recent peer-reviewed journal papers and reviews on milk sensing and connected monitoring came first, and every candidate was then checked against its publisher or DOI record.

2. Inclusion and Exclusion Criteria

A source made the cut if it was peer reviewed or an official standard, dealt with milk-quality sensing, adulteration screening, IoT monitoring or machine learning on milk data, described a method, system or review, and had a DOI or other publication record we could verify. We left out marketing material, blog posts, vendor datasheets and anything we could not verify.

3. Screening Outcome and Limitations

Eight core sources remained: one systematic review [1], two primary system studies [2], [3], three standards and guidelines [4]–[6], and two methodological papers on model evaluation [7], [8]. Four more technical references [9]–[12] support implementation details. We did not follow a systematic protocol: there are no logged Scopus or Web of Science queries and no study counts, and the milk-specific set is small and mostly from 2025. Older work on individual sensing techniques is not covered and would be a natural extension. Accuracy figures belong to the original studies and should not be read as the performance of our design. We built no prototype and measured no milk for this paper.

III. Literature Review

1. Milk Quality Indicators and Adulteration

Milk quality has several dimensions that are related but not interchangeable: composition, acidity, temperature history, microbial condition and unwanted substances. A system should not be called a universal milk analyser unless it has been tested for



every task it claims. The reviews discuss dilution with water, added starch or sugar, urea, detergents and neutralising agents, among other concerns [1].

Which method works depends on the target. Some substances show up in spectral changes, while others need a selective chemical reaction, a specific biosensor or a laboratory method. Added water can shift several readings, but so can ordinary differences between milk samples; freezing-point measurement by cryoscopy, as described in ISO 5764 [11], is the standard reference for it. Starch may change optical scattering or react in a targeted test. Urea is present in normal milk at a low level, so a method has to separate normal variation from an abnormal amount. In short, readings need validated reference ranges, and a single threshold for all samples will not do.

2. Screening versus Confirmation

A useful device answers in plain terms: within the expected range, review needed, or laboratory confirmation recommended. It should never say that a named hazardous chemical is present unless the sensor and model were validated for exactly that purpose. When a result could affect a supplier, a payment or public health, a confirmatory test and a traceable record matter even more [6]. Table I links common quality questions to measurement approaches and their limits.

Table 1: Quality Questions and Suitable Measurement Approaches

Quality question	Possible measurement	Important limitation
Cold-chain history	Temperature sensor and time log	Does not prove chemical or microbial safety
Acidity change	Calibrated pH probe	Not specific to one adulterant
Ionic composition change	Conductivity (EC) sensor	Affected by temperature and natural variation
Suspended particles	Turbidity / optical scattering	Cannot identify a substance by itself
Composition pattern	NIR / MIR or multispectral sensing	Needs reference samples and calibration
Volatile spoilage pattern	Gas-sensor array / electronic nose	Cross-sensitivity and drift may be substantial
Specific chemical target	Selective assay or validated sensor	Target-specific validation is essential

3. pH Sensing

A pH probe estimates hydrogen-ion activity and turns it into a pH value. For milk, the probe must be suitable for food samples, cleaned as instructed and calibrated with proper buffers, and temperature compensation may be needed. Protein and fat coat the probe, so cleaning and storage are part of the measurement and not an optional chore. pH helps show acidity changes caused by fermentation or storage. A normal pH, however, does not rule out adulteration, and an odd one may simply mean a handling error or a dirty probe.

4. Electrical Conductivity Sensing

An electrical conductivity (EC) sensor measures how easily the sample carries current. The reading depends on the ions present and on temperature, and also on electrode shape, fouling and the measuring circuit. Comparisons only make sense when temperature is controlled or compensated by a validated method. EC adds useful information, but it is not a chemical fingerprint.

5. Temperature Sensing

A digital sensor such as the DS18B20 hands a temperature reading straight to a microcontroller. Where the sensor sits, how fast it responds and how well it touches the sample all change the result. The system should store units and timestamps, notice missing values and keep sample temperature separate from room temperature. Temperature matters for another reason as well: it affects the output of the pH and EC sensors.



6. Turbidity and Optical Sensing

A turbidity sensor usually shines light through the sample and measures how much is scattered or absorbed. Colour, bubbles, container shape, ambient light and suspended particles all disturb the reading. Cheap modules can work in a controlled demonstration, but they should not be presented as identifying a named adulterant without proper experiments. Near-infrared (NIR), mid-infrared (MIR) and multispectral devices record many wavelengths and give richer composition information, at the price of careful calibration and many more reference samples; ISO 9622 gives guidelines for mid-infrared spectrometry of milk [5].

7.. Gas Sensors and Electronic Noses

An electronic nose combines several gas-sensitive elements and reads the overall response pattern instead of trusting a single sensor. It may help when spoilage changes the volatile compounds above the sample. Humidity, temperature, cleaning chemicals and unrelated smells affect the same sensors, though, so long-term stability and the link between the headspace and the signal have to be tested before an array is used for milk decisions.

8. ESP32 and Signal Acquisition

The ESP32 is a practical controller for reading digital and analogue sensors, running basic checks and sending records over Wi-Fi [10]. Analogue signals may need conditioning, filtering and a suitable analogue-to-digital path. Firmware should never assume that a value is valid; it needs range checks, timeout handling, sensor-status flags and a record of the calibration used.

Table II lists the main design checks for each component.

Table 2: Practical Sensor-Selection Checklist

Component	Role in prototype	Design checks
pH probe	Acidity-related signal	Buffer calibration, cleaning, temperature
EC module	Ionic conductivity signal	Reference solution and temperature compensation
DS18B20 or equivalent	Sample temperature	Placement, response time, timestamp
Turbidity / optical module	Light scattering or attenuation	Sample holder, bubbles, ambient light
ESP32 controller	Read, timestamp and transmit	ADC range, power, connectivity, error handling
Reference test	Ground-truth label	Suitable laboratory method and sample traceability

9. IoT Communication, Storage and Dashboards

The device can send records over Wi-Fi through an authenticated API or a lightweight publish-subscribe protocol such as MQTT [9]. Each message should carry the device ID, sample ID, timestamp, sensor values, calibration version and result status. Cloud storage keeps the history and supports trends and reports, so data must be protected in transit and dashboard access controlled. A local queue can hold records during an outage and upload them later. Prasad and Chidambaranathan [2] describe an IoT-enabled system that digitises the milk collection stage and assesses quality with machine learning, and Fizza et al. [3] report an inexpensive AI-powered IoT sensor for continuous farm-to-factory monitoring. On the dashboard, wording matters: “screening flag, confirm with laboratory” is more honest than “chemical detected” when the sensor cannot identify the chemical directly.

10. Machine Learning and Sensor Fusion

Machine learning can combine readings that are hard to interpret one at a time, but it cannot fix a poor sensor or unreliable labels. A model trained on a handful of samples from one place tends to learn local conditions instead of general patterns, so



the data-collection plan matters at least as much as the algorithm [1]. Every record should carry the sensor readings, timestamps, sample source, storage conditions, calibration information and a trustworthy reference label. Repeated readings of the same sample must stay together when the data are split into training and test sets. Otherwise near-duplicates land on both sides and the results look better than they really are.

Table III summarises candidate methods. Logistic regression and decision trees make good starting points because it is easy to see what they are doing. Random forests and gradient boosting handle non-linear relationships, support vector machines suit moderately sized datasets, and deep networks only pay off when there is enough data, typically spectral or image-like inputs. A simpler model that behaves consistently on independent batches and can explain itself may be a better choice than one that is hard to audit. Feature importance helps a reviewer, but it does not prove that the model has found a causal chemical signature. Open-source libraries such as scikit-learn [12] provide all of these methods.

Table 3: Candidate Analytical Methods and Appropriate Use

Method	Potential advantage	Main caution
Threshold rules	Simple to explain and implement	Normal variation may overlap target changes
Logistic regression	Interpretable baseline	May underfit complex patterns
Decision tree / random forest	Models non-linear interactions	May learn source-specific shortcuts
Support vector machine	Useful for moderate-sized feature sets	Needs scaling and careful tuning
Gradient boosting	Captures complex relationships	Needs careful validation and monitoring
Spectral ML / deep learning	Uses multi-wavelength patterns	Needs more reference data and independent testing

11. Data Trust and the Limits of Sensing

A connected record is only as good as the signal and label behind it. Drifting sensors, fouled probes, dirty containers and informal labels produce stored results that look authoritative and are wrong. Reliable screening therefore needs documented calibration, raw data kept next to processed values, traceable sample identifiers and laboratory confirmation for anything that affects payment or public health [1], [4], [6]. Without timestamps, sensor identity, calibration details and reference labels, old data may be useless for model development. Traceability also helps tell a real change in the milk from a faulty probe or a software update.

12. Standards and Quality Assurance

Established standards provide the reference frame for any screening device. ISO 707 gives guidance on sampling [4], ISO 9622 on mid-infrared spectrometry [5], ISO 5764 on freezing-point determination [11], and the Codex code of hygienic practice describes good practice along the milk chain [6]. A screening device should be tested against these reference procedures, not against informal judgement, and each record should name the reference method behind its label. None of the sources we reviewed shows a low-cost sensor array replacing them, so we position our design as a first-line screen that points toward these methods.

IV. Comparative Analysis of Existing Studies

Table IV summarises the representative primary studies and the review article. Because they differ in scope, sensing method and evaluation, we compare them qualitatively and not numerically.



Table 4: Comparison of Representative Studies on Iot-Based Milk Quality Monitoring

Author, year	Type / objective	Technologies	Main contribution	Scope limitation
Mhapsekar et al., 2025 [1]	Systematic review; IoT and AI in milk quality	Sensors, IoT, machine learning	Surveys sensing, AI and IoT options for milk quality monitoring	Review; no new system
Prasad & Chidambaranathan, 2025 [2]	System; IoT-enabled milk collection	IoT, machine learning	Connects the collection stage to data collection and quality analysis	Depends on dataset and labels
Fizza et al., 2025 [3]	Sensor system; farm-to-factory monitoring	Low-cost AI + IoT sensor	Shows connected sensing with model-based estimation	Composition error differs from adulteration accuracy
Proposed design (this paper)	Conceptual screening architecture	pH, EC, temperature, optical, ESP32, ML	Traceable multi-sensor screening with a confirmation pathway	Conceptual; needs bench and field tests

Three patterns stand out. First, the primary studies are system proposals or prototypes, not long-term field deployments [2], [3]. Second, the review literature calls the field promising but dependent on calibration, varied reference data and testing beyond a single laboratory [1]. Third, because sensors, tasks and evaluation methods differ so much, the verified evidence does not let us name a best sensor combination.

1. Why Published Results Cannot Be Compared Directly

A study can report very high accuracy simply because it separates a small set of prepared samples under controlled conditions. The picture changes when milk comes from new farms, when samples contain mixtures, or when temperature and sensor ageing affect the readings. Studies also differ in how they define adulteration, which reference method they use and how balanced their classes are. Composition-estimation error is not the same thing as adulteration-detection accuracy, so comparisons should name the task and the metric and not suggest that every figure measures the same capability.

2. Lessons for the Proposed Project

The first step for our project is to check whether the chosen sensors can repeatedly tell defined samples from suitable controls. Only when that baseline is stable does it make sense to train a classifier or build more elaborate cloud analytics. A few carefully calibrated sensors with a known link to the target are worth more than many cheap modules whose link to the target is unclear, and a dashboard should show the evidence behind a result instead of an unexplained score. A traceable record of reference tests, sample preparation, calibration and repeat measurements is worth more than an unsupported accuracy claim.

3. Summary of Findings

To sum up, the verified evidence supports five points. Connected sensing and model-based analysis of milk have been shown in prototype form [2], [3]. The review literature expects benefits for screening but stresses calibration and varied reference data [1]. Low-cost general sensors measure general properties and cannot identify a named adulterant by themselves. Reported accuracies cannot be compared directly across studies. And field evidence on drift, cost and operator use is still missing, which is exactly where our design concentrates.



V. Research Challenges and Identified Gaps

The gaps below come from the reviewed literature and from our own reasoning about the system design.

1. Selectivity and Cross-Sensitivity

A sensor can respond to several substances, and a reading can change for reasons that have nothing to do with the target. This is the central difficulty of low-cost chemical screening [1]. More work is needed to show which signal combinations are informative enough for a clearly defined task, and when a selective assay is unavoidable.

2. Calibration and Drift

Electrodes age, optical surfaces get dirty and gas-sensor baselines wander. A calibration done once cannot keep readings dependable for months. Designs need calibration intervals, reference checks, drift alerts and a procedure for replacing or re-calibrating sensors.

3. Sample and Batch Diversity

Milk changes with the animal, the feed, the stage of lactation, the season, the time of collection and the storage history. A model trained on a narrow set of samples can easily mistake these differences for adulteration [1]. Datasets should cover several sources and batches, carry useful metadata and be tested on independent samples.

4. Mixed Adulterants and Concentration Ranges

Real samples may contain more than one unusual substance, often at lower concentrations than in demonstrations. Models trained on single adulterants may not carry over to mixtures, and preparing such samples has to be done safely and under supervision.

5. Reference Labels and Ground Truth

A model is only as good as its labels. Informal judgements, or labels based on one indirect sensor, introduce errors [7], [8]. Wherever possible a validated reference method should supply the label, and the measured reference value should be kept, not only a yes or no.

6. Connectivity, Power and Cybersecurity

Collection centres often have patchy internet. The device should therefore store records locally and synchronise later without creating duplicates. Credentials do not belong in source code, communication should be authenticated, and access to supplier or batch information should be restricted.

7. Limited Field Evidence

The primary studies are prototypes [2], [3]. How such systems behave across seasons, farms, operators, cleaning cycles and sensor replacements has not been shown.

8. Human Factors and Cost

Results that are hard to read, or alerts that fire too often, wear down trust, and operators need to know what the device measures and when to repeat a test. The full cost of replacement sensors, calibration standards, cleaning supplies, operator time and confirmatory tests is rarely reported, even though a device is only practical if it can be maintained and its decisions justify the expense.

VI. Proposed System Design and Research Directions

1. Proposed Conceptual Architecture

Fig. 1 shows the architecture we propose, derived from the systems in [1]–[3]. A milk sample is handled by a fixed protocol (layer 1) and measured by pH, conductivity, temperature and optical sensors (layer 2). An ESP32-class controller timestamps every reading, checks ranges and keeps the raw values (layer 3). The edge layer applies rule checks and, once validated, a trained model, and shows no decision when a sensor fails (layer 4). Records go over Wi-Fi to secure storage with a local queue for outages (layer 5), and the dashboard presents the result as within range, review needed or confirm in laboratory (layer 6). Keeping these roles separate stops a connectivity failure from being mistaken for a quality result.

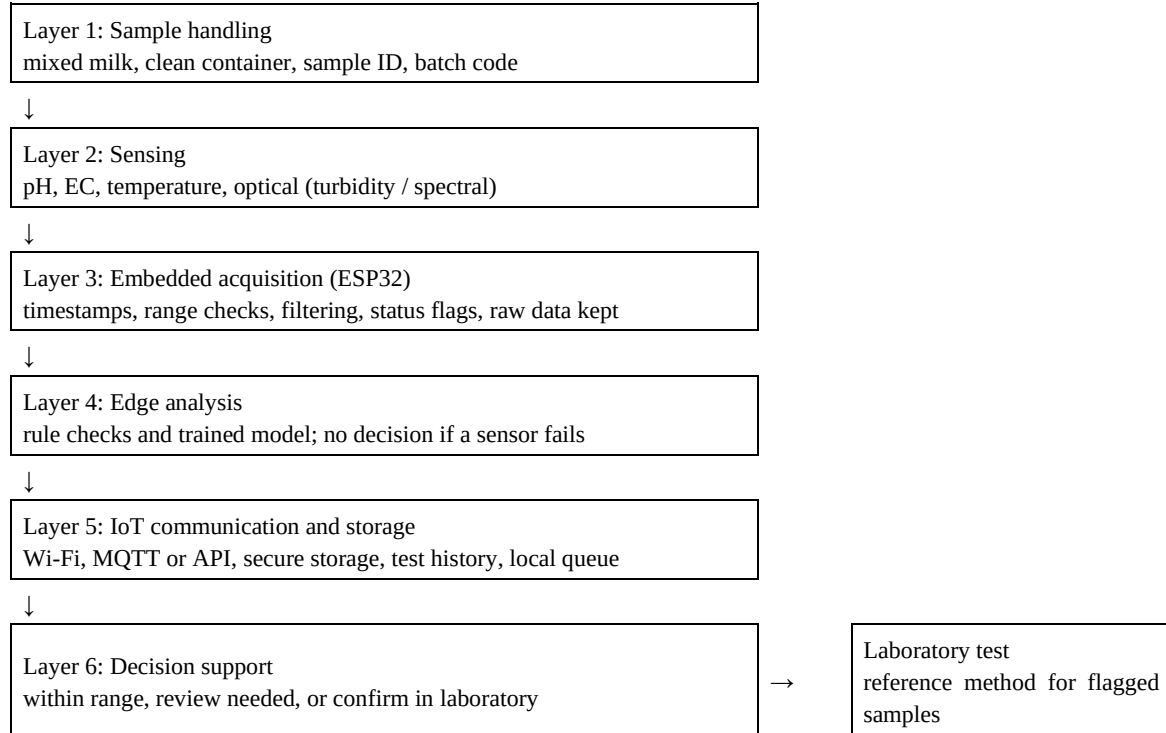


Fig. 1. Proposed conceptual architecture for IoT-based milk quality screening (conceptual, not experimentally validated)

2. Selected Sensors and Hardware

The prototype uses low-cost sensors that measure general properties of milk. Each one was chosen because it has a clear role, a known operating range and a calibration procedure (Table V). A food-suitable pH probe gives an acidity-related signal, an electrical conductivity (EC) module gives an ionic signal, a DS18B20 records sample temperature (which is also the input for temperature compensation), and a turbidity module provides a light-scattering signal. An ESP32 reads the sensors, timestamps and checks the values and sends records over Wi-Fi [9], [10]. The first screening targets are dilution-type changes and general quality indicators. Specific chemicals such as urea or detergents need selective assays or spectral methods, so we leave them as future work. Near-infrared or multispectral sensing and a gas-sensor array could be added later for targets that need more specificity [1], [5].



Table 5: Sensors and Hardware Selected for the Prototype

Component	Measures	Role in the prototype
pH probe (food-suitable)	pH	Acidity-related signal; buffer calibration
EC module	Conductivity	Ionic signal; temperature compensation
DS18B20 sensor	Sample temperature	Cold-chain context; compensation input
Turbidity / optical module	Light scattering	Suspended-particle signal
ESP32 controller	Acquisition, Wi-Fi	Timestamps, range checks, MQTT or API upload
Optional later: NIR / gas array	Composition / volatiles	More specific targets; needs reference data

3. Working Principle

The working principle has six steps. (1) The operator mixes the milk sample following the test protocol and records the sample ID, batch code and collection time [4]. (2) The sensors measure pH, conductivity, temperature and optical signals at a repeatable volume and temperature. (3) The ESP32 timestamps each reading, rejects impossible values and stores raw and processed values together with the calibration version. (4) The edge layer returns within range, review needed, or no decision if a sensor is unavailable. (5) The record is sent to secure storage and the dashboard shows the readings, the calibration status and the next step. (6) Any flagged sample goes to the laboratory, and the reference result is stored with the record for later model validation.

4. Sampling and Data Record

A consistent sampling routine is essential. The milk should be mixed as the protocol says, placed in a clean container and measured at a repeatable volume and temperature [4]. A dirty container or an inconsistent routine can ruin a result even when the electronics are working perfectly. Table VI shows the record structure we suggest. Values should always be stored with their units, and raw measurements should be kept next to processed values so a reviewer can trace how the final result came about.

Table 4: Suggested Measurement Record Structure

Field	Type / unit	Purpose
sample_id, device_id	text	Trace a result to a sample and a device
timestamp	ISO 8601	Order events; detect missing data
temperature_c	degrees C	Cold-chain context; compensation input
ph	pH units	Acidity-related signal
conductivity	uS/cm	Ionic signal; compensated value stored
turbidity	raw units	Scattering or attenuation signal
calibration_version	text	Link readings to calibration state
connectivity_status	text	Separate network faults from quality results
screening_result	category	Within range, review or confirm, no decision



5. Signal Processing and Calibration

Calibrating a pH probe with two buffers gives a straight-line mapping from probe voltage V to pH. If the probe reads V_1 and V_2 in buffers of known pH_1 and pH_2 , the slope and offset follow from (1) and the reading from (2).

$$m = (\text{pH}_2 - \text{pH}_1) / (V_2 - V_1), \quad b = \text{pH}_1 - m \cdot V_1 \quad (1)$$

$$\text{pH} = m \cdot V + b \quad (2)$$

Conductivity depends on temperature, so we can compensate to a reference temperature T_{ref} with a linear correction as in (3). The coefficient a has to be determined experimentally for the sample type, not assumed.

$$\text{EC}_{\text{ref}} = \text{EC}_T / [1 + a (T - T_{\text{ref}})] \quad (3)$$

Repeated readings reveal unstable signals. After invalid values are thrown away, we take the median of the remaining n readings as in (4). It copes with isolated spikes better than the mean does.

$$X^* = \text{median} \{ x_1, x_2, \dots, x_n \} \quad (4)$$

6. Decision Workflow

Fig. 2 shows the decision workflow, including the pathway to laboratory confirmation. A small rule-based safety layer keeps the basic checks running when the internet is down, and a trained model may later estimate whether a sample resembles validated reference groups. The system should abstain when readings fall outside the training range or when a required sensor is not available. That makes the device safer and easier to audit.

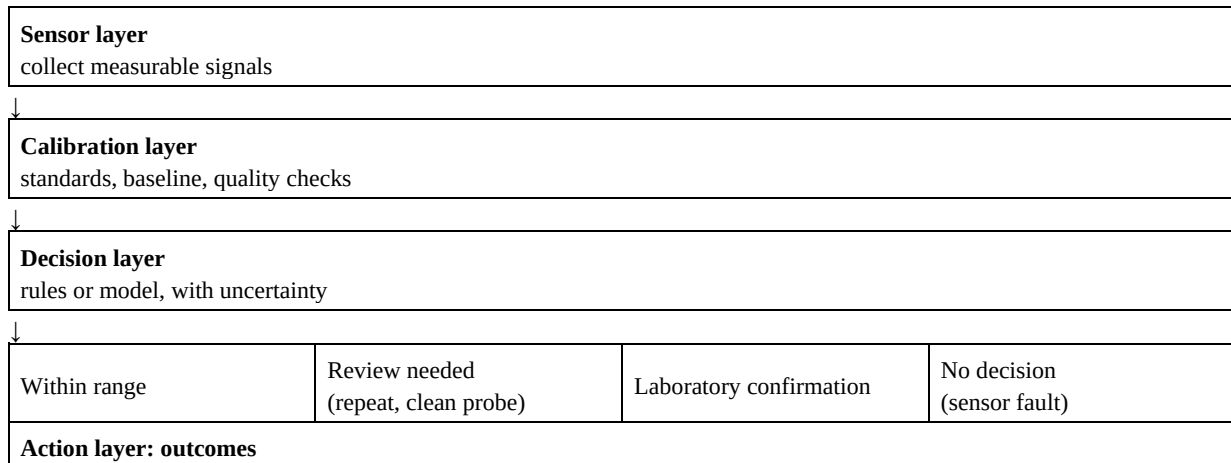


Fig. 2. Decision workflow with an explicit confirmation pathway

7. Evaluation Measures

Accuracy alone can hide serious mistakes when suspicious samples are rare [7], [8]. Fig. 3 shows the four possible outcomes of a screening decision against a reference label, and the measures in (5)–(8) follow from it.



	Device result: Flagged	Device result: Not flagged
Reference positive	TP correct alert	FN missed problem
Reference negative	FP false alert	TN correct pass

Fig. 3. Confusion matrix for screening decisions against a reference method (FN is the costliest error for safety screening)

$$\text{Sensitivity} = \text{TP} / (\text{TP} + \text{FN}), \quad \text{Specificity} = \text{TN} / (\text{TN} + \text{FP}) \quad (5)$$

$$\text{Precision} = \text{TP} / (\text{TP} + \text{FP}) \quad (6)$$

$$\text{False-negative rate} = \text{FN} / (\text{TP} + \text{FN}) = 1 - \text{Sensitivity} \quad (7)$$

$$F_1 = 2 \times \text{Precision} \times \text{Sensitivity} / (\text{Precision} + \text{Sensitivity}) \quad (8)$$

Sensitivity tells us how many reference-positive samples are caught, specificity how many reference-negative samples are correctly passed, precision how often an alert is right, and the false-negative rate how often a real problem is missed. Where the dataset allows, a confusion matrix and confidence intervals should be reported. The final test set must contain samples or batches that were not used during development, and for use across farms a stricter test holds out whole farms or time periods. Tests should also cover calibration stability, repeatability, response time, data loss and the share of samples that need confirmatory testing.

8. Implementation Plan for a Student Prototype

A staged plan lowers the risk of building a polished interface around measurements nobody has validated. Stage 1 defines the intended use: which quality questions the prototype screens, which substances or conditions are in scope and what the output may claim. If no reliable reference method exists for a target, that target is future work, not a finished capability. Stage 2 tests each sensor on the bench and records calibration readings, repeatability, response time and behaviour after cleaning. Hazardous adulterants must not be used without laboratory supervision and safety procedures. Stage 3 builds the firmware and the data structure of Table VI. Stage 4 builds the dashboard, and any demonstration data must be clearly marked as demo mode so nobody mistakes it for a real test. Stage 5 develops the model on enough independent samples with reference labels, splitting by batch or source, starting from a simple baseline and keeping a final holdout set untouched until the design is fixed. Stage 6 is a small supervised pilot that compares device output with the reference method; its results should not be used for supplier payment or safety decisions until the evidence and the required approvals support that. Table VII lists the evidence to keep at each stage.

Table 7: Suggested Development and Validation Checklist

Stage	Deliverable	Evidence to keep
1. Scope	Written target and limitations	Target list, reference method, intended decision
2. Sensor test	Repeatable raw readings	Calibration records, blanks, repeated measurements
3. Acquisition	Reliable timestamps and error flags	Firmware version, logs, units
4. Dashboard	History and traceable result display	Screenshots, test records, access controls
5. Model validation	Independent evaluation	Split method, confusion matrix, metrics
6. Pilot	Comparison with reference results	Disagreement log, drift checks, operator feedback



9. Failure Modes and Mitigation

Table VIII lists failure modes we can foresee, with a mitigation for each. They come from the sensor and system limitations in Sections III and V, and they should be tested on purpose during bench validation.

Table 8: Failure Modes and Mitigation

Failure mode	Effect	Mitigation
Probe fouling	Slow or biased readings	Defined cleaning step; check against a buffer
Sensor drift	Gradual shift in results	Calibration schedule; drift alert
Unclean container	Contaminated sample	Fixed sampling protocol [4]
Sensor disconnected	Missing or false values	Status flags; no decision shown
Network outage	Lost or duplicate records	Local queue; duplicate-free sync
Out-of-range sample	Unreliable model output	Abstain; request laboratory test
Exposed credentials	Unauthorised access	Authenticated, encrypted communication

10. Cost and Maintenance Considerations

A device is only practical if it can be maintained and if the quality of its decisions justifies the expense. An evaluation should therefore count the full operating cost, including replacement probes, calibration buffers, cleaning supplies, connectivity, operator time and confirmatory laboratory tests. Reports should also describe failure cases, uncertain results and maintenance demands, not just best-case accuracy.

11. Dashboard and Alert Design

The dashboard should show the current sample, the individual readings, the calibration status, the result category and the reason for any alert, plus test history, report export and device status. It should suggest a next step, for example repeating the test, cleaning the probe or sending the sample for confirmation. It should explain uncertainty and never hide a sensor failure behind a green status. Demonstration data must appear only in a clearly marked demo mode. If registration or alerts are included, the interface should say what information is collected, and notifications should not be promised unless the service has actually been configured and tested.

12. Edge-First Operation and Model Updates

An edge-first design keeps the basic checks and a transparent fallback on the controller, while the cloud handles history, model updates and reports. This makes the system more resilient where connectivity is unreliable, because a test can still be completed, recorded and queued while the network is down. The fallback should use simple, documented rules so that an operator can understand why a sample was flagged. Model updates should be versioned and evaluated on independent data before they reach every device, and each stored result should name the model version that produced it, so earlier decisions stay traceable after an update.

13. Responsible Use

Operators need to know what the device measures, what a result means and when to repeat a test. Results should not be used for supplier-payment or safety decisions until the evidence and the required approvals support that use, and hazardous adulterants must never be used in testing without laboratory supervision and safety procedures. These conditions follow from the screening role of the device and from the food-safety principles in [6].



14. Directions Requiring Investigation

Our priorities are: (i) define a few measurable targets and match each to a suitable sensing principle, since a selective method for one adulterant is more defensible than a universal label produced from several indirect measurements [1]; (ii) build a reference dataset with natural variation, known positive samples, repeated measurements and independent batches, split by batch and not by row [7], [8]; (iii) make decisions uncertainty-aware so that the system abstains outside the training range; (iv) run edge-first with local queueing for unreliable internet, and version and evaluate model updates before deployment [2]; (v) try spectral sensing or selective chemical assays for targets that need more specificity [5]; (vi) set calibration intervals, drift alerts and sensor-replacement procedures; (vii) study the full operating cost, including sensors, cleaning, calibration standards, operator time and confirmatory tests; and (viii) run a supervised field pilot that compares the device with the reference method across operators and days. Sensor fusion should also be justified by independent testing, because adding more sensors does not automatically make a system better.

VII. Conclusion and Future Scope

IoT can make milk-quality measurements easier to collect, store and review, and machine learning may help merge several signals into a useful screening decision. The literature we reviewed shows that connected sensing and model-based analysis can be built into working prototypes [1]–[3]. Still, a connected device is not automatically a chemical analyser. pH, conductivity, temperature and turbidity each describe a particular property, and none of them can identify every adulterant on its own. The evidence is dominated by prototypes, and cross-sensitivity, drift, batch variation, label quality and cost are still open. We outlined a layered design that combines consistent sampling, calibrated sensors, an ESP32-class controller, edge checks, connected storage and a dashboard that keeps the test history. The guiding idea is to be honest about what each measurement can and cannot tell us: a suspicious reading should lead to review or confirmatory testing, not to a claim that a particular chemical has been found.

Future work should widen this review to more milk-sensing studies, build the proposed architecture as a bench prototype, and validate it against suitable reference methods on independent milk batches. Spectral sensing or selective assays, uncertainty-aware decisions, edge-first operation and a supervised field pilot are natural next steps, each to be judged on cost, usability and the reliability of the data it records. Until then, the architecture should be read as a design framework, not a proven product. For a student team building this prototype, our advice is simple. Start with a small set of well-understood sensors and calibrate each one. Fix one or two clear targets before any model is trained. Store raw readings, calibration versions and sample identifiers with every record, and get reference-laboratory labels for every sample used in model development. Finally, report the failures and negative findings along with the successes.

The limits of this review should be kept in mind. It is narrative and not systematic, the verified milk-specific evidence is small and recent, and no experiment was run, so we do not know how the proposed architecture performs. The equations, thresholds and checklists in Section VI are general design guidance. They need to be adapted and validated for the chosen sensors, targets and milk population before any claim about detection capability is made.

Acknowledgment

The authors thank the Department of Computer Engineering, Jaihind College of Engineering, Kuran, for guidance and support. No external funding or experimental facility is claimed for this review.



References

1. R. Mhapsekar, D. Kilbane, S. Davy, L. Abraham, M. Fenelon, and N. O'Shea, "A systematic review of the internet of things and artificial intelligence applications in milk quality monitoring and analysis," *Int. J. Dairy Technol.*, vol. 78, no. 3, art. e70049, 2025, doi: 10.1111/1471-0307.70049.
2. R. D. Prasad and S. Chidambaranathan, "Digitizing milk collection stage with IoT-enabled system for quality assessment using ML," *J. Inf. Syst. Eng. Manag.*, vol. 10, no. 29s, pp. 915–928, 2025, doi: 10.52783/jisem.v10i29s.4604.
3. K. Fizza, A. Banerjee, D. Georgakopoulos, et al., "An inexpensive AI-powered IoT sensor for continuous farm-to-factory milk quality monitoring," *Sensors*, vol. 25, no. 14, art. 4439, 2025, doi: 10.3390/s25144439.
4. International Organization for Standardization, ISO 707:2008, Milk and milk products — Guidance on sampling. Geneva, Switzerland: ISO, 2008.
5. International Organization for Standardization, ISO 9622:2013, Milk and liquid milk products — Guidelines for the application of mid-infrared spectrometry. Geneva, Switzerland: ISO, 2013.
6. WHO/FAO, Codex Alimentarius: Code of hygienic practice for milk and milk products, CAC/RCP 57-2004. Rome: FAO/WHO.
7. T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, 2006.
8. D. M. W. Powers, "Evaluation: From precision, recall and F-measure to ROC, informedness, markedness and correlation," *J. Mach. Learn. Technol.*, vol. 2, no. 1, pp. 37–63, 2011.
9. OASIS, MQTT Version 3.1.1, OASIS Standard, Oct. 2014.
10. Espressif Systems, ESP32 Series Datasheet. Shanghai, China: Espressif Systems.
11. International Organization for Standardization, ISO 5764:2009 | IDF 108:2009, Milk — Determination of freezing point — Thermistor cryoscope method. Geneva, Switzerland: ISO, 2009.
12. F. Pedregosa et al., "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.