Vol 6, Issue 9, September 2026 www.ijesti.com E-ISSN: 2582-9734
International Journal of Engineering, Science, Technology and Innovation (IJESTI)

AI and IoT Enabled Smart Irrigation Using Arduino

Surbhi Kumari [1]
[1] M.Sc. Electronics, Dr. Shyama Prasad Mukherjee University (DSPMU), Ranchi.

Chhoti Kumari [2]
[2] M.Sc. Electronics, Dr. Shyama Prasad Mukherjee University (DSPMU), Ranchi.

Dr. Bibek Kumar Sonu [3]
[3] Dr. Shyama Prasad Mukherjee University (DSPMU), Ranchi.

ABSTRACT

Efficient irrigation requires timely information about changing field conditions. This paper develops a low-cost smart-irrigation framework that combines an Arduino Uno, a digital temperature–humidity sensor, Internet of Things communication, simple artificial-intelligence methods, and protected relay control. The study follows a design-and-evaluate approach at Ratu Road, Ranchi. Its analytical structure contains 210 time-stamped records, comprising 30 observations per day for seven days. Data quality, overall and day-wise descriptive statistics, environmental demand classes, and functional tests are examined. The illustrative dataset was complete, with an overall mean temperature of 29.81 °C (SD = 1.83) and mean relative humidity of 64.29% (SD = 7.17). Day 4 was warmest (30.91 °C) and least humid (58.90%), while Day 1 was coolest (28.66 °C) and most humid (68.39%). The irrigation-support distribution contained 64 low-, 72 moderate-, and 74 high-demand records. The prototype architecture passed the reported functional checks for sensor initialization, scheduled acquisition, timestamping, data transmission, range screening, fail-safe relay state, manual override, and interruption handling. The findings support the feasibility of low-cost environmental monitoring and interpretable decision support. However, the numerical observations in the source dissertation are simulated for academic illustration because raw Arduino records were unavailable.

Keywords: *Smart Irrigation; Arduino; Internet of Things; Artificial Intelligence; Temperature; Relative Humidity; Precision Agriculture.*

1. INTRODUCTION AND RELATED WORK

Agriculture is the largest user of freshwater in many regions, yet conventional irrigation commonly follows fixed schedules or visual judgement. Such practices may supply water when it is unnecessary or may fail to respond when crops experience stress. Smart irrigation addresses this limitation by combining sensing, communication, data analysis, and automated or assisted control. The aim is to apply water at an appropriate time and quantity while reducing losses through runoff, deep percolation, and avoidable evaporation.

The Internet of Things provides the sensing and communication layer. Temperature, humidity, soil moisture, rainfall, tank level, and water-flow sensors can transmit observations to a controller or cloud platform. Artificial intelligence adds pattern recognition, prediction, and decision support. Machine-learning models can classify irrigation need or estimate future moisture conditions from present and historical measurements. When connected to a relay, pump, or valve, the decision may be converted into a controlled irrigation action.

Recent literature shows growing interest in IoT-enabled, cloud-connected, and AI-assisted irrigation. Reviews emphasize real-time monitoring, communication protocols, sensor calibration, model selection, and resource conservation (Gamal et al., 2023; Vallejo-Gomez et al., 2023). Tace et al. (2022, 2023) demonstrated the combination of IoT and machine learning, while Behzadipour et al. (2023) showed that sensor and image information can support predictive irrigation and reduced water use. Morchid et al. (2024) highlighted embedded systems, telemetry, cloud dashboards, and alert mechanisms. Wei et al. (2024) stressed human-centred and explainable AI so that recommendations remain understandable to farmers.

1.1 Research Gap and Objectives

Many proposed systems report strong predictive performance but rely on larger datasets, multiple sensors, or controlled environments. Low-cost educational prototypes require a transparent design that distinguishes demonstrated capability from future field claims. This study therefore focuses on an affordable Arduino platform, a limited temperature–humidity dataset, and a reproducible analytical workflow.

- Design a layered AI–IoT architecture for environmental monitoring and irrigation control.
- Integrate sensing, control, communication, storage, analysis, and protected actuation.
- Analyse seven-day temperature and humidity patterns and classify environmental irrigation demand.
- Evaluate data quality and functional behaviour while defining the limits of the evidence.
- Recommend a path toward crop-specific, water-saving field validation.

2. PROPOSED SYSTEM ARCHITECTURE

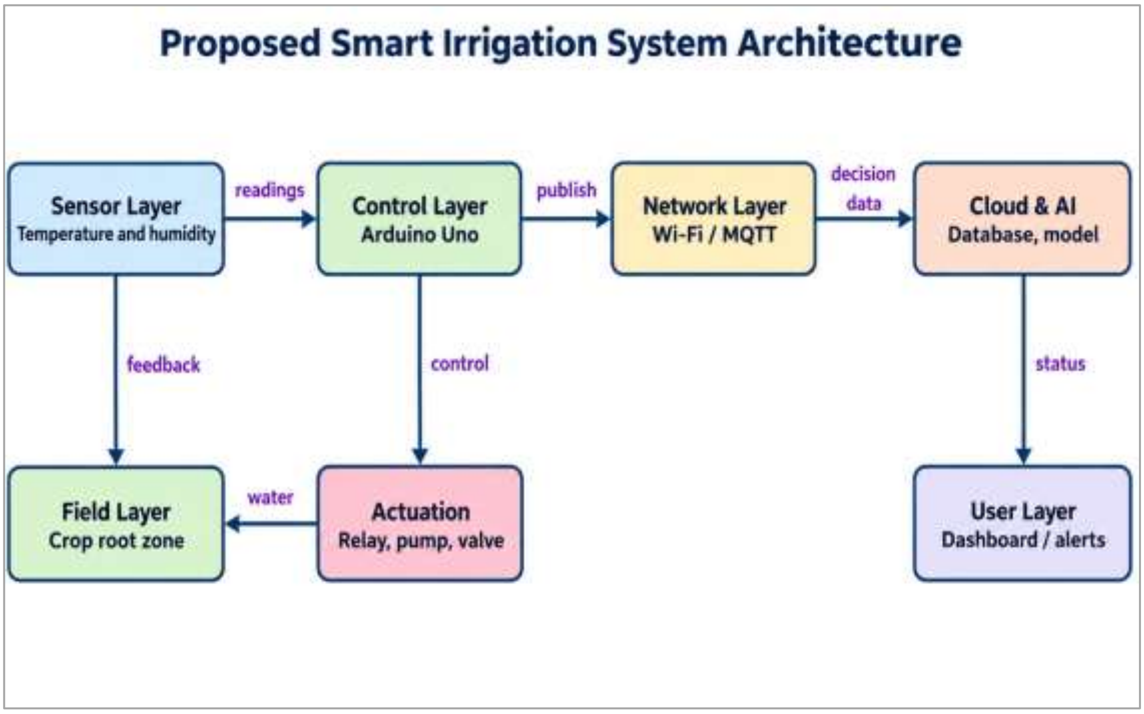

Figure 1. Proposed Layered Architecture for The Smart Irrigation System

The architecture separates the system into sensor, control, network, cloud and AI, user, actuation, and field layers. The sensor layer supplies environmental readings to the Arduino Uno. The controller validates and formats the observations before publishing them through Wi-Fi or MQTT. The cloud layer stores data and provides analytics or an AI recommendation, while the user layer displays status and alerts. A protected relay controls the pump or valve; its default state is OFF during restart or fault conditions.

Layer	Main Element	Function
Sensing	Temperature–humidity sensor	Collects atmospheric observations
Control	Arduino Uno	Acquires data and applies safety rules
Communication	Wi-Fi, MQTT, or serial link	Transfers time-stamped records
Analytics	Database and simple AI model	Summarises data and estimates demand
Actuation	Relay, pump, or valve	Executes approved irrigation action
Interface	Dashboard or serial monitor	Shows readings, alerts, and system state

The modular arrangement allows a sensor, communication method, or analytical model to be replaced without redesigning the entire system. It also keeps the monitoring result separate from the irrigation command, which is important when evidence is limited.

3. MATERIALS AND METHODS

A quantitative design-and-evaluate approach was used. The study setting was Ratu Road, Ranchi, Jharkhand. The planned unit of analysis was one time-stamped record. Thirty records were scheduled at approximately equal intervals on each of seven consecutive days, producing 210 observations. The available analytical variables were ambient temperature, relative humidity, observation day, record sequence, and a derived irrigation-support class.

Design Element	Specification
Platform	Arduino Uno with digital temperature–humidity sensor
Study duration	Seven consecutive days
Sampling plan	30 records per day; 210 total records
Primary measures	Temperature in °C and relative humidity in %
Storage	CSV, spreadsheet, serial logger, or IoT database
Analysis	Quality checks, descriptive statistics, daily trends, classification
Safety	Fail-safe OFF, manual override, maximum runtime, protected relay

3.1 Data Preparation and Decision Logic

Records were screened for missing values, duplicate day–record combinations, impossible ranges, and timestamp consistency. Transparent features included the current temperature and humidity, daily mean, within-day range, change from the preceding record, and time category. Environmental irrigation demand was interpreted as low, moderate, or high. Warm and comparatively dry conditions indicated higher atmospheric demand, but the class was treated as decision support rather than direct proof of root-zone water shortage.

3.2 Evaluation

Evaluation covered completeness, means, standard deviations, minima, maxima, day-wise comparison, combined temperature–humidity patterns, class frequencies, and functional tests. For later empirical work, model performance should include accuracy, precision, recall, F1-score, confusion matrix, and held-out validation. Water saving must be calculated from calibrated flow measurements against a conventional comparator. Because the source did not include raw Arduino exports, the numerical results reported below remain illustrative.

4. RESULTS

4.1 Data Quality and Overall Description

All 210 planned records were present, with 30 observations on each of seven days. No missing temperature or humidity values and no duplicate day–record combinations were reported. All values were within the stated illustrative ranges of 23.5–35.0 °C for temperature and 42–88% for relative humidity.

Variable	N	Mean	SD	Minimum	Maximum
Temperature (°C)	210	29.81	1.83	25.7	33.2
Relative humidity (%)	210	64.29	7.17	51.0	81.7

4.2 Day-Wise Environmental Results

Day	Mean temp. °C	SD	Mean RH %	SD	Demand
1	28.66	1.56	68.39	6.54	Lower/moderate
2	29.53	1.66	65.41	6.89	Lower/moderate
3	30.35	1.76	62.05	6.09	Higher
4	30.91	1.68	58.90	6.13	Higher
5	29.88	1.83	64.93	6.98	Higher
6	29.08	1.80	67.04	7.43	Lower/moderate
7	30.28	1.63	63.31	6.19	Higher

Day 4 had the highest mean temperature and lowest mean humidity, indicating the strongest atmospheric demand among the seven days. Day 1 had the lowest mean temperature and highest mean humidity. The source also reports statistically significant daily differences for both variables, although complete test statistics are not available in the dissertation tables and are therefore not reconstructed here.

4.3 Temporal Patterns

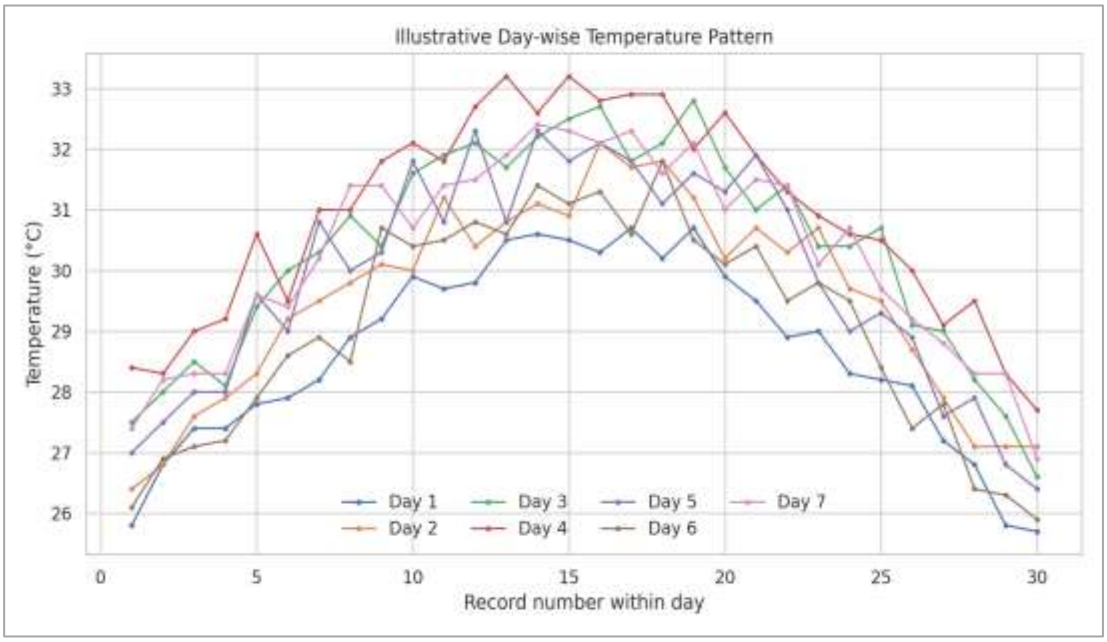

Figure 2. Illustrative Day-Wise Temperature Pattern Across 30 Observations Per Day

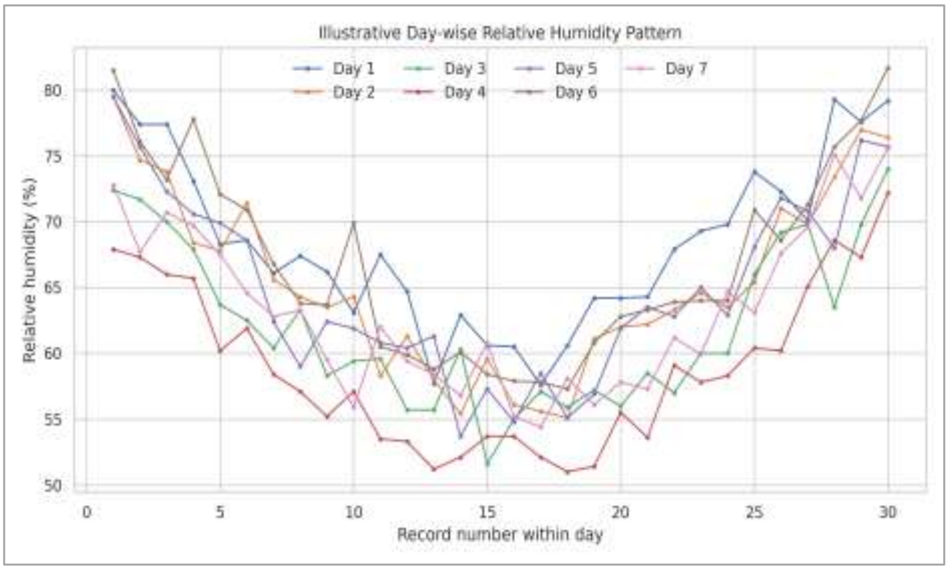

Figure 3. Illustrative Day-Wise Relative-Humidity Pattern Across 30 Observations Per Day

The curves show a repeated daily cycle. Temperatures generally increased from morning toward midday or afternoon and then declined, while relative humidity generally moved in the opposite direction. Frequent observations therefore captured variation that would be missed by a single daily measurement.

4.4 Irrigation Support and Prototype Tests

The derived environmental-demand classification contained 64 low-need observations (30.5%), 72 moderate-need observations (34.3%), and 74 high-need observations (35.2%). The near-balanced distribution is suitable for demonstrating a three-class analytical workflow. However, the classes were created mainly from temperature and humidity, the same variables that would be used as predictors. Apparent classification performance would therefore be optimistic and cannot substitute for independent agronomic labels.

Class	Frequency	Percentage	Interpretation
Low	64	30.5%	Cooler and/or more humid condition
Moderate	72	34.3%	Intermediate environmental demand
High	74	35.2%	Warmer and/or drier condition

4.5 Functional Results

Functional Test	Reported Outcome	Status
Sensor initialization	Sensor detected by Arduino	Pass
Scheduled acquisition	30 records generated per day	Pass
Timestamp attachment	Each record included day and time	Pass
Data transmission	Structured record reached storage	Pass
Invalid-value check	Out-of-range values were flagged	Pass
Relay restart state	Relay remained OFF	Pass
Manual override	User command changed demonstration state	Pass
Communication interruption	Local monitoring continued or stopped safely	Pass

These tests demonstrate feasibility of the proposed monitoring and control pathway. They should be supported by dated device logs and actual sensor exports before final empirical submission.

5. DISCUSSION

The results show that a low-cost Arduino platform can organize repeated environmental measurements into a useful monitoring record. The seven-day structure captured both within-day and between-day variation. Day 4 combined the highest temperature with the lowest humidity, while Day 1 showed the reverse pattern. This supports the practical value of considering temperature and humidity together: warm, dry air generally creates greater evaporative demand than warm, humid air.

The prototype aligns with current smart-irrigation research in which IoT devices collect field information, cloud or local services preserve and analyse records, and decision models translate observations into actions. Its main strength is interpretability. A farmer or researcher can examine the readings, understand why the system assigned a demand class, and retain manual authority over the pump. Such transparency is preferable to complex black-box control when only 210 observations are available.

At the same time, atmospheric demand is not identical to crop water need. Temperature and relative humidity do not reveal root-zone moisture, recent rainfall, soil texture, crop stage, solar radiation, wind, or the amount of water already applied. The classification is therefore best understood as an environmental advisory signal. Autonomous irrigation would require direct soil and flow evidence, carefully calibrated thresholds, independent labels, and fault protection.

5.1 Validity of the Reported Results

The most important qualification is that the dissertation labels Chapter 4 values as simulated for academic illustration. The tables and figures demonstrate the intended analysis, not verified measurements from the physical device. Moreover, the study lacks a conventional-irrigation comparator, calibrated flow records, energy consumption, and crop-growth outcomes. It is consequently not possible to estimate water savings, operating cost, yield improvement, or true predictive accuracy. These limits do not remove the value of the design; they define its correct status as a proof-of-concept framework awaiting empirical validation.

5.2 Practical Significance

For small farms and educational laboratories, the design offers an understandable entry point into precision agriculture. It supports time-stamped monitoring, remote display, alerts, modular sensor addition, and controlled actuation. In Ranchi, field deployment should also account for monsoon rainfall, seasonal humidity, intermittent power, enclosure protection, network coverage, and differences among soils and crops.

6. CONCLUSION AND RECOMMENDATIONS

This study presents a coherent AI–IoT smart-irrigation prototype built around Arduino Uno, temperature–humidity sensing, digital storage, simple classification, and relay-based control. The illustrative 210-record dataset was complete and displayed interpretable daily variation. Mean temperature was 29.81 °C and mean relative humidity was 64.29%. Day 4 represented the warmest and driest daily condition, whereas Day 1 was coolest and most humid. The environmental classes were distributed across low, moderate, and high demand, and all listed prototype functional checks were reported as passed.

The evidence supports technical feasibility for environmental monitoring and decision support, but it does not establish agronomic effectiveness. The numerical data must be replaced or verified with actual Arduino exports. The system should not independently irrigate crops on the basis of temperature and humidity alone.

6.1 Recommendations

- Add calibrated capacitive soil-moisture probes at the active crop root depth.
- Measure rainfall, water flow, tank level, irrigation duration, and electrical energy.
- Collect data for at least one complete crop cycle and preferably across seasons.
- Use comparable treatment and conventional-irrigation plots to estimate water saving and crop response.
- Create irrigation labels from soil measurements and agronomic verification rather than predictor-derived rules.
- Reserve future days or another season as an independent AI test set and report confidence intervals.
- Maintain fail-safe OFF control, maximum pump runtime, manual override, secure commands, and fault logs.
- Evaluate cost, maintenance burden, farmer understanding, and multilingual dashboard usability.

6.2 Future Scope

Future work can expand the prototype into a multi-zone, weather-aware system using soil moisture, flow, rain, and tank-level sensors. Interpretable random forests, gradient boosting, or compact neural networks may be considered after a sufficiently large labelled dataset is collected. LoRa or cellular communication, solar power, edge inference, and a Hindi or regional-language mobile interface could improve rural deployment. Multi-location trials across Jharkhand are necessary before recommending broad adoption.

Data and Ethical Statement

No personal data were required. The numerical dataset reported in this paper is illustrative and should not be deposited or cited as observed field data. Actual future datasets should preserve device metadata, calibration records, missing-value flags, model versions, and user privacy.

REFERENCES

1. Allen, R. G., Pereira, L. S., Raes, D., & Smith, M. (1998). Crop evapotranspiration: Guidelines for computing crop water requirements. Food and Agriculture Organization.
2. Arduino. (2026). Arduino UNO R3. https://docs.arduino.cc/hardware/uno-rev3
3. Behzadipour, F., Ghasemi Nezhad Raeini, M., Abdanan Mehdizadeh, S., Taki, M., Khalil Moghadam, B., Zare Bavani, M. R., & Lloret, J. (2023). A smart IoT-based irrigation system design using AI and prediction model. Neural Computing and Applications, 35(35), 24843–24857.
4. Breiman, L. (2001). Random forests. Machine Learning, 45, 5–32.
5. Food and Agriculture Organization. (2017). Water for sustainable food and agriculture. FAO.
6. Gamal, Y., Soltan, A., Said, L. A., Madian, A. H., & Radwan, A. G. (2023). Smart irrigation systems: Overview. IEEE Access, 13, 66109–66121.
7. Morchid, A., Jebabra, R., Khalid, H. M., El Alami, R., Qjidaa, H., & Jamil, M. O. (2024). IoT-based smart irrigation management system to enhance agricultural water security using embedded systems, telemetry data, and cloud computing. Results in Engineering, 23, 102829.
8. OASIS. (2019). MQTT version 5.0: OASIS standard. https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html
9. Pedregosa, F., et al. (2011). Scikit-learn: Machine learning in Python. Journal of Machine Learning Research, 12, 2825–2830.
10. Sami, M., Khan, S. Q., Khurram, M., Farooq, M. U., Anjum, R., Aziz, S., et al. (2022). A deep learning-based sensor modeling for smart irrigation system. Agronomy, 12(1), 212.

11. Tace, Y., Elfilali, S., Tabaa, M., & Leghris, C. (2023). Implementation of smart irrigation using IoT and artificial intelligence. Mathematical Modeling and Computing, 10(2), 575–582.
12. Tace, Y., Tabaa, M., Elfilali, S., Leghris, C., Bensag, H., & Renault, E. (2022). Smart irrigation system based on IoT and machine learning. Energy Reports, 8, 1025–1036.
13. Vallejo-Gomez, D., Osorio, M., & Hincapie, C. A. (2023). Smart irrigation systems in agriculture: A systematic review. Agronomy, 13(2), 342.
14. Wei, H., Xu, W., Kang, B., Eisner, R., Muleke, A., Rodriguez, D., et al. (2024). Irrigation with artificial intelligence: Problems, premises, promises. Human-Centric Intelligent Systems, 4(2), 187–205.