University of L'Aquila

Information Engineering Computer Science and Mathematics

Software Engineering for Autonomous Systems

Project Report
Poultry Farm Management

Team Members

Abu Saleh

Aruzhan Amangeldina

Tesfay W. Tesfay

Submitted To

Professor Davide Di Ruscio

L'Aquila, 02 February 2026

Contents

1 Introduction

Modern poultry farming requires precise and continuous control of environmental conditions to ensure animal welfare, productivity, and economic sustainability. Poultry birds are highly sensitive to changes in temperature and air quality, and even brief deviations from optimal conditions can lead to stress, illness, or mortality. As farms scale to include multiple large barns, manual monitoring and control become impractical and error-prone. Environmental regulation in poultry houses is a complex task due to strong interactions between physical, biological, and mechanical processes. Ventilation affects not only temperature but also carbon dioxide and ammonia concentrations, while bird activity contributes additional heat and gas emissions. These interdependencies create a dynamic system that cannot be effectively managed using static or rule-based control alone, particularly under changing external conditions or equipment failures.

Self-Adaptive Systems (SAS) offer a systematic approach to handling such complexity. By continuously monitoring system state, analyzing deviations from desired goals, planning corrective actions, and executing control decisions, SAS can autonomously maintain stable operating conditions. The MAPE-K feedback loop provides a well-established architectural model for implementing this adaptive behavior. This project presents an Autonomous Poultry Farm Manager that applies the MAPE-K architecture to a simulated poultry house environment. The system integrates a physics- and biology-based simulator with distributed monitoring and control services, enabling safe experimentation and evaluation of adaptive control strategies. The goal is to demonstrate how self-adaptive system principles can be applied to a real-world, safety-critical agricultural domain.

2 Objectives

The main objective of this project is to design and implement a self-adaptive system capable of autonomously managing environmental conditions in a poultry farm. The system aims to maintain optimal conditions for animal welfare while ensuring stability, robustness, and efficient resource usage. The specific objectives of this work are as follows:

- Design a Physics-Based Simulation Environment to develop a realistic simulation of a poultry house that models thermodynamic behavior, air quality dynamics (CO_2 and ammonia), and biological processes such as bird heat generation and activity.
- Apply the MAPE-K Self-Adaptive Architecture To implement the Monitor, Analyze, Plan, Execute, and Knowledge components as distributed services, enabling continuous feedback-driven adaptation to environmental changes.
- Enable Autonomous Environmental Regulation To autonomously control actuators such as fans, heaters, inlets, lighting, feed dispensers, and water valves based on real-time sensor data and predefined policies.
- Ensure System Stability and Safety To incorporate control mechanisms such as hysteresis

and proportional controllers to prevent oscillations, actuator conflicts, and unsafe operating conditions.

- Support Configurability and Scalability To allow multiple farms or zones to be managed through centralized configuration, enabling flexible experimentation with different environmental thresholds and physical parameters.
- Provide a Testbed for Adaptive Control Research To create a reusable simulation platform for evaluating self-adaptive control strategies in a safety-critical agricultural domain without risking real animals or infrastructure.

3 Functional Requirements

Functional requirements describe the specific behaviors and capabilities that the system must provide in order to fulfill its intended purpose. They define what the system should do in terms of data collection, processing, storage, visualization, and alerting.

4 Non-Functional Requirements

Non-functional requirements define the quality attributes and constraints of the system rather than specific behaviors. They describe how the system should perform, focusing on aspects such as performance, reliability, scalability, security, and maintainability.

5 System Architecture

The Autonomous Poultry Farm Manager is designed as a distributed self-adaptive system following the MAPE-K (Monitor, Analyze, Plan, Execute, Knowledge) architectural pattern. This architecture enables continuous feedback-driven adaptation while maintaining a clear separation of concerns between sensing, decision-making, and actuation. At the core of the system is the Managed System, which represents the poultry farm environment. In this work, the managed system is implemented as a simulation that models the physical and biological dynamics of a poultry house, including temperature evolution, air quality (CO_2 and ammonia), bird activity, and resource consumption. The environment exposes its state through simulated sensors and accepts control commands through simulated actuators.

5.1 Monitor

The Monitor component is responsible for collecting sensor data published by the environment via an MQTT message bus. Its primary role is data ingestion and normalization. All incoming sensor readings are stored in a time-series database, forming the system's historical knowledge and enabling later analysis and visualization.

Table 1: Functional Requirements of the Autonomous Poultry Farm Manager

ID	Name	Description
FR1	Sensor Data Acquisition	The system shall continuously collect environmental data including temperature, CO_2, ammonia, feed level, water level, and bird activity from the simulated environment.
FR2	Environment Simulation	The system shall simulate poultry house physics and biology, including heat transfer, gas accumulation, bird metabolism, and resource consumption over time.
FR3	Data Storage	The system shall store all sensor readings and system states in a time-series knowledge base for historical analysis and decision-making.
FR4	Symptom Detection	The system shall analyze sensor data to detect abnormal or critical conditions based on predefined thresholds and policies.
FR5	Autonomous Planning	The system shall generate control plans to resolve detected symptoms by selecting appropriate actuator actions.
FR6	Actuator Control	The system shall control actuators including fans, heaters, inlets, lights, feed dispensers, and water valves according to generated plans.
FR7	Feedback Loop Execution	The system shall continuously execute the MAPE-K feedback loop to adapt to changing environmental conditions in real time.
FR8	Hysteresis Control	The system shall apply hysteresis logic to prevent rapid actuator toggling and ensure stable control behavior.
FR9	Proportional Control	The system shall support proportional control strategies to adjust actuator intensity based on deviation from target conditions.
FR10	Configurability	The system shall allow modification of environmental thresholds, physical parameters, and farm layout through a centralized configuration file.
FR11	Manual Override	The system shall allow manual actuator commands via MQTT while maintaining autonomous conflict resolution.
FR12	Multi-Zone Support	The system shall support multiple farms or zones operating concurrently with independent sensing and control.

5.2 Analyzer

The Analyzer component periodically queries the knowledge base to evaluate the current system state against predefined thresholds and policies. Rather than issuing direct control commands, the Analyzer detects high-level symptoms such as abnormal temperature or poor air quality and publishes these assessments as status events. This abstraction simplifies decision-making and decouples diagnosis from action planning.

Table 2: Non-Functional Requirements of the Autonomous Poultry Farm Manager

ID	Name	Description
NFR1	Performance	The system shall process sensor data and execute control decisions within bounded time intervals to support near real-time adaptation.
NFR2	Scalability	The system shall support the addition of multiple farms or zones without requiring architectural changes.
NFR3	Reliability	The system shall continue operating under partial service failures and recover gracefully from transient faults.
NFR4	Stability	The system shall avoid oscillatory behavior in actuator control by enforcing hysteresis and minimum activation durations.
NFR5	Accuracy	The simulation shall approximate real-world poultry house dynamics with sufficient fidelity to support control strategy evaluation.
NFR6	Modularity	Each MAPE-K component shall be independently deployable and maintainable as a separate service.
NFR7	Configurability	System behavior shall be adjustable through external configuration files without requiring code modifications.
NFR8	Maintainability	The system shall be structured to support easy debugging, testing, and extension of control logic and simulation models.
NFR9	Interoperability	Components shall communicate using standardized messaging protocols to ensure loose coupling and extensibility.
NFR10	Observability	The system shall provide monitoring and visualization of sensor data and control actions for analysis and debugging.
NFR11	Reproducibility	The simulation shall produce repeatable results under identical configurations and input conditions.
NFR12	Security	The system shall restrict unauthorized access to control interfaces and configuration parameters.

5.3 Planner

The Planner component receives status events and determines appropriate corrective actions. It applies control strategies such as proportional controllers and hysteresis-based rules to compute stable and efficient responses. The Planner produces high-level plans that specify desired actuator behavior without binding to low-level execution details.

5.4 Executor

The Executor component translates these plans into concrete actuator commands and dispatches them to the environment through MQTT. This layer abstracts hardware-specific details and ensures consistent command execution across different zones or farms.

5.5 Knowledge

The Knowledge component underpins all adaptation processes. It consists of a time-series database for storing sensor history, a centralized configuration file defining thresholds and physical parameters, and retained MQTT topics for sharing the current system state. This shared knowledge enables learning, traceability, and reproducibility.
Communication between all components is achieved through a lightweight publish–subscribe messaging architecture using MQTT. This design supports loose coupling, scalability, and fault isolation, making the system extensible to multiple farms, zones, or future control strategies.

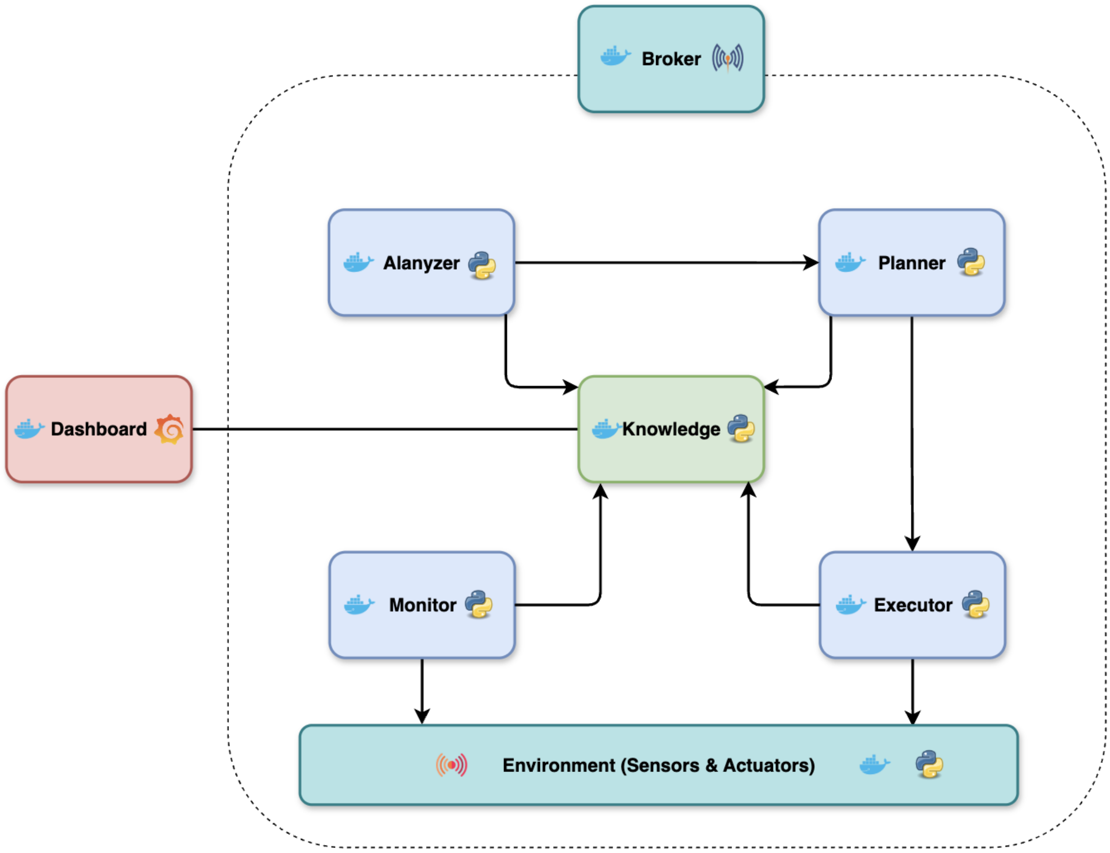

Figure 1: System Architecture

6 System Taxonomy

- **Reason**: *Change in Context* and *User Interaction*. The system continuously adapts to dynamic environmental conditions (temperature, ammonia) (Context) and accommodates manual operator overrides (User).
- **Level**: *Application Level and Context*. The adaptation logic is implemented as high-level software microservices (Planner, Analyzer), external to the operating system or physical hardware.

- **Time**: *Reactive.* The system operates on a feedback loop that triggers corrective actions only after monitoring detects threshold violations, rather than proactively forecasting them.

- **Technique**: *Parameter Adaptation and Context.* The system modifies configurable parameters (e.g., fan speed 0–100%, heater intensity) to achieve goals, without altering the software architecture or structure.

- **Control**: *External, Centralized, Rules and Goals based.* A distinct MAPE-K manager controls the environment, with decision-making centralized in a single Planner service rather than distributed across independent actuators.

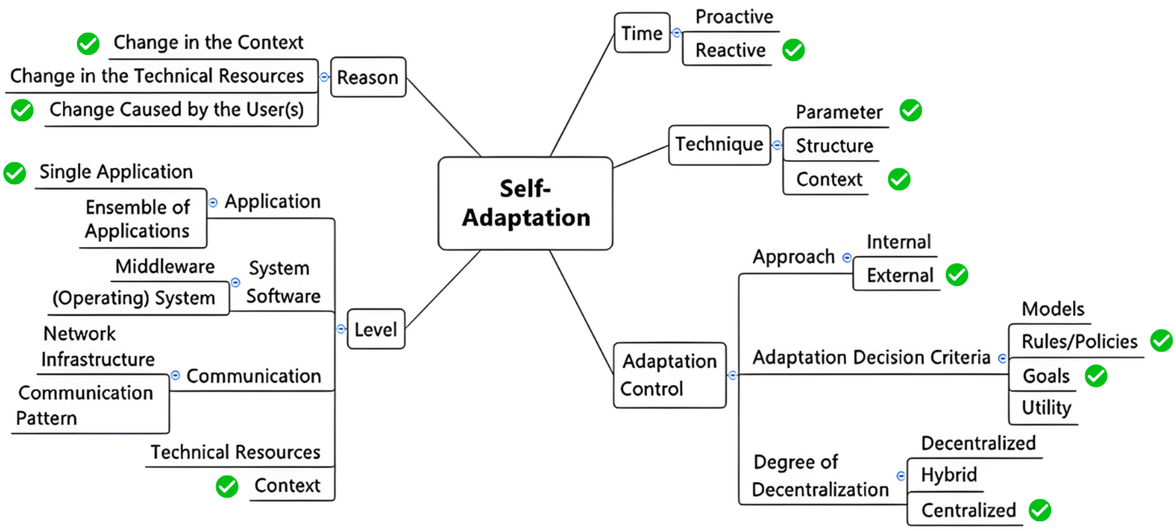

Figure 2: System Taxonomy

7 Managed Resources

The managed resources of the Autonomous Poultry Farm Manager consist of the sensing and actuation elements that interface directly with the poultry house environment. These resources form the boundary between the adaptive control system and the physical (simulated) farm, enabling observation of environmental conditions and execution of control actions.

7.1 Sensors

Sensors provide continuous observations of the environmental and operational state of the poultry house. In this system, sensors are simulated to reflect real-world behavior, including noise, delays, and gradual state changes. The primary sensors include:

- Temperature Sensor: Measures the ambient air temperature within the poultry house. This sensor is critical for detecting heat or cold stress conditions and directly influences fan and heater control.
- Carbon Dioxide CO_2 Sensor: Monitors CO_2 concentration generated by bird respiration and insufficient ventilation. Elevated levels indicate inadequate air exchange.
- Ammonia NH_3 Sensor: Measures ammonia concentration resulting from waste decomposition. This sensor reflects long-term air quality and ventilation effectiveness.
- Feed Level Sensor: Reports the remaining feed quantity in the feeder, enabling autonomous feed replenishment.
- Water Level Sensor: Measures the available water level in drinkers or storage tanks to ensure continuous access for birds.
- Bird Activity Sensor: Estimates the overall activity level of the flock, which indirectly affects heat generation, gas emissions, and resource consumption.

7.2 Actuators

Actuators enable the system to influence the environment in response to detected conditions. Similar to sensors, actuators are simulated with realistic constraints such as delayed effects and minimum activation durations. The system manages the following actuators:

- Heater: Increases ambient temperature to prevent cold stress during low-temperature conditions.
- Fan: Provides ventilation to reduce temperature and remove excess CO_2 and ammonia from the poultry house.
- Inlet: Controls the intake of fresh air, working in conjunction with the fan to regulate airflow and air quality.
- Light: Regulates lighting cycles to influence bird activity, feeding behavior, and circadian rhythms.
- Feed Dispenser: Dispenses feed when low feed levels are detected, typically operating in discrete activation intervals.
- Water Valve: Controls water supply to maintain adequate water levels for the flock.

8 Control Strategy and Environment Model

This section describes the decision-making logic implemented in the Planner component and the physical rules governing the simulated poultry house environment. Together, these models define how the system adapts to changing conditions and how control actions affect the environment over time.

8.1 Planner Control Strategy

The Planner component implements a feedback-based control strategy to maintain environmental conditions within predefined limits. Control decisions are derived from sensor data and translated into actuator commands using a combination of proportional control and hysteresis mechanisms.

8.1.1 Fan Control

Fan operation is governed by a proportional-only (P) controller based on temperature and carbon dioxide (CO_2) deviations from their respective setpoints. The fan control signal is computed as:

$$\text{Fan}_{level} = K_p^T \cdot \max(0, T - T_{set}) + K_p^{CO_2} \cdot \max(0, CO_2 - CO_{2,set})$$

To ensure safety and robustness, several modifiers are applied. If ammonia (NH_3) concentration exceeds a critical threshold, an additional fan boost is applied. During cold conditions, fan output is capped to prevent excessive heat loss. When the heater is active, a minimum fan level is enforced to guarantee sufficient air circulation. All fan commands are constrained within predefined bounds and subject to rate limiting to avoid abrupt changes.

8.1.2 Heater Control

Heater activation is managed using a hysteresis-based controller to prevent rapid on–off cycling. The heater is activated when the temperature falls below a lower threshold and deactivated when it exceeds an upper threshold around the setpoint. Minimum on and off durations are enforced to protect system stability.

When active, heater intensity is determined by a proportional controller:

$$\text{Heater}_{level} = \min\left(100, K_p^H \cdot (T_{set} - T)\right)$$

A minimum heater output is maintained while the heater is on.

8.1.3 Inlet Control

The inlet opening percentage is coupled with fan speed to regulate fresh air intake. The baseline inlet position increases proportionally with fan level and is further adjusted when CO_2 or NH_3 concentrations exceed their thresholds. Inlet commands are constrained to safe operating limits and rate limited to ensure smooth transitions.

8.1.4 Feed and Water Control

Feed and water replenishment are controlled using bang–bang (hysteresis) logic. Refill actions are triggered when resource levels fall below a lower threshold and deactivated once an upper threshold is reached, preventing excessive actuator toggling.

8.1.5 Light Control

Lighting is regulated using a proportional controller based on bird activity levels. The system increases lighting intensity when activity drops below a minimum threshold and reduces lighting when activity becomes excessive. A day–night cycle is enforced by defining minimum light levels for daytime and nighttime periods.

8.2 Environment Model

The environment model simulates the physical and biological dynamics of a poultry house using mass and energy balance equations. State updates are performed at fixed time intervals.

8.2.1 Temperature Dynamics

Temperature evolution is driven by heat gains from heaters and birds and heat losses due to conduction and ventilation:

$$\frac{dT}{dt} = \frac{Q_{heater} + Q_{birds} - Q_{loss} - Q_{vent}}{C_{thermal}}$$

where bird heat production depends on flock size and activity level, and ventilation losses depend on airflow rate and indoor–outdoor temperature difference.

8.2.2 Carbon Dioxide Dynamics

CO_2 concentration is modeled as a balance between biological generation and removal through ventilation:

$$\frac{d[CO_2]}{dt} = G_{CO_2} - V_{CO_2}$$

where generation increases with bird activity and ventilation removes CO_2 proportionally to airflow rate.

8.2.3 Ammonia Dynamics

Ammonia concentration evolves according to generation from waste, ventilation removal, and natural decay:

$$\frac{d[NH_3]}{dt} = G_{NH_3} - V_{NH_3} - D_{NH_3}$$

Generation rates increase with temperature and bird activity, while decay models chemical decomposition over time.

8.2.4 Ventilation Flow Rate

The effective ventilation flow rate is determined by fan speed and inlet opening:

$$\text{Flow} = \text{Flow}_{inf} + \text{Flow}_{max} \cdot \frac{\text{Fan}_{level}}{100} \cdot \left(0.2 + 0.8 \cdot \frac{\text{Inlet}_{pct}}{100}\right)$$

8.3 Analyzer Thresholds

The Analyzer component evaluates sensor readings against predefined configuration limits to detect abnormal conditions. Thresholds define acceptable operating ranges for temperature, air quality, resource availability, and bird activity, enabling timely symptom detection and adaptive responses.

9 Monitoring Dashboard

To support system observability and evaluation, a monitoring dashboard is provided for real-time visualization of the poultry farm environment and control behavior. The dashboard is implemented using Grafana and is directly connected to the system's time-series knowledge base. The dashboard presents live and historical data for key environmental variables, including temperature, carbon dioxide, ammonia, feed and water levels, and bird activity. Actuator states such as fan speed, heater level, inlet position, and lighting intensity are also visualized, enabling clear inspection of control decisions over time. These visualizations allow operators and researchers to verify system behavior, analyze adaptation patterns, and identify potential anomalies. Screenshots of the dashboard are included to illustrate system performance under different operating conditions.

Figure 3: Sensors Dashboard

10 Conclusion

This report presents the design and implementation of an Autonomous Poultry Farm Manager based on the MAPE-K self-adaptive architecture. The system integrates a physics- and biology-based environment simulator with distributed monitoring, analysis, planning, and execution components to autonomously regulate environmental conditions in a poultry house. Through the

Figure 4: Actuator Level Dashboard

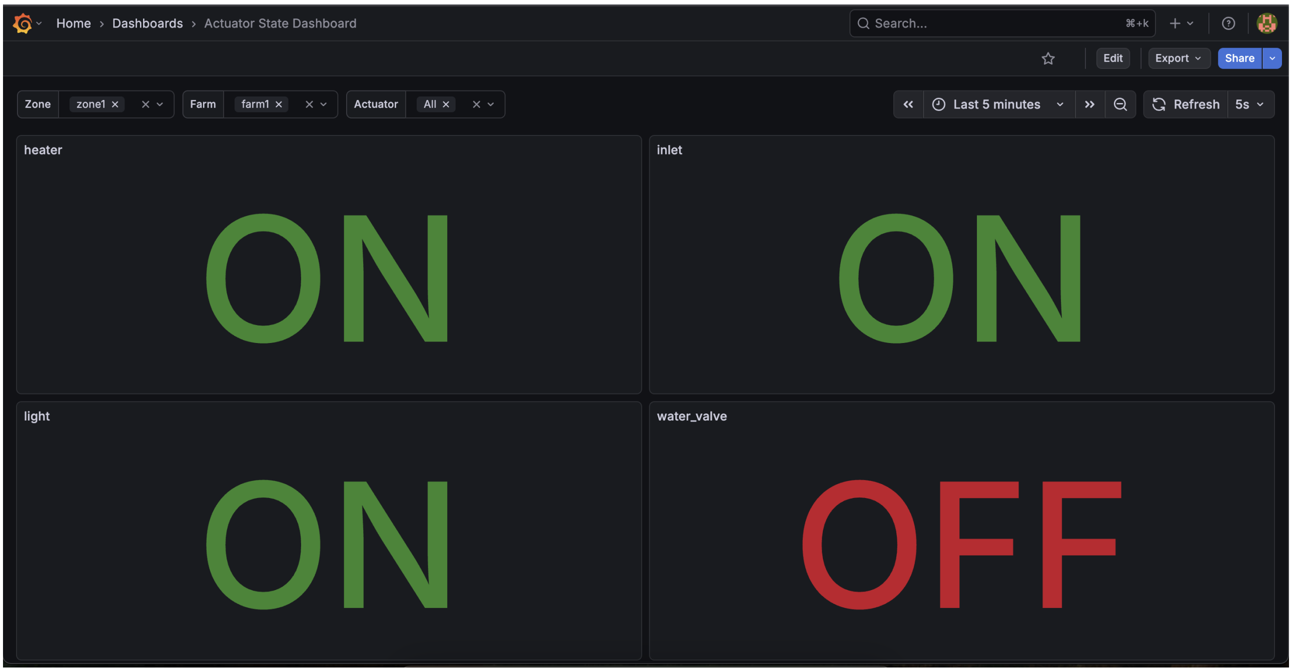

Figure 5: Actuator State Dashboard

use of proportional controllers, hysteresis mechanisms, and coupled environment models, the system demonstrates stable and realistic adaptive behavior in response to changing temperature, air quality, and resource levels. The simulator enables safe experimentation and evaluation of control strategies without risking animal welfare, while the modular microservice architecture supports scalability and extensibility.

Overall, this work illustrates the applicability of self-adaptive systems to safety-critical agricultural domains and provides a practical foundation for future research on intelligent control, learning-based adaptation, and real-world deployment in smart farming environments.

Project Repository

The full source code, configuration files, and Docker setup for this project are publicly available at: https://github.com/Saleh7127/SE4AS_PoultryFarmManager