Program Architecture and Design Requirements

1. Functional Requirements (grouped by domain)

I. ***Sensor monitoring:*** continuous monitoring of temperature, humidity, and motion sensors, log sensor data periodically.

II. Actuator Control: automated control of heating, humidity and egg turning based on thresholds. Remote manual control for heating ventilation and egg turning.

III. Data storage and processing: store data in DBMS, use prediction algorithms to estimate optimal hatching times and detect anomalies.

IV. Remote access: provide graphs showing log data and the predicted future values on the mobile user interface, include user input handling for controlling the thresholds manually.

V. Alerts and notifications: send alerts for anomalies (low/high temperature or humidity), notify user when hatching or manual intervention is required.

VI. User Interface: real-time data visualization with graphical trends and prediction graphs, control buttons for remote setup of the operation modes (constants/thresholds).

2. Quality Attributes

I. Performance: [HIGH PRIORITY]

- ✓ The system shall respond to actuator control commands within 2 seconds of receiving sensor data or user input, ensuring real-time operation.
- ✓ The server shall process prediction computations within 500 milliseconds for datasets of up to 10,000 records.(minimum standard for simple computing machines)

II. Scalability: [HIGH PRIORITY]

- ✓ The system shall support the addition of up to 10 sensors and actuators without performance degradation.(can scale for 10 more products)
- ✓ The back-end shall handle up to 1,000 concurrent users accessing the mobile application and database.(multi-user interface minimum requirement)

III. Reliability: [HIGH PRIORITY]
    - ✓ The system shall operate with an uptime of 99.9% under standard conditions.
    - ✓ Communication between the micro-controller and server shall succeed in at least 95% of attempts during network fluctuations.(4.9% fault should be handled)

IV. Maintainability: [MIDIUM PRIORITY]
    - ✓ The system's modular codebase shall allow for the addition or removal of a feature within 4 hours by a qualified developer.
    - ✓ Each code module shall be documented to provide an average readability score of at least 85% (measured by software documentation standards of Flesch-Kincaid).

V. Security: [LOW PRIORITY]
    - ✓ The mobile application shall implement two-factor authentication with a success rate of 99% during penetration tests.

VI. Usability: [HIGH PRIORITY]
    - ✓ The mobile application interface shall have a user task completion time of less than 10 seconds for common tasks.
    - ✓ The app shall achieve a usability rating of 4.5/5 during user experience testing with a sample size of 10 users(students).