AI & SUSTAINABILITY IN VET EDUCATION

Name of the Author/s: Politeknika Txorierri

Title of WP: WP3 – AI-SAVES@ProjectHub

Activity Reference: Greenhouse Ventilation & Climate Control

Logo:

POLITEKNIKA TXORIERRI

Co-funded by the European Union

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

Table of Contents

1. Project Title: 3
2. Project Category: 3
3. Project Description: 3
4. Keywords: 3
5. Target Audience: 3
6. Project Objectives: 3
7. Project Timeline: 4
8. Resources Provided: 4
9. Tools and Technologies Used: 5
10. Step-by-Step Implementation Plan: 5
11. Evaluation Criteria: 6
12. Languages Available: 7
13. Contributors: 7
14. Feedback and Continuous Improvement: 7

Co-funded by the European Union

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

1. **Project Title:**

   Smart Greenhouse Climate Control with Raspberry Pi 5

2. **Project Category:**

   IoT • Automation • Sustainability

3. **Project Description:**

   This project aims to design and program a smart greenhouse climate control system using a Raspberry Pi 5.

   The system measures temperature with a DHT11 sensor and automatically controls three actuators:

   - A servo motor that opens or closes a ventilation hatch.
   - A fan to simulate air circulation.
   - Three LEDs (red, green, yellow) that indicate temperature ranges.

   Students begin by simulating the system in Tinkercad using an Arduino UNO and a potentiometer to represent temperature.

   Once the control logic is verified, they move to the real implementation with Raspberry Pi 5, where the program is written in Python using the gpiozero and adafruit-circuitpython-dht libraries.

   The goal is to understand how sensors and actuators work together to maintain optimal climate conditions, while reinforcing concepts of automation, programming, and sustainability.

4. **Keywords:**

   Raspberry Pi 5, Python, DHT11, Servo Motor, Fan Control, GPIOZero, Automation, Greenhouse, IoT, Sustainability, Climate Regulation

5. **Target Audience:**

   Vocational students and beginners in Python programming interested in IoT, automation, and environmental sustainability.

6. **Project Objectives:**

   - Simulate and test a simple automation system in Tinkercad.
   - Read temperature data from a DHT11 sensor and apply decision logic.
   - Control multiple actuators (servo, fan, LEDs) simultaneously.

Co-funded by the European Union

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

- Develop step-by-step Python code for real hardware.
- Calibrate and debug an electronic control system.
- Document results and communicate conclusions clearly.

7. Project Timeline:

Task	Estimated Hours	Description
Introduction & planning	1 hour	Project overview and sustainability context
Tinkercad simulation setup	5 hours	Build and wire a virtual Arduino circuit and test and adjust logic
Raspberry Pi hardware setup	2 hours	Assemble real circuit (DHT11, servo, fan, LEDs)
Python coding & testing	4 hours	Develop the main control code step by step
Debugging and calibration	4 hours	Refine thresholds and actuator behaviour
Technical documentation	2 hours	Write a short technical report
Final demonstration & reflection	2 hours	Presentation and analysis of results
Total	**20 hours**	

Co-funded by the European Union

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

8. **Resources Provided:**

- Step-by-step Tinkercad simulation guide (Arduino-based).
- Simplified Python script for Raspberry Pi 5 (using gpiozero).
- Circuit diagram (Fritzing or manual).
- Project report template and rubric.
- Example code snippets and troubleshooting guide.

9. **Tools and Technologies Used:**

- Hardware: Raspberry Pi 5, DHT11 temperature sensor, SG90 servo, fan (controlled via relay or transistor), 3 LEDs (red/green/yellow), resistors, breadboard, jumper wires.
- Software: Tinkercad Circuits, Python 3, gpiozero, adafruit-circuitpython-dht, Geany IDE.
- Design Tools: Fritzing, Excel/Calc (for result tables), word processor for documentation.

10. **Step-by-Step Implementation Plan:**

Phase A – Simulation (Arduino in Tinkercad)

a) Build a virtual Arduino circuit with a potentiometer, servo, and three LEDs.

b) Program logic to simulate temperature behaviour:

- Potentiometer value → simulated temperature.
- Red LED = hot (>25 °C), Green = optimal (20–25 °C), Yellow = cold (<20 °C).
- Servo position changes accordingly (0°, 90°, 180°).

c) Observe and record the results in a table (input value → LED/servo response)

Co-funded by the European Union

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

1. **Phase B – Real Hardware (Raspberry Pi)**

   a) Wire components to the Raspberry Pi using the same logic tested in Tinkercad.

      - DHT11 → GPIO 4
      - Servo → GPIO 12
      - Fan → GPIO 17
      - LEDs → GPIO 24 / 25 / 16

   b) Install required Python libraries:

      sudo apt update
      sudo apt install -y python3-gpiozero python3-libgpiod python3-pip
      pip install adafruit-circuitpython-dhtWire up: DHT11 to GPIO; servo powered by external 5 V (common GND); relay module for heater/fan control; LEDs with resistors.

   c) Write the Python code to read the sensor and control the actuators.
   d) Test the system and check that behaviour matches the simulation results.

2. **Phase C – Debugging, Calibration and Reflection**

   a) Adjust threshold values and servo angles to achieve smooth operation.

   b) Identify causes of instability (sensor errors, wiring, timing).

   c) Document the calibration process and improvements.

3. **Phase D – Validation & Optimization**

   a) Prepare a short report including:

      - Wiring diagram and component list.
      - Code explanation (with comments).
      - Observations and conclusions.

   b) Present the project to the class, demonstrating live how temperature changes affect the system.

Co-funded by the European Union

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.

11. Evaluation Criteria:

- **Simulation (Tinkercad) (20%):** Correct logic and working virtual circuit.
- **Hardware Implementation (20%):** Proper wiring and assembly of Raspberry Pi circuit.
- **Programming (30%):** Stable sensor readings, correct logic, actuator control.
- **Debugging & calibration (10%):** Evidence of testing, problem-solving and adjustment.
- **Documentation & presentation (20%):** Clear report, explanation and demonstration.

12. Languages Available:

English

13. Contributors:

- **Miren Arrien (Project Lead)** – Lead educator responsible for project development and coordination.
- **Enaitz Elejalde (Technical Expert)** – Provided expertise in IOT.
- **Aitor Rey (Technical Expert)** – Provided expertise in IOT.
- **Jokin Goioaga (Technical Expert)** – Provided expertise in IOT.

14. Feedback and Continuous Improvement:

Students reflect on their simulation and hardware results, discussing:

- Differences between the virtual and real environments.
- How threshold adjustments affect energy efficiency and system stability.
- What improvements could make the system more reliable (e.g., humidity-based control, servo speed tuning).

Feedback sessions at the end of each phase promote continuous improvement, teamwork, and sustainability awareness.

Co-funded by the European Union

This project has been funded with support from the European Commission. This communication reflects the views only of the author, and the Commission cannot be held responsible for any use which may be made of the information contained therein.