AI & SUSTAINABILITY IN VET EDUCATION

Name of the Author/s: Politeknika Txorierri

Title of WP: WP3 – AI-SAVES@ProjectHub

Activity Reference: Wildlife water trough

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: 4

2. Project Category: 4

3. Project Description: 4

4. Keywords: 5

5. Target Audience: 5

6. Project Objectives: 5

7. Project Timeline: 5

8. Resources Provided: 6

9. Tools and Technologies Used: 6

10. Step-by-Step Implementation Plan: 6

11. Evaluation Criteria: 9

12. Languages Available: 9

13. Contributors: 9

14. Feedback and Continuous Improvement: 9

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:

Wildlife care through IOT: Smart water trough

2. Project Category:

IOT

3. Project Description:

This project aims to design and implement a smart water trough that automatically controls its water level, monitorizes use by animals and send alerts depending on environmental conditions.

The system uses an ESP32 microcontroller with an ultrasonic distance sensor to track the water level inside the trough's reservoir, ensuring that animals always have access to a safe and reliable drinking point, and a motion detection PIR sensor, which will be used to detect animal activity. When the measured level drops below the minimum threshold, aa alarm notification is automatically sent to the system manager through MQTT. Additionally, animal presence will be monitorized and activity levels (none, low, medium, high) recorded.

To make the system more adaptive, temperature forecasts obtained from a web service help determine the recommended minimum level for the next day: if high temperatures are expected, the system requires a higher water reserve to compensate for increased consumption and evaporation.

All collected data will be recorded, allowing students to analyze how environmental factors influence water demand in wildlife habitats. From a sustainability perspective, the project promotes responsible resource use by identifying low-water conditions before they become critical, reducing unnecessary maintenance trips and preventing waste. This predictive and automated monitoring approach supports ecological management practices and reinforces the importance of protecting wildlife through efficient and sustainable technological solutions.

The Raspberry Pi acts as the main controller and MQTT broker. It reads the water level, obtains weather data, calculates thresholds, and sends IoT alerts. The ESP32 will be placed in the field as a remote node with the connected sensors.

The system working architecture is the following:

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. Measure the tank level.
2. Calculate the fill percentage.
3. Obtain the maximum temperature for tomorrow from an online application.
4. Adjust the minimum recommended threshold.
5. Publish data to the MQTT panel.
6. Send an alert if thresholds are exceeded.
7. Detect animals and record their activity.

4. Keywords:

IOT, Vocational Education, Automation, Sustainability, Enviromental care, Hands-on Learning

5. Target Audience:

Both educators and Students

6. Project Objectives:

- Develop practical knowledge in IOT.
- Understand how to design and implement an smart watering system.
- Gain hands-on experience in programming microcomputers and microcontrollers.
- Understand system-level design: connecting sensors, controlling actuators, and managing data flow.
- Improve teamwork, documentation, and presentation skills.
- Explore sustainability through energy-efficient design choices

7. Project Timeline:

Task	Estimated Hours
Raspberry and ESP32 hardware and programming	8 hours
Practice and validation	4 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.

Task	Estimated Hours
Documentation and presentation preparation	3 hours
Total	**15 hours**

8. Resources Provided:

- Raspberry Pi, ESP32 and sensor kits (AI-SAVES Kit)
- Programming guides
- Needed software and hardware

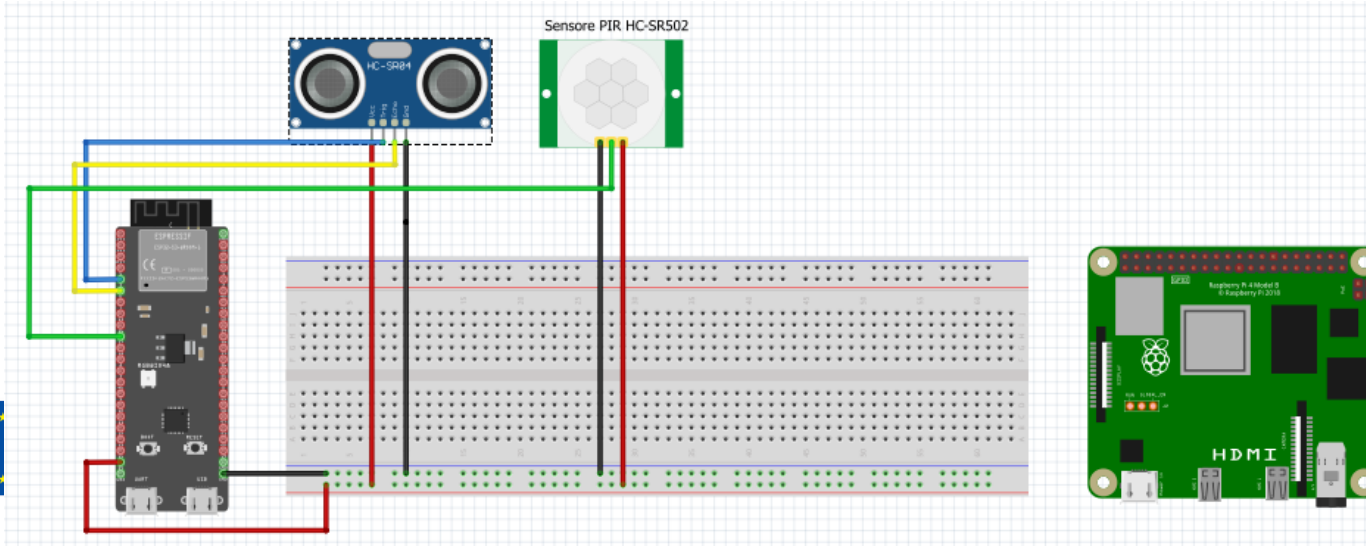
9. Tools and Technologies Used:

- Raspberry PI 4 and ESP32
- Hardware: ultrasonic sensor, PIR sensor,
- Python and C/C++ programming languages, with Geay and IDE programming tools.
- GitHub for version control and project collaboration
- Fritzing for circuit design and testing

10. Step-by-Step Implementation Plan:

1. Set up the Raspberry Pi with the necessary software (Raspbian OS, Python).
2. Conduct practical sessions where students build and program the IOT driven domotic system.

a. Assemble circuit according to the ESP32 pin mapping.



b. Load the needed/provided code in the ESP32 and check locally it works correctly; the object distance measurement and presence detection must be checked.

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.

c. Activate Python virtual environment in the Raspberry Pi:

cd project_folder_path

python -m venv .venv

source .venv/bin/activate

d. Install required libraries:

pip install paho-mqtt requests

e. Run the needed/provided script using Geany

python watering-control.py

f. Load the needed/provided file in the ESP32 using IDE

Watering-wifi-mqtt.ino

g. Be sure Raspberry PI and ESP32 are connected to the same wifi network SSID

h. Use the different inputs/outputs to check the system is running properly and check data are correctly saved in the .csv file. This is the program flow

ESP32

↓ MQTT

Raspberry PI: on_message()

↓

handle_level_message() / handle_pir_message()

↓

append_csv()

↓

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.

writer.writerow() ← data is saved in **CSV file**

3. Evaluate student understanding through project demonstrations.

11. Evaluation Criteria:

- **Hardware Implementation (20%):** Properly connected circuit, no loose wires, correct sensor placement.
- **Code Functionality (50%):** Script runs without errors, correctly detects water level and activates actuators, correctly detects animals (optional), including MQTT funcionality.
- **Understanding & Explanation (15%):** Ability to explain how the system works and its real-world applications.
- **Creativity & Optimization (15%):** Suggestions for improvements, such as adding additional automation features.

12. Languages Available:

English

13. Contributors:

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

14. Feedback and Continuous Improvement:

- **Surveys**: At the end of each milestone, students and educators will complete feedback surveys to assess the effectiveness of the project.
- **Project Reviews**: After project completion, a final review meeting will be held to discuss potential improvements for future AI-based projects.
- **Data Analysis**: Feedback data will be analyzed to identify strengths and areas for improvement, informing future projects.

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.

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.