AI & SUSTAINABILITY IN VET EDUCATION

Nombre de los/las autores/as: Politeknika Txorierri

Título del WP: WP3 – AI-SAVES@ProjectHub

Referencia de la Actividad: Ventilación y Control Climático de Invernadero

Logo:

POLITEKNIKA TXORIERRI

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

TABLA DE CONTENIDOS

1. Título de Proyecto: 5
2. Categoría del Proyecto: 5
3. Descripción del Proyecto: 5
4. Palabras Clave: 5
5. Público Objetivo: 5
6. Objetivos del Proyecto: 6
7. Cronograma del Proyecto: 6
8. Recursos Proporcionados: 6
9. Herramientas y Tecnologías Utilizadas: 7
10. Plan de implementación paso a paso: 7
11. Criterios de Evaluación: 8
12. Idiomas Disponibles: 8
13. Contribuidores/as: 8
14. Retroalimentación y mejora Continua: 8

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

1. **Título de Proyecto:**

Control Climático Inteligente para Invernadero con Raspberry Pi 5

2. **Categoría del Proyecto:**

IoT, Automatización, Sostenibilidad.

3. **Descripción del Proyecto:**

Este proyecto tiene como objetivo diseñar y programar un sistema inteligente de control climático para un invernadero utilizando una Raspberry Pi 5.

El sistema mide la temperatura con un sensor DHT11 y controla automáticamente tres actuadores:

- Un servomotor que abre o cierra una compuerta de ventilación.
- Un ventilador para simular la circulación de aire.
- Tres LEDs (rojo, verde, amarillo) que indican rangos de temperatura.

El alumnado comienza simulando el sistema en Tinkercad utilizando un Arduino UNO y un potenciómetro para representar la temperatura.

Una vez verificada la lógica de control, se pasa a la implementación real con Raspberry Pi 5, donde el programa se escribe en Python utilizando las librerías gpiozero y adafruit-circuitpython-dht.

El objetivo es comprender cómo los sensores y actuadores trabajan conjuntamente para mantener condiciones climáticas óptimas, reforzando conceptos de automatización, programación y sostenibilidad.

4. **Palabras Clave:**

Raspberry Pi 5, Python, DHT11, Servomotor, Control de ventilador, GPIOZero, Automatización, Invernadero, IoT, Sostenibilidad, Regulación climática

5. **Público Objetivo:**

Estudiantes de Formación Profesional y principiantes en programación Python interesados en IoT, automatización y sostenibilidad ambiental.

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

6. Objetivos del Proyecto:

- Simular y probar un sistema de automatización sencillo en Tinkercad.
- Leer datos de temperatura desde un sensor DHT11 y aplicar lógica de decisión.
- Controlar múltiples actuadores (servo, ventilador, LEDs) simultáneamente.
- Desarrollar código Python paso a paso para hardware real.
- Calibrar y depurar un sistema electrónico de control.
- Documentar resultados y comunicar conclusiones de forma clara.

7. Cronograma del Proyecto:

Tarea	Horas estimadas	Descripción
Introducción y planificación	1 hora	Visión general del proyecto y contexto de sostenibilidad
Configuración de simulación en Tinkercad	5 horas	Construir y cablear un circuito virtual con Arduino y probar y ajustar la lógica
Montaje del hardware Raspberry Pi	2 horas	Ensamblar el circuito real (DHT11, servo, ventilador, LEDs)
Programación en Python y pruebas	4 horas	Desarrollar el código principal de control paso a paso
Depuración y calibración	4 horas	Ajustar umbrales y comportamiento de los actuadores
Documentación técnica	2 horas	Redactar un breve informe técnico
Demostración final y reflexión	2 horas	Presentación y análisis de resultados
Total	**20 horas**	

8. Recursos Proporcionados:

- Guía paso a paso de simulación en Tinkercad (basada en Arduino).
- Script simplificado en Python para Raspberry Pi 5 (usando gpiozero).
- Esquema del circuito (Fritzing o manual).
- Plantilla de informe del proyecto y rúbrica de evaluación.
- Fragmentos de código de ejemplo y guía de resolución de problemas.

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

9. Herramientas y Tecnologías Utilizadas:

- Hardware: Raspberry Pi 5, sensor de temperatura DHT11, servomotor SG90, ventilador (controlado mediante relé o transistor), 3 LEDs (rojo/verde/amarillo), resistencias, protoboard, cables jumper.
- Software: Tinkercad Circuits, Python 3, gpiozero, adafruit-circuitpython-dht, Geany IDE.
- Herramientas de diseño: Fritzing, Excel/Calc (para tablas de resultados), procesador de textos para documentación.

10. Plan de implementación paso a paso:

1) **Fase A: Simulación (Arduino en Tinkercad)**
   - **a)** Construir un circuito virtual con Arduino que incluya un potenciómetro, un servo y tres LEDs.
   - **b)** Programar la lógica para simular el comportamiento de la temperatura:
     - i) Valor del potenciómetro → temperatura simulada.
     - ii) LED rojo = calor (>25 °C), verde = óptimo (20–25 °C), amarillo = frío (<20 °C).
     - iii) La posición del servo cambia en consecuencia (0°, 90°, 180°).
   - c) Observar y registrar los resultados en una tabla (valor de entrada ⊠ respuesta LED/servo).

2) **Fase B: Hardware real (Raspberry Pi)**
   - **a)** Cablear los componentes a la Raspberry Pi utilizando la misma lógica probada en Tinkercad.
     - i) DHT11 → GPIO 4
     - ii) Servo → GPIO 12
     - iii) Ventilador → GPIO 17
     - iv) LEDs → GPIO 24 / 25 / 16
   - b) Instalar las librerías Python necesarias:

```
sudo apt update

sudo apt install -y python3-gpiozero python3-libgpiod python3-pip

pip install adafruit-circuitpython-dht
```

   - c) Realizar el cableado: DHT11 al GPIO correspondiente; servo alimentado por fuente externa de 5 V (GND común); módulo relé para control del calefactor/ventilador; LEDs con resistencias.
   - d) Escribir el código Python para leer el sensor y controlar los actuadores.
   - e) Probar el sistema y comprobar que el comportamiento coincide con los resultados de la simulación.

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

3) **Fase C: Depuración, calibración y reflexión**
   a) Ajustar los valores de umbral y los ángulos del servo para lograr un funcionamiento suave.
   b) Identificar causas de inestabilidad (errores del sensor, cableado, temporización).
   c) Documentar el proceso de calibración y las mejoras realizadas.

4) **Fase D: Validación y optimización**
   a) Preparar un informe breve que incluya:
      i) Esquema de cableado y lista de componentes.
      ii) Explicación del código (con comentarios).
      iii) Observaciones y conclusiones.
   b) Presentar el proyecto a la clase, demostrando en directo cómo los cambios de temperatura afectan al sistema.

11. Criterios de Evaluación:

- **Simulación (Tinkercad) (20%):** Lógica correcta y circuito virtual funcional.
- Implementación del hardware (20%): Cableado y montajes adecuados del circuito con Raspberry Pi.
- **Programación (30%):** Lecturas estables del sensor, lógica correcta y control de actuadores.
- **Depuración y calibración (10%):** Evidencias de pruebas, resolución de problemas y ajustes.
- **Documentación y presentación (20%):** Informe claro, explicación coherente y demostración adecuada.

12. Idiomas Disponibles:

- inglés y español

13. Contribuidores/as:

- **Miren Arrien (responsable del proyecto) –** Docente principal responsable del desarrollo y la coordinación del proyecto.
- **Enaitz Elejalde (Experto técnico) –** Aportó experiencia en IoT.
- **Aitor Rey (Experto técnico) –** Aportó experiencia en IoT.
- **Jokin Goioaga (Experto técnico)** – Aportó experiencia en IoT.

14. Retroalimentación y mejora Continua:

El alumnado reflexiona sobre los resultados obtenidos tanto en la simulación como en el hardware real, analizando:

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.

- Diferencias entre el entorno virtual y el entorno real.
- Cómo los ajustes de umbrales afectan a la eficiencia energética y a la estabilidad del sistema.
- Qué mejoras podrían hacer el sistema más fiable (por ejemplo, control basado en humedad, ajuste de velocidad del servo).

Las sesiones de retroalimentación al final de cada fase fomentan la mejora continua, el trabajo en equipo y la conciencia sobre sostenibilidad.

Cofinanciado por la Unión Europea

Este proyecto ha sido financiado con el apoyo de la Comisión Europea. Esta comunicación refleja únicamente las opiniones del autor, y la Comisión no puede ser considerada responsable del uso que pueda hacerse de la información contenida en la misma.