AI & SUSTAINABILITY IN VET EDUCATION

Name of the Author/s: Sucessos Criativos, Lda

Title of WP: WP3 – AI-SAVES@ProjectHub

Activity Reference: 3.2 – Project template for AI-SAVES@ProjectHub

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: 4

5. Target Audience: 4

6. Project Objectives: 4

7. Project Timeline: 5

8. Resources Provided: 5

9. Tools and Technologies Used: 6

10. Step-by-Step Implementation Plan: 6

Step 1: Set Up the Raspberry Pi Environment (from "Getting Started with Raspberry Pi 5") 6

Step 2: Updating GPIO Libraries for Raspberry Pi 5 7

Step 3: Connect the LED Circuit 8

Step 4: Write the Python Code to Blink the LED 8

11. Evaluation Criteria: 11

12. Languages Available: 11

13. Contributors: 12

17. Feedback and Continuous Improvement: 12

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:

First Steps with AI-VET Kit: Blinking an LED with Raspberry Pi 5

---

2. Project Category:

Microelectronics Fundamentals

---

3. Project Description:

This project continues the setup of the Raspberry Pi 5 by extending its configuration into physical interaction. Participants will prepare their Raspberry Pi for development, install the necessary software, and run a simple Python script to blink an LED. This marks the first tangible outcome of their work — turning digital code into real-world action — and sets the stage for future projects involving sensors and actuators.

---

4. Keywords:

Raspberry Pi, Python, Virtual Environment, GPIO, LED, Geany

---

5. Target Audience:

Educators, learners, and makers with or without prior experience who are beginning their journey into microelectronics and physical computing.

---

6. Project Objectives:

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.

- Configure a Python virtual environment with GPIO support for Raspberry Pi 5.
- Use Geany to write and execute Python code.
- Create a simple physical circuit to blink an LED using GPIO.

---

7. Project Timeline:

The estimated duration is approximately **2.5 to 3 hours**, including environment setup and physical assembly.

---

8. Resources Provided:

- AI-VET Kit (Raspberry Pi 5, LED, Breadboard, Jumpers, Resistor)

MATERIAL	
Raspberry Pi 5	
27W USB Type-C Power Supply for Raspberry Pi 5	
Flat Jumper Wires F/F	

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.

Breadboard	
SINGLE-FILE WIRE FOR BREADBOARDS	
Resistor 330 Ohm 1/4 W	
Diffused 5mm LED Yellow	

---

9. Tools and Technologies Used:

- Raspberry Pi OS
- Python 3 (in virtual environment)
- `lgpio` library
- Geany IDE
- Terminal (bash)

---

10. Step-by-Step Implementation Plan:

Step 1: Set Up the Raspberry Pi Environment (from "Getting Started with Raspberry Pi 5")

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.

This project builds directly on the tasks completed in “Getting Started with Raspberry Pi 5: Setup, Configuration, and Hello World in Python.” If you're beginning here, or need a refresher, you’ll go through the essential setup steps to prepare your Raspberry Pi for hardware interaction.

In this step, you will:

- Install Raspberry Pi OS using the Raspberry Pi Imager tool.
- Configure system settings, such as Wi-Fi, keyboard layout, and local time zone.
- Enable critical interfaces like SPI, I2C, 1-Wire, and VNC through the Raspberry Pi Configuration menu.
- Connect to the Pi remotely using TigerVNC Viewer by identifying its IP address via ifconfig.
- Perform a full system update

Step 2: Updating GPIO Libraries for Raspberry Pi 5

On the Raspberry Pi 5, there were changes in how GPIO (General Purpose Input/Output) pins are accessed and managed on the newer hardware. The Raspberry Pi 5 introduces different hardware and firmware, and the previous RPi.GPIO library, which was widely used on earlier Raspberry Pi models for controlling the GPIO pins, does not support the new architecture of the Raspberry Pi 5.

Therefore, it needs to be uninstalled to avoid conflicts or incompatibility issues. The lgpio library is the recommended replacement for GPIO access on the Raspberry Pi 5. It supports the new hardware and offers the necessary functionality to interact with the GPIO pins on this version of the Raspberry Pi.

To do this, execute the following commands on the terminal window:

```
source env/bin/activate

sudo apt install python3-rpi-lgpio
```

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.

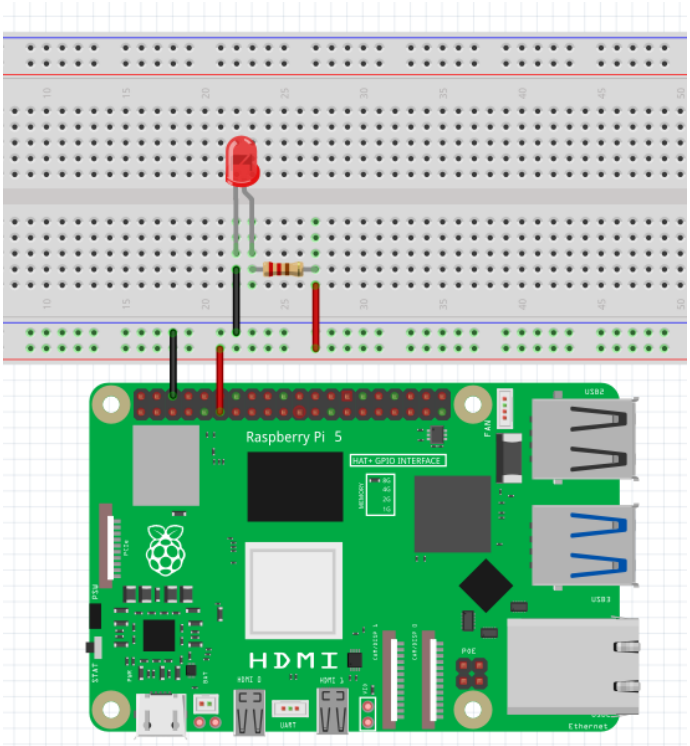
MAKE SURE YOU REBOOT AFTER THESE COMMANDS!!!

Now you are ready to start!

Step 3: Connect the LED Circuit

Wiring:

- Connect **GPIO17** (pin 11) to the long leg (anode) of the LED via a resistor.
- **Connect** the short leg (cathode) to a GND pin (e.g., pin 6).



Step 4: Write the Python Code to Blink the LED

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.

After make the circuit you are ready to start coding!



Make sure that:
Already complete all the steps on 4.1, 4.3 and 5.1.
The virtual environment **env** *is activated*

Open Geany and write the following code. Check the comments to fully understand each line:

```
import lgpio
import time
# Define the GPIO pin number (e.g., GPIO 17)
```

Now, you can run the program and see the result:

```
try:
    # Infinite loop to blink the LED
    while True:
        lgpio.gpio_write(h, LED_PIN, 1)  # Turn on LED
        time.sleep(1)                    # Wait for 1 second
        lgpio.gpio_write(h, LED_PIN, 0)  # Turn off LED
        time.sleep(1)                    # Wait for 1 second
except KeyboardInterrupt:
    # Catch the exception if the user interrupts the program (Ctrl + C)
    print("Stopping LED blinking...")
    time.sleep(1)
    lgpio.gpio_write(h, LED_PIN, 0)      # Turn off the LED before exit
    lgpio.gpiochip_close(h)
```

Code Breakdown:

- `import lgpio` and `import time`:
  These modules allow the script to interact with the GPIO hardware and manage delays.

- `LED_PIN = 17`:
  This defines the GPIO pin number to which the LED is connected (GPIO17, physical pin 11).

- `h = lgpio.gpiochip_open(0)`:
  Opens the first GPIO chip (`/dev/gpiochip0`) to start communication with the GPIO pins.

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.

- `lgpio.gpio_claim_output(h, LED_PIN)`:
  Sets GPIO17 as an **output pin** so it can control the LED.

- `while True:` loop:
  This is an infinite loop that repeatedly:
  - Turns the LED **on** (`lgpio.gpio_write(..., 1)`)
  - Waits **1 second**
  - Turns the LED **off** (`lgpio.gpio_write(..., 0)`)
  - Waits **another second**

- `except KeyboardInterrupt:`
  If the user presses `Ctrl+C` to stop the script, this block:
  - Turns the LED off
  - Closes the GPIO connection
  - Prints a message to confirm safe shutdown

Save *the program as* **blink.py**

Now, you can run the program and see the result:

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.

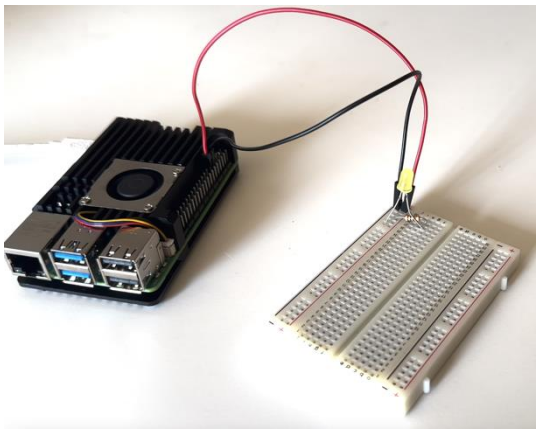


Figure 1- Led is ON.

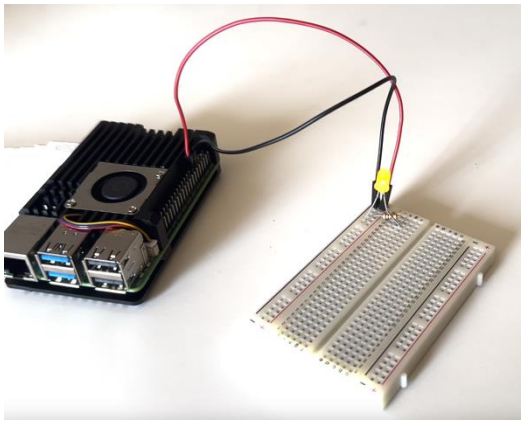


Figure 2- Led is OFF.

Now, you can run the program and see the result:

11. Evaluation Criteria:

- **Participant Feedback:** Gather reflections on the clarity, usability, and learning value of the project.
- **Project Completion:** Verify that the setup and LED blinking task are completed successfully.
- **Technical Accuracy:** Assess whether the circuit and code function as intended.
- **Confidence & Readiness:** Determine if participants feel prepared to continue with more advanced projects.

---

12. Languages Available:

English, **Portuguese** (translations in progress)

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.

13. Contributors:

Fátima Pais, Luís Soares – Project Designers and Facilitators

---

17. Feedback and Continuous Improvement:

- **Surveys**: At the end of this 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.

Fátima Pais, Luís Soares – Project Designers and Facilitators

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.