AI & SUSTAINABILITY IN VET EDUCATION

Nome do(s) autor(es): Sucessos Criativos, Lda

Título do WP: WP3 – IA-SAVES@ProjectHub

Referência de Atividade: 3.2 – Modelo de projeto para IA-SAVES@ProjectHub

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

Índice

1. Título do Projeto: 4
2. Categoria do Projeto: 4
3. Descrição do Projeto: 4
4. Palavras-chave: 4
5. Público-alvo: 4
6. Objetivos do Projeto: 4
7. Cronologia do Projeto: 5
8. Recursos Fornecidos: 5
9. Ferramentas e Tecnologias Utilizadas: 6
10. Plano de Implementação Passo a Passo: 7
Passo 1: Prepare o Raspberry Pi para Interação de Hardware 7
Passo 2: Ligue o Sensor de LDR 7
Passo 3: Escreve o programa Python para Detetar Luz 8
11. Critérios de Avaliação: 12
12. Línguas Disponíveis: 12
13. Colaboradores: 12
17. Feedback e Melhoria Contínua: 13

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

1. Título do Projeto:

Primeiros Passos com o Kit AI-VET: Controlar a NeoPixel Strip

---

2. Categoria do Projeto:

Fundamentos da Microeletrónica

---

3. Descrição do Projeto:

Este projeto demonstra como controlar uma fita LED WS2812 NeoPixel usando um Raspberry Pi 5. Os alunos irão explorar como criar efeitos visuais de iluminação usando comunicação SPI e código Python. O projeto introduz controlo programável de LEDs, manipulação de cores RGB e padrões de luz animados — perfeitos para aplicações interativas ou decorativas.

---

4. Palavras-chave:

NeoPixel, WS2812, Raspberry Pi, LEDs RGB, SPI, Python, Animação de Luz

---

5. Público-alvo:

Educadores, aprendizes e criadores, com ou sem experiência prévia, que estão a iniciar o seu percurso na microeletrónica e computação física.

---

6. Objetivos do Projeto:

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

- Ligue uma fita LED WS2812 NeoPixel ao Raspberry Pi usando SPI
- Instale e use a `biblioteca Python` adafruit-circuitpython-neopixel-spi
- Gerar efeitos de iluminação dinâmica usando loops e cores aleatórias
- Compreenda como funciona o SPI no contexto do controlo RGB de LEDs

---

7. Cronologia do Projeto:

A duração estimada é de aproximadamente **2,5 a 3 horas**, incluindo a montagem do ambiente e a montagem física.

---

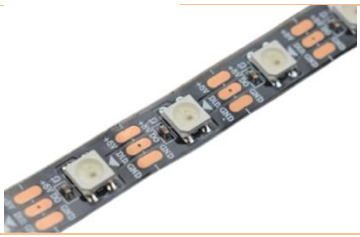
8. Recursos Fornecidos:

- Kit AI-VET (Raspberry Pi 5, LED, Breadboard, Jumpers, Fita LED RGB WS2812)

MATERIAL	
Raspberry Pi 5	

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

Fonte de alimentação USB Tipo-C de 27W para Raspberry Pi 5	
Cabos Flat Jumper F/F	
Breadboard	
FIO EM FILA ÚNICA PARA PLACAS DE ENSAIO	
WS2812 Fita LED Programável RGB 15x LED	

---

9. Ferramentas e Tecnologias Utilizadas:

- Raspberry Pi OS
- Python 3
- `adafruit-circuitpython-neopixel-spi` library
- Geany IDE
- Terminal (bash)

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

10. Plano de Implementação Passo a Passo:

Passo 1: Prepare o Raspberry Pi para Interação de Hardware

Este projeto baseia-se diretamente nas tarefas concluídas em **"Começar com Raspberry Pi 5: Configuração, Configuração e Hello World em Python."** Se estás a começar por aqui, ou precisas de uma atualização, começa por preparar o teu Raspberry Pi para comunicação a nível de hardware usando Python.

Neste passo, irá:

- **Instale o Raspberry Pi OS** usando a ferramenta Raspberry Pi Imager
- **Configure definições do sistema** como Wi-Fi, layout do teclado e fuso horário
- **Ative as interfaces principais** através do menu de Configuração do Raspberry Pi:
    - **SPI** (necessário para controlo NeoPixel)
    - **I²C**, 1-Wire e VNC (opcionais, mas úteis para outros projetos)
- Use o comando `ifconfig` no terminal para **identificar o endereço IP do seu Raspberry Pi** e ligue-se remotamente via **TigerVNC Viewer**, se necessário

Instalar as bibliotecas Python Necessárias

Agora irá instalar as bibliotecas usadas para interagir com as faixas NeoPixel baseadas em SPI e, opcionalmente, com ecrãs OLED:

- **Para controlar LEDs NeoPixel**, instale o driver compatível com Adafruit SPI:

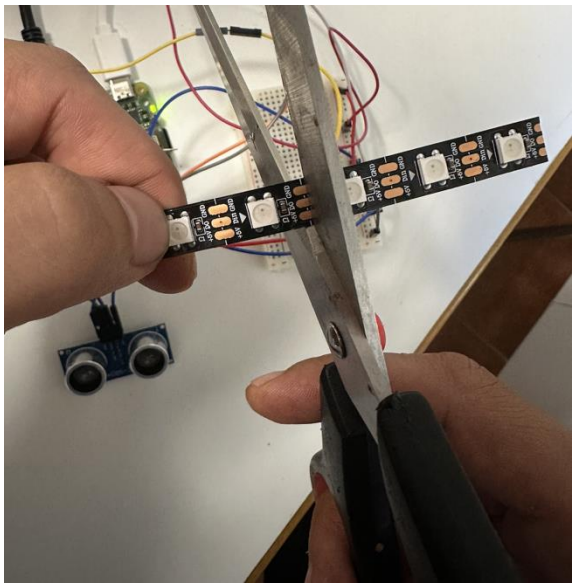
```
pip3 instale adafruit-circuitpython-neopixel-spi
```

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

Passo 2: Prepare a Tira

É possível personalizar o comprimento cortando para ter exatamente o número de LEDs de que precisas. É importante cortar a tira cuidadosamente nas linhas de corte designadas, que estão marcadas entre cada LED: Estas linhas indicam pontos de corte seguros, garantindo que não danifica a cablagem interna:



Depois de cortar, terá de soldar os conectores necessários às pastilhas expostas (5V, GND e Data In/Out). Certifique-se de corresponder à orientação dos conectores mostrada na imagem, garantindo a ligação correta e um funcionamento fiável:

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.



O resultado final é mostrado na imagem seguinte:



Passo 3: Construir o Circuito

Neste passo, vai ligar a sua fita LED NeoPixel ao Raspberry Pi usando SPI. A fita WS2812 requer uma temporização precisa para comunicação, tornando o SPI uma interface excelente.

Ligações NeoPixel Strip:

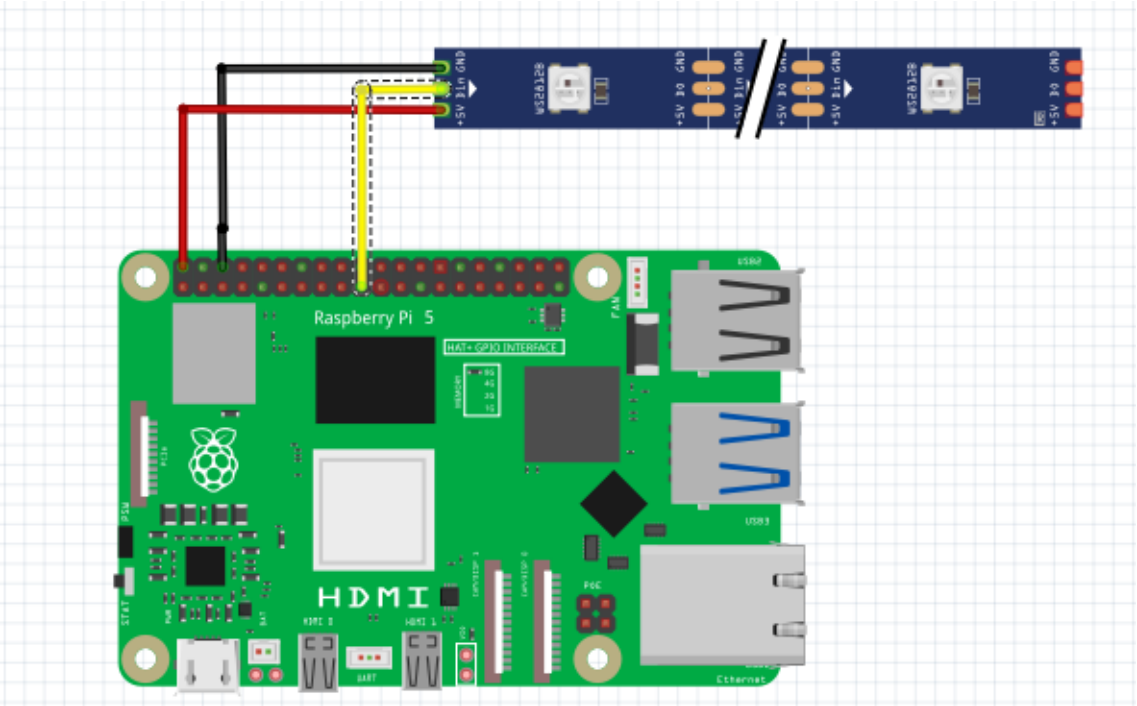
Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

- **5V → Raspberry Pi 5V (pino 2)**
  Alimenta a fita LED

- **GNDRaspberry Pi GND (pino 6) →**
  Estabelece um terreno comum

- **Dados InGPIO 10 (pino 19) →**
  O pino SPI MOSI (Master Out, Slave In) envia dados RGB para a faixa LED



Passo 4: Escreva o programa Python para controlar os LEDs

Com a cablagem concluída, é altura de escrever a sua primeira animação de iluminação usando SPI.

Abra o Geany e introduza o seguinte código:

```
Tabuleiro de importação
```

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

```
importar neopixel_spi como Neopixel
Tempo de importação
importar aleatório # Importar aleatório para gerar cores aleatórias
# Defina o número de LEDs na faixa NeoPixel
NUM_PIXELS = 6
# Defina a ordem das cores para o NeoPixel (Verde, Vermelho, Azul)
PIXEL_ORDER = neopixel. GRB
# Defina um atraso entre mudanças de cor
ATRASO = 0,1
# Inicializar a comunicação SPI para controlar LEDs NeoPixel
SPI = Conselho. SPI()
# Criar um objeto NeoPixel para controlar os LEDs via SPI
píxeis = neopíxel. NeoPixel_SPI(spi,
                                NUM_PIXELS,
                                pixel_order=PIXEL_ORDER,
                                auto_write=Falso)  # Definir auto_write para
False para atualizações manuais
# Laço principal para mudar as cores dos LEDs
embora Verdadeiro:
    # Gerar uma cor aleatória para cada iteração
    cor = (random.randint(0, 255), random.randint(0, 255), random.randint(0,
255))  # Cor RGB aleatória

    # Iterar por cada pixel e defini-lo para a cor gerada aleatoriamente
    para i no intervalo(NUM_PIXELS):
        pixels[i] = cor # Atribuir a cor aleatória ao pixel atual
        pixels.show() # Atualizar os LEDs para mostrar a nova cor
        time.sleep(DELAY) # Atraso entre cada mudança de LED

        # Limpe os LEDs definindo-os para 0 (desligado) após o atraso
        pixels.fill(0)
```

Guarde o ficheiro como **`neopixel_random.py`**

Decomposição do Código:

Configuração e Importações

Importa a biblioteca `neopixel_spi e os módulos` aleatórios `e` temporais `do Python` para lógica de iluminação e animação.

Configuração

- `NUM_PIXELS` define quantos LEDs estão ativos
- A ordem dos pixels está definida para GRB (comum nas tiras WS2812)

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

Controlo de Píxeis

- É gerada uma cor RGB aleatória
- Cada pixel é definido para essa cor com um pequeno atraso
- Após a exibição da animação, a tira é limpa

Passo 5: Execute o programa e observe a saída

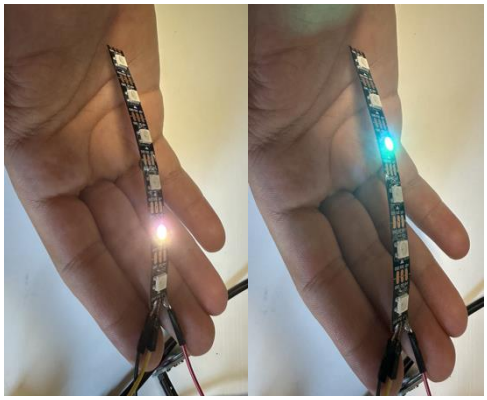
Vamos acender as coisas!

Para **executar** o script:

1. **Guarda** o programa como `neopixel_random.py`
2. No **terminal**, escreva:

```
Python3 neopixel_random.py
```

Deves ver a tua fita a mostrar cores aleatórias, um LED de cada vez, num loop.



Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

11. Critérios de Avaliação:

- **Configuração de Hardware**: A fita NeoPixel está devidamente soldada e ligada por cabo
- **Comunicação SPI**: Raspberry Pi SPI está ativado e a tira recebe dados
- **Saída Visual**: A fita LED mostra alterações dinâmicas de cor conforme programado
- **Execução de Código**: O script Python corre sem erros e atualiza os LEDs em tempo real
- **Exploração Criativa**: Código modificado ou estendido pelo aprendiz para efeitos de iluminação personalizados

12. Línguas Disponíveis:

Inglês, **Português** (traduções em progresso)

13. Colaboradores:

Fátima Pais, Luís Soares – Designers e Facilitadores de Projetos

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.

17. Feedback e Melhoria Contínua:

- **Inquéritos**: No final deste marco, alunos e educadores irão preencher inquéritos de feedback para avaliar a eficácia do projeto.
- **Revisões de Projetos**: Após a conclusão do projeto, será realizada uma reunião final de revisão para discutir potenciais melhorias para futuros projetos baseados em IA.
- **Análise de Dados**: Os dados de feedback serão analisados para identificar pontos fortes e áreas a melhorar, informando projetos futuros.

Co-funded by the European Union

Este projeto foi financiado com o apoio da Comissão Europeia. Esta comunicação reflete apenas as opiniões do autor, e a Comissão não pode ser responsabilizada por qualquer uso que possa ser feito da informação nela contida.