AI & SUSTAINABILITY
IN VET EDUCATION

AI-VET MOOC - шаблон за поднесување проект

Funded by
the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

1. Наслов на проектот:

„Раскрсница“

2. Категорија на проектот:

☐ **Интелигентни системи за автоматизација и контрола**

3. Опис на проектот:

Овој проект претставува систем за управување со сообраќај на раскрсница кој ќе може автоматски да го регулира движењето на возилата и пешаците. На раскрсницата се ставени четири семафори за регулација на движење на автомобилите и четири семафори за регулација на движење на пешаците. Поврзан е и 7-сегментен дисплеј со заедничка катода кој го одбројува преостанатото време на светење на црвената и зелената диода. Се користат ЛЕД диоди во три бои кои наизменично светат црвено-жолто-зелено-жолто-црвено и управувани програмски се вклучуваат-исклучуваат по следниов редослед: зелена 10 секунди, жолта 2 секунди, зрвена 16 секунди, жолта 2 секунди... Семафорите за пешаците работат по следниов редослед: црвена 16 секунди и зелена 10 секунди.

4. Клучни зборови:

семафор, раскрсница, сообраќај, регулација, 7-сегментен дисплеј

5. Целна публика:

Студенти

6. Цели на проектот:

Целта на овој проект е да се изработи паметен систем за управување со сообраќај на раскрсница кој автоматски ќе го регулира движењето на возилата и пешаците. Во текот на изработката на проектот се стекнуваат основни знаења за работа со Raspberry Pi, програмирање во Python и користење на различни библиотеки. Исто така, се учи како се поврзуваат и комбинираат различни електронски компоненти, како и начинот на нивно функционирање во еден заеднички систем. Со оваа вежба се учи:

1. Да се инсталира оперативен систем за Raspberry Pi
2. Хардверско поврзување на елементите
3. Разбирање на основна интеграцијата на хардвер и софтвер (Python, библиотеки).
4. Да се демонстрира интеракција хардвер-софтвер со користење на Python

7. Временска рамка на проектот:

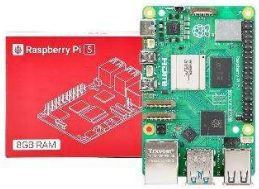
Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

- Датум на почеток: 15.3.2026
- Датум на завршување: 1.4.2026

8. Користени алатки и технологии:

1. Компоненти на комплетот AI-VET:

	Компонента	picture	Функција
1.	Raspberry Pi 5 – 8GB		Се користи како централна управувачка единица („мозок“) на системот. По добивањето на резултатите од камерата, ги обработува и ги пренесува до дисплејот за прикажување.
2.	Raspberry Метално куќиште со вентилатор		Служи за заштита и ладење на уредот. Металното куќиште го штити Raspberry Pi од прашина, механички оштетувања и случајни допири, додека вентилаторот помага во намалување на температурата при работа.
3.	LED диоди – црвена, зелена, жолта		Со помош на диодите ги изработуваме семафорите за автомобилите и пешаците
4.	7-сегментен дисплеј со заедничка катода		Го покажува преостанатото време на светење на црвените и зелените диоди
5.	Отпорници – 330 Ω		Улогата на отпорниците е да ги заштитат диодите од прегорување

Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

6.	Протоборд и жици за спојување	 	**Жиците за спојување (jumper wires)** служат за меѓусебно поврзување на компонентите на протобордот **(breadboard)** и со микроконтролерот (Raspberry Pi).
7.	micro HDMI to HDMI кабел 1m		Се користи за поврзување на системот со дисплеј, пренесувајќи видео и аудио сигнали.
8.	27W USB Напојување од тип C за Raspberry Pi 5		Кабел за напојување
9.	Micro SDHC картица 64GB		Се користат за сериско поврзување и напојување на сите наведени компоненти, овозможувајќи нивна меѓусебна комуникација и синхронизирана работа.

2. AI-VET-Прирачник за склопување со упатства чекор-по-чекор
3. Python код и библиотеки
4. Datasheet за 7-сегментен дисплеј

9. План за имплементација чекор-по-чекор:

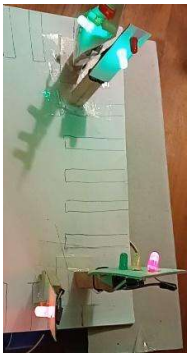
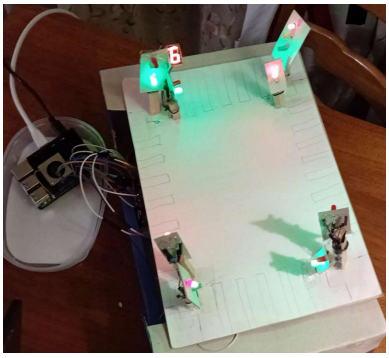
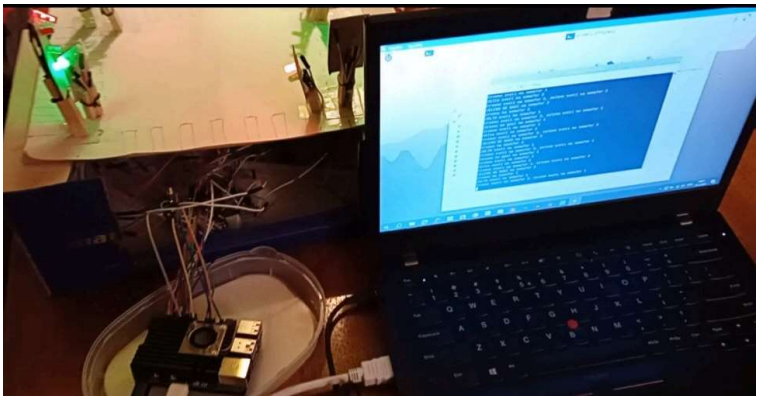
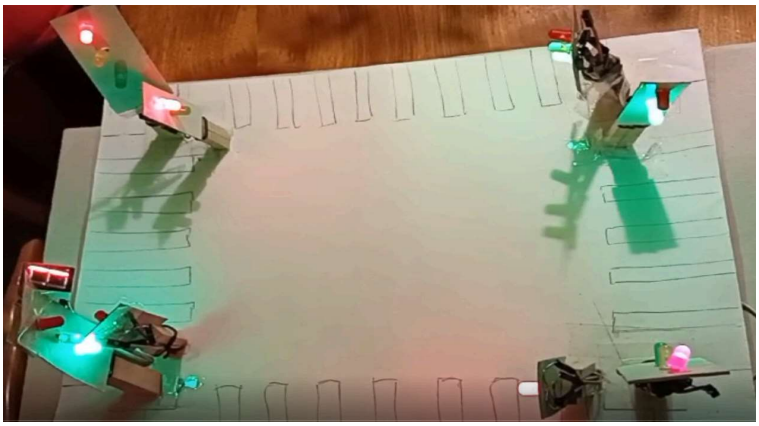
1. **План-идеа што сакаме да изработиме:** Современа раскрсница која ќе регулира сообраќај. Почнувме со работа на еден семафор и потоа постепено ги додаваме прво сите семафори за регулација на движење на автомобилите а потоа и останатите четири семафори за регулација на движење на пешаците. На крај додаваме 7-сегментен дисплеј кој ќе го покажува преостанатото време од светење на црвената и зелената диода.
2. **Подготовка на хардверските компоненти:** Соберете ги сите потребни материјали - Raspberry Pi 5, диоди, отпорници, 7-сегментен дисплеј, жици, протоборд, кабли за напојување, и потоа следи поврзување.

Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

3. **Склопување на Raspberry Pi 5 и кутија за Pi 5 со вентилатор со контролирана температура.**
4. **Инсталирање на Raspbian OS на SD картичката и конфигурирање пристап до мрежата преку VNC или SSH и ажурирајте ги системските пакети.**
5. **Интегрирање на хардверските компоненти:** поврзување на диодите, отпорниците и 7-сегментниот дисплеј на протоборд и поврзување со жици со Raspberry Pi 5
6. **Програмирање со Python и користење на библиотеки**
7. **Тестирање на работата на проектот**



Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

10- Код

```
from gpiozero import LED
import lgpio
import time

RED1 = 4
YELLOW1 = 3
GREEN1 = 23
YELLOW2 = 15
GREEN2 = 2
RED2 = 18
PR1 = 24
PG1= 25
PR2= 17
PG2= 14
D1=27
D2=10
D3=22
D4=9
D5=11
D6=5
D7=6
D8=13

segments = {
    'a': LED(22),
    'b': LED(27),
    'c': LED(5),
    'd': LED(6),
    'e': LED(13),
    'f': LED(10),
    'g': LED(9)
}

numbers = {
    0:['a','b','c','d','e','f'],
    1:['b','c'],
    2:['a','b','g','e','d'],
    3:['a','b','g','c','d'],
    4:['f','g','b','c'],
    5:['a','f','g','c','d'],
    6:['a','f','g','c','d','e'],
    7:['a','b','c'],
    8:['a','b','c','d','e','f','g'],
    9:['a','b','c','d','f','g']
}
```

Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

```
h = lgpio.gpiochip_open(0)
lgpio.gpio_claim_output(h, RED1)
lgpio.gpio_claim_output(h, YELLOW1)
lgpio.gpio_claim_output(h, GREEN1)
lgpio.gpio_claim_output(h, RED2)
lgpio.gpio_claim_output(h, GREEN2)
lgpio.gpio_claim_output(h, YELLOW2)
lgpio.gpio_claim_output(h, PR1)
lgpio.gpio_claim_output(h, PG1)
lgpio.gpio_claim_output(h, PR2)
lgpio.gpio_claim_output(h, PG2)

def display_number(num):
    for seg in segments.values():
        seg.off()

    for seg in numbers[num]:
        segments[seg].on()

try:
    while True:
        print("Crveno sveti na semafor 1, zeleno sveti na semafor 2")
        lgpio.gpio_write(h, RED2, 0)
        lgpio.gpio_write(h, GREEN1, 0)
        lgpio.gpio_write(h, YELLOW1, 0)
        lgpio.gpio_write(h, YELLOW2, 0)
        lgpio.gpio_write(h, RED1, 1)
        lgpio.gpio_write(h, GREEN2, 1)
        lgpio.gpio_write(h, PG1, 0)
        lgpio.gpio_write(h, PR1, 1)
        lgpio.gpio_write(h, PG2, 1)
        lgpio.gpio_write(h, PR2, 0)

        for i in range(9, -1, -1):
                display_number(i)
                time.sleep(1)

        print("ZELENO SE GASI na semafor 2")
        lgpio.gpio_write(h, RED1, 1)
        lgpio.gpio_write(h, GREEN1, 0)
        lgpio.gpio_write(h, YELLOW2, 1)
        lgpio.gpio_write(h, YELLOW1, 0)
        lgpio.gpio_write(h, GREEN2, 0)
        lgpio.gpio_write(h, PG1, 0)
        lgpio.gpio_write(h, PR1, 1)
        lgpio.gpio_write(h, PG2, 0)
        lgpio.gpio_write(h, PR2, 1)
```

Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

```
for seg in segments.values():
    seg.off()

time.sleep(2)

print("Crveno na semafor 2")
lgpio.gpio_write(h, RED2, 1)
lgpio.gpio_write(h, GREEN1, 0)
lgpio.gpio_write(h, YELLOW1, 0)
lgpio.gpio_write(h, YELLOW2, 0)
lgpio.gpio_write(h, RED1, 1)
lgpio.gpio_write(h, GREEN2, 0)
lgpio.gpio_write(h, PG1, 0)
lgpio.gpio_write(h, PR1, 1)
lgpio.gpio_write(h, PG2, 0)
lgpio.gpio_write(h, PR2, 1)
time.sleep(2)

print("ZOLTO sveti na semafor 1")
lgpio.gpio_write(h, RED2, 1)
lgpio.gpio_write(h, GREEN1, 0)
lgpio.gpio_write(h, YELLOW1, 1)
lgpio.gpio_write(h, YELLOW2, 0)
lgpio.gpio_write(h, RED1, 1)
lgpio.gpio_write(h, GREEN2, 0)
lgpio.gpio_write(h, PG1, 0)
lgpio.gpio_write(h, PR1, 1)
lgpio.gpio_write(h, PG2, 0)
lgpio.gpio_write(h, PR2, 1)
time.sleep(2)

print("Crveno sveti na semafor 2, zeleno sveti na semafor 1")
lgpio.gpio_write(h, RED2, 1)
lgpio.gpio_write(h, GREEN1, 1)
lgpio.gpio_write(h, YELLOW1, 0)
lgpio.gpio_write(h, YELLOW2, 0)
lgpio.gpio_write(h, RED1, 0)
lgpio.gpio_write(h, GREEN2, 0)
lgpio.gpio_write(h, PG1, 1)
lgpio.gpio_write(h, PR1, 0)
lgpio.gpio_write(h, PG2, 0)
lgpio.gpio_write(h, PR2, 1)

for i in range(9, -1, -1):
    display_number(i)
    time.sleep(1)
```

Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

```
        print("Zeleno se gasi na semafor 1")
        lgpio.gpio_write(h, RED2, 1)
        lgpio.gpio_write(h, GREEN1, 0)
        lgpio.gpio_write(h, YELLOW1, 1)
        lgpio.gpio_write(h, YELLOW2, 0)
        lgpio.gpio_write(h, RED1, 0)
        lgpio.gpio_write(h, GREEN2, 0)
        lgpio.gpio_write(h, PG1, 0)
        lgpio.gpio_write(h, PR1, 1)
        lgpio.gpio_write(h, PG2, 0)
        lgpio.gpio_write(h, PR2, 1)

        for seg in segments.values():
            seg.off()

        time.sleep(2)

        print("Crveno sveti na semafor 1")
        lgpio.gpio_write(h, RED2, 1)
        lgpio.gpio_write(h, GREEN1, 0)
        lgpio.gpio_write(h, YELLOW1, 0)
        lgpio.gpio_write(h, YELLOW2, 0)
        lgpio.gpio_write(h, RED1, 1)
        lgpio.gpio_write(h, GREEN2, 0)
        lgpio.gpio_write(h, PG1, 0)
        lgpio.gpio_write(h, PR1, 1)
        lgpio.gpio_write(h, PG2, 0)
        lgpio.gpio_write(h, PR2, 1)
        time.sleep(2)

        print("Zolto sveti na semafor 2")
        lgpio.gpio_write(h, RED2, 1)
        lgpio.gpio_write(h, GREEN1, 0)
        lgpio.gpio_write(h, YELLOW1, 0)
        lgpio.gpio_write(h, YELLOW2, 1)
        lgpio.gpio_write(h, RED1, 1)
        lgpio.gpio_write(h, GREEN2, 0)
        lgpio.gpio_write(h, PG1, 0)
        lgpio.gpio_write(h, PR1, 1)
        lgpio.gpio_write(h, PG2, 0)
        lgpio.gpio_write(h, PR2, 1)
        time.sleep(2)

except KeyboardInterrupt:
    print("Stopping LED blinking...")
    lgpio.gpio_write(h, RED2, 0)
    lgpio.gpio_write(h, GREEN1, 0)
    lgpio.gpio_write(h, YELLOW1, 0)
```

Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.

```
    lgpio.gpio_write(h, RED1, 0)
    lgpio.gpio_write(h, GREEN2, 0)
    lgpio.gpio_write(h, YELLOW2, 0)
    lgpio.gpio_write(h, PG1, 0)
    lgpio.gpio_write(h, PR1, 0)
    lgpio.gpio_write(h, PG2, 0)
    lgpio.gpio_write(h, PR2, 0)
    lgpio.gpiochip_close(h)
    for seg in segments.values():
        seg.off()
```

11. Критериуми за евалуација:

Функционалност на системот (70%): Дали проектот работи и дали работат сите барања (целите што сме ги поставиле).
Идејно решение за поставеност на елементите (10%): едноставноста и практичноста за користење
Естетика (10%): изгледот на крајниот продукт
Временска рамка (10%): За кои временски рок е изработен.

12. Кредити

Ана Николоска, III-4 - Работеше самостојно од идеа, истражување, склопување, инсталирање, хардверско решение, поврзување и софтверско програмирање.

Професор Гордана Наумовска - Обезбеди насоки и поддршка во текот на работењето на проектот.

13. Линк до видео презентација (макс. 5 минути)

https://drive.google.com/file/d/13fSXODzDGP7Hw23YBkBZC-H5RMIYOgQc/view?usp=drive_link

14- Изјави
Изјавувам дека ги прочитав и се согласувам со следните изјави:

1- Проектот е оригинален и развиен од наведените членови на тимот.
2- Сите технологии и извори се правилно цитирани.
3- Проектот е развиен со користење на компоненти од комплетот AI-VET.
4- Ние го овластуваме ширењето на проектот за образовни и промотивни цели (неексклузивни права).

Funded by the European Union

Овој проект е финансиран со поддршка од Европската комисија. Оваа комуникација ги одразува само ставовите на авторот, а Комисијата не може да се смета за одговорна за каква било употреба што може да се направи на информациите содржани во неа.