```python
from machine import Pin, PWM
import network
import socket
import uasyncio as asyncio

# Configure WiFi credentials
ssid = 'MySSID'
password = 'MyPassword'

# Setup the servo pin
servo_pin = Pin(4)
pwm = PWM(servo_pin, freq=50)


def angle_to_duty(angle):
    min_duty = 25   # Minimum duty cycle for 0°
    max_duty = 125  # Maximum duty cycle for 180°
    return int((angle / 180) * (max_duty - min_duty) + min_duty)

# Function to set the servo angle
def set_servo_angle(angle):
    duty_cycle = angle_to_duty(angle)
    pwm.duty(duty_cycle)

# Servo control variables
swinging = False

async def servo_swing():
    global swinging
    while True:
        if swinging:
            set_servo_angle(45)   # Swing to 45°
            await asyncio.sleep_ms(500)
            set_servo_angle(0)    # Return to 0°
            await asyncio.sleep_ms(500)
        else:
            await asyncio.sleep_ms(100)

async def handle_web_requests():
    addr = socket.getaddrinfo('0.0.0.0', 80)[0][-1]
    s = socket.socket()
    s.bind(addr)
    s.listen(1)
    s.settimeout(0.1)

    while True:
        try:
            cl, addr = s.accept()
            request = cl.recv(1024).decode()

            global swinging
            if '/start' in request:
                swinging = True
            elif '/stop' in request:
                swinging = False
                set_servo_angle(0)

            html = """<!DOCTYPE html>
            <html>
            <head> <title>Servo Control</title> </head>
```

```python
            <body>
            <h1>Control Your Servo Motor</h1>
            <form action="/start" method="get">
                <button type="submit">Start Swinging</button>
            </form>
            <form action="/stop" method="get">
                <button type="submit">Stop Swinging</button>
            </form>
            </body>
            </html>
            """

            cl.send('HTTP/1.0 200 OK\n\n')
            cl.send(html)
            cl.close()
        except OSError:
            pass
        await asyncio.sleep_ms(10)

async def main():
    # Setup WiFi
    wlan = network.WLAN(network.STA_IF)
    wlan.active(True)
    wlan.connect(ssid, password)

    while not wlan.isconnected():
        await asyncio.sleep_ms(100)

    print('Network Config:', wlan.ifconfig())

    # Create tasks
    task1 = asyncio.create_task(servo_swing())
    task2 = asyncio.create_task(handle_web_requests())

    # Run tasks
    await asyncio.gather(task1, task2)

# Run the async main function
try:
    asyncio.run(main())
except KeyboardInterrupt:
    print("Stopping the servo")
finally:
    pwm.duty(0)  # Set duty cycle to 0
    pwm.deinit()  # Stop PWM signal
    print("PWM deinitialized")
```