Hier findest du meine Raspberry-Pi-Kurse als praxisnahe Schritt-für-Schritt-Projekte.
Jeder Kurs ist eigenständig aufgebaut, verständlich erklärt und so konzipiert, dass du das Gelernte direkt zu Hause oder im Alltag einsetzen kannst.

Free (Mini) Kurs: n8n auf Raspberry Pi im LAN (Docker & Compose)

Automationen lokal betreiben – ohne Cloud, ohne Abo
Zielgruppe: Technik-Einsteiger, Selbsthoster & Maker

Kurzüberblick	Details
Dauer	ca. **60 Minuten**
Betrieb	**LAN-only (HTTP)**
Ziel	n8n stabil, persistent, update-fähig
Optional	Python via Task Runner

Kursbeschreibung

In diesem Free-Kurs richtest du dir Schritt für Schritt **n8n** auf deinem Raspberry Pi ein – stabil, update-sicher und komplett im **Heimnetz (LAN)**. Du nutzt **Docker & Docker Compose**, damit dein Setup sauber strukturiert bleibt und du n8n später problemlos starten, stoppen und aktualisieren kannst.

Der Fokus liegt auf einem praxisnahen Setup, das nicht nur „irgendwie läuft“, sondern wirklich **betriebssicher** ist: mit **persistenter Datenspeicherung**, klaren Ports und den typischen Troubleshooting-Fällen, die in der Praxis fast immer auftauchen. Optional aktivierst du außerdem **Python** über den Task Runner, um Workflows flexibel zu erweitern.

Meta-Posts automatisieren: Für das Senden von Meta-Posts (ausgehende API-Requests) ist **kein eingehendes HTTPS** nötig.
HTTPS ist erst relevant, wenn externe Dienste **Webhooks an dich senden** sollen.

Was du im Kurs lernen wirst

- Docker & Docker Compose auf Raspberry Pi OS Lite 64-bit installieren
- n8n im LAN starten und im Browser erreichbar machen
- Persistenz korrekt einrichten (damit nichts verloren geht)
- Häufige Fehler beheben (EACCES, Encryption-Key-Mismatch, Cookie-Warnung)
- Logs, Status und Ports sicher prüfen
- Optional: Python im Code-Node aktivieren (Task Runner)

Für wen ist dieser Kurs gedacht?

- Technik-Einsteiger ohne große Vorkenntnisse
- Selbsthoster, die Automationen lokal betreiben möchten
- Maker & Raspberry-Pi-Fans
- Alle, die Cloud-Abos vermeiden möchten
- Nutzer, die ein wartungsfreundliches Docker-Setup wollen

Kursaufbau

- Einführung: Zielbild & Architektur im Überblick
- Docker + Compose installieren und testen
- n8n per docker-compose im LAN einrichten
- Persistenz richtig setzen (damit Updates sicher sind)
- Troubleshooting: die 3 Klassiker aus der Praxis
- Optional: Python Task Runner aktivieren und testen
- Runbook: Start/Stop/Update + Next Steps

Kurs: Pico 2W Webserver mit MicroWebSrv & Bootstrap

MicroPython · WLAN · MicroWebSrv · HTML/Bootstrap · Dashboard-Projekt
Zielgruppe: Einsteiger, Maker, IoT-Neugierige, Technikfans
Projektstatus: **Kurs ist live auf Udemy**

Direktlink zum Kurs: Jetzt auf Udemy ansehen

Kurzüberblick	Details
Plattform	Udemy
Thema	Webserver auf dem Pico 2W (MicroPython)
UI	Moderne Oberflächen mit Bootstrap
Praxis	Dashboard + GPIO/Hardware-Anbindung

Kursbeschreibung

In diesem Kurs lernst du Schritt für Schritt, wie du mit dem **Raspberry Pi Pico 2W** einen vollständigen, stromsparenden **Webserver im Heimnetz** aufbaust – **ohne Linux**, sondern direkt mit **MicroPython** auf dem Mikrocontroller. Du startest bei null: Einrichtung, MicroPython, WLAN-Verbindung und die erste Seite im Browser.

Danach baust du mit **MicroWebSrv** echte Webserver-Strukturen auf (Routen/Handler, statische & dynamische Seiten). Im nächsten Schritt wird es modern: Mit **HTML & Bootstrap** erstellst du Weboberflächen, die nicht nach „Bastelprojekt“, sondern nach echter Web-App aussehen.

Im Praxisteil steuerst du **GPIOs** (z. B. LEDs) über den Browser und kannst Sensoren integrieren. Zum Abschluss machst du dein Projekt **autostartfähig**, sodass der Pico 2W als kleiner Webserver dauerhaft läuft.

Was du im Kurs lernen wirst

- Pico 2W verstehen: Mikrocontroller vs. Computer
- MicroPython installieren (Thonny, Firmware)
- WLAN-Verbindung & IP-Adresse sicher beherrschen
- MicroWebSrv einrichten und „Hello World“ ausliefern
- Routen/Handler: URL → Funktion verstehen
- Statische & dynamische Webseiten ausliefern
- Moderne UI mit HTML & Bootstrap bauen
- GPIOs per Weboberfläche steuern (Dashboard)
- Autostart: Pico 2W Webserver dauerhaft betreiben

Für wen ist dieser Kurs gedacht?

- Absolute Einsteiger ohne Vorkenntnisse
- Maker & Bastler, die mit IoT starten wollen
- Alle, die MicroPython praktisch lernen möchten
- Schüler, Dozenten & Technik-Interessierte
- Webdesigner/Entwickler, die Hardware-Web-Projekte umsetzen möchten
- Alle, die schnelle „Aha-Erlebnisse“ im Heimnetz wollen

Updates, neue Kurse oder Bonusmaterial?

Wenn du künftig Updates, neue Kurse oder Bonusmaterialien bekommen möchtest, trage dich in meinen Newsletter ein.