Serie: Digitalcraft WikiCore – CoreStack **Ziel:** Modernes Self-Hosting auf dem Raspberry Pi 4 mit Docker/Compose – inkl. n8n, Gitea, VS Code Server, Filebrowser und DokuWiki als **Digitalcraft WikiCore**. **Hinweis:** Die SSD ist bereits vorbereitet (Raspberry Pi OS läuft direkt auf der 1TB-SSD). Das SSD-Setup wurde in einem früheren Video gezeigt.

---

1) Titelvorschläge (SEO + Klickmotiv)

1. **Raspberry Pi 4 CoreStack: Docker + n8n + DokuWiki (WikiCore) in einem System**
2. **Modern Self-Hosting auf dem Raspberry Pi 4: Mein Digitalcraft CoreStack**
3. **Pi 4 (2GB) als Home-Server: Docker-Stack mit n8n, Gitea, VS Code & DokuWiki**
4. **Von Null zum CoreStack: Docker-Plattform auf SSD (Pi 4) – sauber & wartbar**
5. **Digitalcraft WikiCore: Die moderne Kurs-Zentrale auf Raspberry Pi 4 (Docker)**

---

2) Thumbnail-Hook (kurz, hart, klickbar)

1. **CORESTACK in 30 MIN**
2. **Pi4 + Docker = MODERN**
3. **n8n + WikiCore + Git**
4. **SD? NEIN. SSD!**

---

3) Kapitelmarken (YouTube Chapters)

1. **0:00** Intro: Digitalcraft WikiCore & CoreStack-Ziel
2. **0:45** Voraussetzung: Raspberry Pi OS läuft direkt auf SSD (Link zum SSD-Video)
3. **1:15** Digitalcraft-Standard: Ordnerstruktur `/srv/docker/corestack`
4. **2:30** System-Update + Architektur-Check (64-bit?)
5. **4:00** Docker + Compose installieren
6. **7:00** CoreStack Deploy: Postgres + n8n + DokuWiki + Gitea + VS Code + Filebrowser
7. **11:30** Proof: Alle Dienste laufen (URLs/Ports)
8. **13:30** Ausblick: **Video 02 = Migration Pi3 → Pi4 (Cutover)** + Checkliste

---

4) Sprechertext (fertig zum Ablesen)

Intro (0:00 – 0:45)

„Willkommen im **Digitalcraft WikiCore-Universum**. Heute bauen wir den **CoreStack** auf einem Raspberry Pi 4 mit 2GB RAM – modern, wartbar und reproduzierbar: Docker + Compose, dazu n8n, Gitea, VS Code Server, Filebrowser und unser Kurs-Wiki **Digitalcraft WikiCore**."

On-Screen: Overlay „Digitalcraft WikiCore – CoreStack v1".

Voraussetzung (0:45 – 1:15)

„Wichtig: Das Betriebssystem läuft bei mir **direkt auf der 1TB-SSD**. Das SSD-Setup habe ich bereits in einem eigenen Video gezeigt – heute starten wir genau dort: System ist da, jetzt kommt die Plattform."

On-Screen: `df -h` + kurzer Blick in `lsblk`.

Digitalcraft-Standard (1:15 – 2:30)

„Bevor wir irgendwas installieren, legen wir unseren **Digitalcraft-Standard** fest: Alles, was Docker betrifft, liegt sauber unter `/srv/docker/corestack`. So bleibt das System nachvollziehbar – und später ist Migration oder Backup kein Ratespiel."

Docker Installation (2:30 – 7:00)

„Jetzt kommt das Fundament: Docker Engine und das Compose-Plugin. Profi-Hinweis: 64-bit ist langfristig die bessere Basis. Ich zeige euch kurz den Check – danach installieren wir Docker sauber."

CoreStack Deploy (7:00 – 11:30)

„Jetzt wird's spannend: Wir starten unseren ersten Digitalcraft CoreStack. Ziel ist nicht Overkill, sondern: **alle Dienste laufen stabil**, Daten sind persistent, und wir haben eine Plattform, die wir später migrieren und erweitern können."

Proof & Ausblick (11:30 – Ende)

„Wenn ihr das hier seht, habt ihr eine moderne Self-Hosting-Plattform auf dem Pi – komplett auf SSD, sauber als Stack. Im nächsten Video kommt der kritische Part: **Migration Pi 3 → Pi 4**, inklusive Cutover und Checks. Link zur Checkliste und zum Compose-Pack findet ihr in der Beschreibung."

---

9) YouTube-Beschreibung (fertig zum Einfügen)

Digitalcraft CoreStack (Pi 4 / 2GB / 1TB SSD):
Heute bauen wir eine moderne Self-Hosting-Plattform mit Docker & Compose – inklusive **n8n**, **Gitea**, **VS Code Server**, **Filebrowser** und **DokuWiki als Digitalcraft WikiCore**.

Ergebnis: Alles läuft als Stack, Daten sind persistent, Struktur ist sauber.
Nächstes Video: **Migration Pi 3 → Pi 4 (Cutover + Checks)**

Downloads (kostenlos): CoreStack-Compose + Checkliste + WikiCore-Startseite → Link in der Beschreibung.

Chapters:
0:00 Intro … (Kapitel einfügen)

---

10) Nächster Schritt (Video 02 Teaser)

Nächstes Video: Migration Pi 3 → Pi 4

1. Freeze (Dienste stoppen)
2. Backup (Volumes/Configs)
3. Transfer (rsync)
4. Restore (Pi 4)
5. Funktionstest + Cutover

Ziel: Umstieg ohne Chaos – nachvollziehbar, wiederholbar, Digitalcraft-Style.
