

Poke-Scan V2 - Daily Report

Mittwoch, 19. Februar 2026 | Tag 2 der Entwicklung

Zusammenfassung

OpenClaw Setup finalisiert und Agent einsatzbereit! Docker Container laufen stabil, Kimi K2.5 als Primary Model bestätigt. Telegram Bot @herbekantebot erstellt und mit OpenClaw verbunden. Erste erfolgreiche Kommunikation mit dem Agent über Telegram. Workspace auf poke-scan-v2 Repository konfiguriert. Grundstein für automatisierte Entwicklung gelegt.

Erledigte Aufgaben

Nr.	Aufgabe	Status
1	OpenClaw Setup Wizard abgeschlossen	✓
2	Docker Container gestartet (docker compose up -d)	✓
3	Logs validiert - keine Anthropic-Fehler mehr	✓
4	Kimi K2.5 als Primary Model bestätigt	✓
5	Telegram Bot @herbekantebot bei BotFather erstellt	✓
6	Bot Token in OpenClaw .env konfiguriert	✓
7	Telegram Bot mit OpenClaw Gateway verbunden	✓
8	Erste Test-Nachricht an Bot gesendet	✓
9	Agent antwortet korrekt über Telegram	✓
10	Workspace auf poke-scan-v2 Repo gesetzt	✓
11	Git-Zugriff des Agents getestet	✓
12	Agent-Richtlinien (openclaw-agent-richtlinien.md) finalisiert	✓

Technische Highlights

- Docker Compose Stack läuft stabil (openclaw-gateway + services)
- NVIDIA NIM API Endpoint: `integrate.api.nvidia.com/v1`
- Primary Model: `openai:moonshotai/kimi-k2.5`
- Telegram Bot API erfolgreich integriert
- Git SSH-Zugriff vom Container aus funktioniert
- Workspace-Pfad: `~/openclaw/workspace/poke-scan-v2`

Git Commits

- [Docs] `openclaw-agent-richtlinien.md` erstellt
- [Docs] `PROJEKT-ANWEISUNGEN.md` aktualisiert
- [Config] `.env.example` mit NVIDIA NIM Variablen

Aktueller Status

Komponente	Status
VPS Server	■ Läuft stabil
Docker Container	■ Alle Services online

OpenClaw Gateway	■ Aktiv
Kimi K2.5 Model	■ Als Primary bestätigt
Telegram Bot	■ Verbunden (@herbekantebot)
Git Integration	■ SSH funktioniert
Workspace	■ poke-scan-v2 konfiguriert
Web-App	■ Noch nicht deployed

■ Nächste Schritte

- Erste echte Entwicklungsaufgabe an Agent senden
- kimiVision.ts Service über Agent erstellen lassen
- Vercel Deployment einrichten
- CardScanner Komponente implementieren
- End-to-End Test der Kartenerkennung

■ Agent-Workflow etabliert

Der Workflow für die Zusammenarbeit ist jetzt klar definiert: 1. Marcel beschreibt Aufgabe an Claude (Chat) 2. Claude plant und schreibt Code 3. Marcel kopiert Anweisungen zu Telegram → @herbekantebot 4. Agent führt aus und meldet Status zurück 5. Vercel deployed automatisch bei Push 6. Marcel prüft Preview und merged nach main