

Aufgaben-Liste: Lokale KI für deine Projekte

Was du selbst machen kannst

Phase 1: Setup & Grundlagen (Heute)

- [] ****Ollama installieren****
- ``brew install ollama`` oder von `ollama.ai`
- ``ollama serve`` starten
- [] ****Erstes Modell herunterladen****
- ``ollama pull llama3.2:3b`` (empfohlen für Start)
- Testen: ``ollama run llama3.2:3b``
- [] ****Open WebUI installieren****
- Docker installieren (falls nicht vorhanden)
- Open WebUI starten
- Browser öffnen: ``http://localhost:3000``
- [] ****Ersten Chat testen****
- Einfache Frage stellen
- Code-Beispiel generieren lassen

Phase 2: Code-Assistent (Diese Woche)

- [] ****Code-Erklärungen****
- Komplexen Code hochladen
- Erklären lassen, was er macht
- Verbesserungsvorschläge einholen
- [] ****Code-Generierung****
- Funktionen schreiben lassen
- Tests generieren
- Dokumentation erstellen
- [] ****Bug-Fixes****
- Fehlerhaften Code hochladen
- Fehler finden lassen
- Fixes vorschlagen lassen
- [] ****Refactoring****
- Alten Code verbessern lassen
- Best Practices anwenden
- Code-Stil vereinheitlichen

Phase 3: Lernen & Anpassen (Nächste Woche)

- [] ****RAG Setup (Retrieval Augmented Generation)****
- Deine Projekt-Dokumentation hochladen
- Code-Base hochladen
- KI lernt aus deinen Dateien

- [] ****Context Memory aktivieren****
- Einstellungen → Memory aktivieren
- KI erinnert sich an frühere Gespräche
- [] ****Custom Prompts erstellen****
- Spezielle Prompts für deine Aufgaben
- Als Templates speichern
- Wiederverwendbar machen
- [] ****Code-Stil lernen lassen****
- Mehrere Code-Beispiele hochladen
- KI lernt deinen Stil
- Generiert Code in deinem Stil

Phase 4: Aktives Eingreifen (Später)

- [] ****Agenten erstellen****
- Agent mit Tools definieren
- Kann Dateien lesen/schreiben
- Kann Commands ausführen
- [] ****Auto-Fixes****
- Agent überwacht Code
- Findet Fehler automatisch
- Schlägt Fixes vor (oder fixt selbst)
- [] ****Code-Reviews****
- Pull Requests automatisch reviewen
- Verbesserungen vorschlagen
- Best Practices prüfen
- [] ****Dokumentation automatisch****
- Code-Analyse
- Dokumentation generieren
- README aktualisieren

Phase 5: Spezialisierung (Fortgeschritten)

- [] ****Fine-tuning****
- Auf deinen Code-Stil trainieren
- Spezifische Domäne lernen
- Bessere Ergebnisse
- [] ****Custom Tools****
- Eigene Tools entwickeln
- Integration mit deinen Tools
- Workflows automatisieren
- [] ****Multi-Agent System****
- Mehrere Agenten für verschiedene Aufgaben
- Zusammenarbeit zwischen Agenten
- Komplexe Aufgaben lösen

Konkrete Projekte für deine moOde-KI

Projekt 1: Code-Dokumentation

- ☐ Alle Scripts analysieren
- ☐ Automatische Dokumentation generieren
- ☐ README aktualisieren

Projekt 2: Bug-Detection

- ☐ Code auf Fehler scannen
- ☐ Potentielle Probleme finden
- ☐ Fixes vorschlagen

Projekt 3: Test-Generierung

- ☐ Tests für alle Scripts generieren
- ☐ Edge Cases finden
- ☐ Test-Coverage verbessern

Projekt 4: Code-Review-Assistent

- ☐ Neue Änderungen reviewen
- ☐ Best Practices prüfen
- ☐ Verbesserungen vorschlagen

Projekt 5: Deployment-Automation

- ☐ Deployment-Steps analysieren
- ☐ Automatisierung vorschlagen
- ☐ Fehler-Prävention

Tägliche Aufgaben

Morgen-Routine

- ☐ KI fragt: "Was arbeitest du heute?"
- ☐ Aufgaben planen lassen
- ☐ Prioritäten setzen

Während der Arbeit

- ☐ Code-Fragen stellen
- ☐ Fehler erklären lassen
- ☐ Lösungen finden lassen

Abend-Routine

- ☐ Code-Review des Tages
- ☐ Verbesserungen notieren

- [] Morgen vorbereiten

Lern-Ressourcen

Dokumentation

- [] Ollama Docs lesen: <https://ollama.ai/docs>
- [] Open WebUI Docs: <https://docs.openwebui.com>
- [] RAG Tutorial durcharbeiten

Community

- [] Ollama Discord beitreten
- [] Open WebUI GitHub Issues lesen
- [] Best Practices sammeln

Experimentieren

- [] Verschiedene Modelle testen
- [] Verschiedene Prompts ausprobieren
- [] Tools kombinieren

Erfolgs-Metriken

Woche 1

- ■ KI läuft lokal
- ■ Erste Code-Generierung funktioniert
- ■ Grundlegende Fragen beantwortet

Woche 2

- ■ RAG funktioniert
- ■ Code-Erklärungen sind hilfreich
- ■ Bug-Fixes werden gefunden

Woche 4

- ■ Agenten arbeiten selbstständig
- ■ Auto-Fixes funktionieren
- ■ Zeitersparnis messbar

Tipps

1. ****Starte klein****: Beginne mit einfachen Aufgaben
2. ****Experimentiere****: Probiere verschiedene Prompts
3. ****Lerne kontinuierlich****: KI wird besser mit mehr Daten
4. ****Dokumentiere****: Notiere, was funktioniert
5. ****Teile Erfahrungen****: Community hilft weiter

Viel Erfolg! ■