KAVANARA

CONSULTING · TECHNICALS · SOLUTIONS

METHODEN & WERKZEUGE

KI-Assistent - Die Zukunft in der Gegenwart

Version 1.0 · 2026

© Matthias Leo Wegner · kavanara.de

KI-Assistent - Die Zukunft in der Gegenwart

Dein persönliches KI-Wissensmanagement-System einrichten. Claude AI mit Obsidian verbinden.

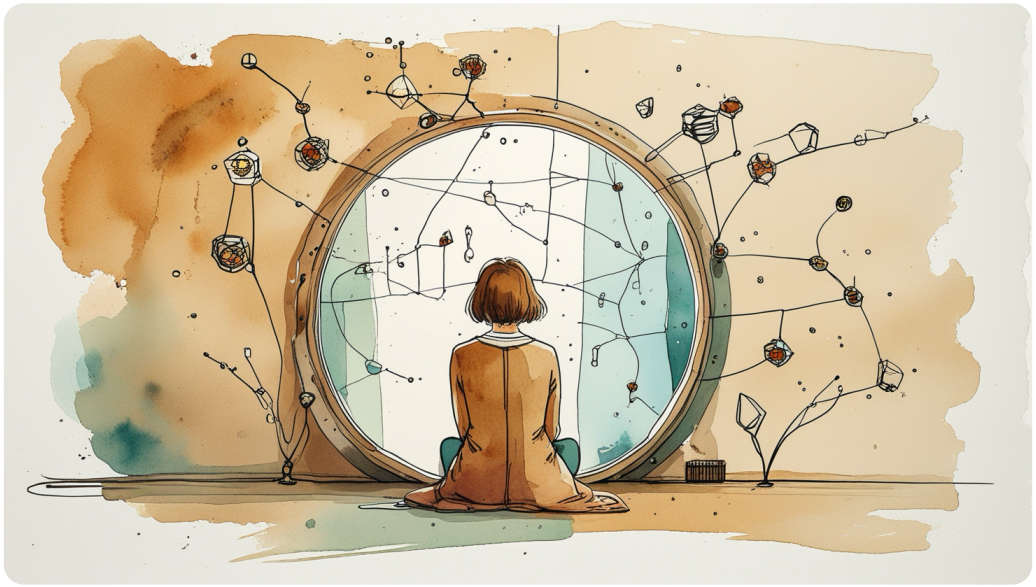
Stell dir vor: Du sprichst mit einer KI, die dein gesamtes Notizsystem kennt. Kein Copy-Paste mehr. Kein manuelles Zusammensuchen. Ein Assistent, der versteht, was du weißt – und damit arbeiten kann.

Das ist Heute möglich. Jetzt. Wer Ironman gesehen hat, kennt das Konzept: Jarvis spricht mit dir.

Die Verbindung von **Claude AI** mit **Obsidian** macht es möglich. Der Schlüssel heißt **Model Context Protocol (MCP)** – ein Standard, der KI-Systeme mit deinen Daten verbindet.

Diese Anleitung erklärt jeden Schritt. Auch wenn du noch nie von APIs oder Servern gehört hast.

Teil 1: Was du verstehen musst



Obsidian – deine Wissensdatenbank

Obsidian ist eine kostenlose Notiz-App, die auf deinem Computer läuft. Das Besondere daran: Deine Notizen sind einfache Textdateien im sogenannten Markdown-Format. Der Vorteil dieses Wiki-Formats liegt in der Trennung von Inhalt und Format – wie die Inhalte in verschiedenen Umgebungen dargestellt werden, entscheidest du selbst, zum Beispiel mit sogenannten CSS-Snippets.

Alle Inhalte gehören dir. Sie liegen nicht in einer Cloud, und du entscheidest, was du damit machst. Jede Datei, die du in Obsidian ablegst, bleibt lokal auf deinem Rechner. Keine Abhängigkeit von irgendeinem Anbieter.

Obsidian speichert die Daten in einem **Vault** – einem Dateiordner, in dem alle deine Notizen liegen. Wie ein Tresor für dein Wissen.

Claude – dein KI-Assistent

Claude ist eine KI von Anthropic, vergleichbar mit ChatGPT. Sie kann Texte schreiben, Fragen beantworten und komplexe Probleme analysieren. Welche KI gerade die „beste" ist, hängt vom Kontext und der Technologieentwicklung ab. Aktuelle Stände und Vergleiche findest du unter https://lmarena.ai/de/leaderboard.

Claude Desktop ist die App für deinen Computer. Direkt chatten, ohne Browser.

MCP – die Brücke

Das Model Context Protocol funktioniert wie ein standardisierter Stecker – denk an USB-C. Es verbindet verschiedene KI-Anwendungen mit verschiedenen Datenquellen. Ein Standard statt tausend Einzellösungen.

Wenn du Claude fragst „Was steht in meiner Notiz über Projektmanagement?", passiert folgendes: Claude erkennt, dass es Zugriff auf Obsidian braucht. Es fragt den MCP Server, ob es dort etwas gibt. Der Server durchsucht deinen Vault und liefert die Ergebnisse. Claude antwortet basierend auf deinen Notizen.

REST API – der Kellner

REST steht für „Representational State Transfer" – ein Datenprotokoll. Eine standardisierte Sprache, mit der Programme miteinander reden. Wie HTTP für Websites, nur für den Datenaustausch zwischen Anwendungen.

API bedeutet „Application Programming Interface" – eine Schnittstelle. Der definierte Weg, wie ein Programm einem anderen sagt: „Gib mir das" oder „Speichere das".

Zusammen ist ein REST API ein standardisierter Kommunikationsweg zwischen Programmen.

Jetzt die Metapher. Denk an ein Restaurant. Du (der Client) gibst dem Kellner eine Bestellung. Der Kellner bringt sie in die Küche. Die Küche bereitet zu. Der Kellner bringt dir das Essen.

Das Local REST API Plugin ist dieser Kellner. Es leitet Anfragen von Claude an deinen Vault weiter und bringt die Ergebnisse zurück. Die „Bestellungen" folgen dabei einem festen Format (REST), damit beide Seiten sich verstehen.

Teil 2: Software installieren

Schritt 1: Obsidian herunterladen

Gehe zu https://obsidian.md/download und lade die Version für dein Betriebssystem herunter.

Unter **Windows** machst du einen Doppelklick auf die heruntergeladene Datei und folgst dem Installationsassistenten. Unter **Mac** öffnest du die .dmg-Datei und ziehst Obsidian in den Applications-Ordner. Unter **Linux** funktioniert es am einfachsten mit AppImage (`chmod +x Obsidian-*.AppImage` und dann `./Obsidian-*.AppImage`) oder via Snap (`sudo snap install obsidian --classic`).

Schritt 2: Deinen ersten Vault erstellen

Beim ersten Start fragt Obsidian, ob du einen neuen Vault erstellen oder einen bestehenden öffnen möchtest. Klicke auf „Create new vault", gib einen Namen ein (zum Beispiel „Mein Wissen"), wähle einen Speicherort wie `Dokumente/Obsidian` und klicke auf „Create".

Wichtig: Merke dir den Pfad. Du brauchst ihn später.

Schritt 3: Claude Desktop installieren

Gehe zu https://claude.ai/download, lade die Desktop-Version herunter und installiere sie. Melde dich dann mit deinem Claude-Konto an – falls du noch keins hast, kannst du es dort einrichten.

Teil 3: Die Obsidian-Plugins

Obsidian wird durch Plugins erweitert. Du brauchst drei davon.

Community Plugins aktivieren

Öffne Obsidian und klicke unten links auf das Zahnrad für die Einstellungen. Wähle „Community plugins" und klicke auf „Turn on community plugins". Bestätige die Warnung – damit erlaubst du die Installation von Erweiterungen aus der Community.

Plugin 1: Local REST API (Pflicht)

Das ist das Herzstück. Es macht deinen Vault für externe Programme zugänglich.

Klicke bei Community plugins auf „Browse" und suche nach „Local REST API". Installiere das Plugin von Adam Coddington und aktiviere es mit „Enable". In den Plugin-Einstellungen findest du den **API Key** und den **Port** (Standard: 27123) – notiere dir beides, du brauchst es später.

Plugin 2: Templater (Empfohlen)

Dieses Plugin ermöglicht fortgeschrittene Vorlagen. Klicke auf „Browse", suche nach „Templater", installiere das Plugin von SilentVoid und aktiviere es.

Plugin 3: Smart Connections (Optional)

Mit diesem Plugin kann Claude Notizen nach Bedeutung finden, nicht nur nach Stichworten – sogenannte semantische Suche. Klicke auf „Browse", suche nach „Smart Connections", installiere das Plugin von Brian Petro und aktiviere es. Beim ersten Start werden deine Notizen indiziert, was je nach Vault-Größe etwas dauern kann.

Teil 4: MCP Tools einrichten

Das MCP Tools Plugin

Das ist die eigentliche Brücke zwischen Obsidian und Claude.

Gehe zu Community plugins, klicke auf „Browse" und suche nach „MCP Tools" oder „Obsidian MCP Tools". Installiere das Plugin von Jack Steam und aktiviere es. Öffne dann die Plugin-Einstellungen – du siehst dort eine Abhängigkeitsprüfung, die anzeigt, welche anderen Plugins erkannt wurden. Klicke auf „Download MCP Server" und warte, bis der Download abgeschlossen ist.

Alternative: Manuelles Setup

Falls das automatische Setup nicht funktioniert, kannst du den MCP Server auch manuell einrichten. Dafür muss Node.js installiert sein (erhältlich unter nodejs.org). Öffne dann ein Terminal und führe folgenden Befehl aus:

```
npx @anthropic-ai/mcp-obsidian --vault /pfad/zu/deinem/vault
```

Ersetze den Pfad mit dem tatsächlichen Pfad zu deinem Vault.

Unten links in Obsidian sollten du mit erfolgreicher Installation bei Eistellungen > Externe Tools > MCP Tools folgendes sehen:

Teil 5: Claude Desktop konfigurieren

Jetzt muss Claude wissen, wo der MCP Server ist.

Konfigurationsdatei finden und bearbeiten

Unter **Windows** findest du die Konfigurationsdatei unter `C:\Users\DEIN_NAME\AppData\Roaming\Claude\claude_desktop_config.json`. Unter **Mac** liegt sie unter `~/Library/Application Support/Claude/claude_desktop_config.json`.

Öffne die Datei mit einem Texteditor. Falls sie nicht existiert, erstelle sie. Füge folgenden Inhalt ein oder ergänze ihn:

```
{
  "mcpServers": {
    "obsidian-mcp-server": {
      "command": "node",
      "args": ["/pfad/zum/obsidian-mcp-server/build/index.js"],
      "env": {
        "OBSIDIAN_API_KEY": "DEIN_API_KEY_HIER",
        "OBSIDIAN_PROTOCOL": "https",
        "OBSIDIAN_HOST": "127.0.0.1",
        "OBSIDIAN_PORT": "27124"
      }
    }
  }
}
```

Ersetze `DEIN_API_KEY_HIER` mit dem API Key aus dem Local REST API Plugin. Passe den Pfad zum MCP Server an und stelle sicher, dass der Port mit dem in Obsidian konfigurierten übereinstimmt. Speichere die Datei und **starte Claude Desktop neu**.

Teil 6: Testen

Nach dem Neustart sollte die Verbindung funktionieren.

Verbindung prüfen

Schreibe in Claude Desktop:

```
Kannst du auf meinen Obsidian Vault zugreifen?
Erstelle eine Testnotiz mit dem Namen "Verbindungstest".
```

Wenn Claude bestätigt und eine Notiz erstellt, ist alles klar.

Weitere Tests

Probiere aus, ob Claude alle Dateien in deinem Vault auflisten kann. Lass es nach Notizen suchen, die ein bestimmtes Wort enthalten. Oder bitte Claude, eine neue Notiz mit einem beliebigen Inhalt zu erstellen. Wenn das alles funktioniert, ist die Integration erfolgreich eingerichtet.

Teil 7: Was du jetzt tun kannst

Die Möglichkeiten sind vielfältig.

Du kannst Claude bitten, aus einem langen Meeting-Protokoll eine strukturierte Zusammenfassung zu erstellen und sie direkt in deinem Vault zu speichern. Du kannst alle Notizen zu einem bestimmten Thema suchen lassen und dir eine Übersicht der Verbindungen erstellen lassen. Du kannst aus Gesprächen mit Claude strukturierte Lernnotizen mit Code-Beispielen generieren. Oder du automatisierst deine täglichen Notizen – mit Abschnitten für Aufgaben, Ideen und Notizen.

Die KI wird zu deinem persönlichen Wissensassistenten, der dein gesamtes Notizsystem kennt und damit arbeiten kann.

Teil 8: Wenn es nicht funktioniert

„Claude kann nicht auf den Vault zugreifen"

Das kann mehrere Ursachen haben. Prüfe zuerst, ob das Local REST API Plugin in Obsidian aktiviert ist – gehe zu Einstellungen, dann Community plugins. Wenn das Plugin aktiv ist, könnte der API Key falsch sein. Kopiere ihn erneut aus den Plugin-Einstellungen, füge ihn in die Claude-Konfiguration ein und starte Claude Desktop neu.

Falls das nicht hilft, könnte deine Firewall die Verbindung blockieren. Erlaube Obsidian und Claude Desktop in deiner Firewall – der Port 27123 oder 27124 muss erreichbar sein. Bei einem Port-Konflikt kannst du einen anderen Port in den Local REST API Einstellungen wählen und die Claude-Konfiguration entsprechend anpassen.

„MCP Server startet nicht"

Prüfe zuerst, ob Node.js installiert ist. Öffne ein Terminal und tippe `node --version`. Wenn keine Version erscheint, installiere Node.js von nodejs.org.

Falls Node.js installiert ist, überprüfe alle Pfade in der Konfiguration – sie müssen korrekt und absolut sein (also vollständige Pfadangaben). Unter Mac und Linux müssen die Skripte außerdem ausführbar sein.

Teil 9: Sicherheit

Alles bleibt lokal

Das ist der größte Vorteil: **Deine Daten bleiben auf deinem Computer.**

Der MCP Server läuft lokal. Die Kommunikation nutzt HTTPS mit Zertifikaten. Deine Notizen werden nicht in irgendwelche Cloud-Systeme hochgeladen. Du behältst die volle Kontrolle.

API Key und Zugriffsrechte

Teile deinen API Key niemals öffentlich. Bei Verdacht auf Kompromittierung generierst du einfach einen neuen. In den Plugin-Einstellungen kannst du außerdem festlegen, welche Zugriffsrechte Claude hat: nur Lesen, Lesen und Schreiben, oder voller Zugriff inklusive Löschen. Wähle die Stufe, die zu deinem Sicherheitsbedürfnis passt.

Was das bedeutet

Die Verbindung von Claude mit Obsidian eröffnet neue Möglichkeiten. Du kannst Notizen automatisch aus Gesprächen erstellen lassen. Du kannst semantisch suchen – nach Bedeutung, nicht nur nach Stichworten. Claude erkennt Zusammenhänge zwischen deinen Notizen und kann strukturierte Lernmaterialien aus komplexen Themen generieren. Tägliche Notizen, Protokolle, Dokumentation – vieles lässt sich automatisieren.

Und das Beste: Du behältst die Kontrolle. Über deine Daten. Über dein Wissen. Über dein System.

Ressourcen

Alle Links zum Nachschlagen:

Obsidian findest du unter https://obsidian.md/download. **Claude Desktop** gibt es unter https://claude.ai/download. Das **Obsidian MCP Tools** Plugin ist auf GitHub unter https://github.com/jacksteamdev/obsidian-mcp-tools verfügbar. Das **Local REST API** Plugin findest du unter https://github.com/coddingtonbear/obsidian-local-rest-api und **Smart Connections** unter https://github.com/brianpetro/obsidian-smart-connections. Die offizielle **MCP Dokumentation** liegt unter https://modelcontextprotocol.io.

Menschen sind zum Geliebtwerden geschaffen, nicht zum Funktionieren.
Und wenn wir Organisationen als Gewächshäuser gestalten
– Räume, die schützen, nähren, ermöglichen –
dann kann etwas wachsen, das größer ist als wir selbst.

KAVANARA
CONSULTING · TECHNICALS · SOLUTIONS

Mehr erfahren: **kavanara.de**