

Presented By Sophie A / 2026-01-21 / Everything Open 2026

# Running Agentic AI Locally

# In this Workshop:

- **Local vs Closed & Vendor Hosted AI**
- Installing the Tools
- **Reactive vs Agentic**
- Models for your Machine
- MCP Servers
- Advanced Tools

# What is Local AI

- o
- o
- o
- o
- o
- 

## Vendor AI

Closed Weight

Used only online and through vendor tool and/or API

Unknown exactly how your data is used

### Examples:

ChatGPT

Grok

Claude

DeepSeek (API / vendor-hosted)

## Local AI

Open Weight / Open Source

Many options of how to run, eg locally through tool, through API,

### Examples:

Llama

Mistral

Deepseek (Local / self-hosted)

The screenshot shows a GitHub repository page for 'local-agentic-mcp-workshop'. The repository was created by 'starship-droid' and has two commits: 'Initial Commit' (89d6f84, 47 minutes ago) and 'meeting\_minutes.md' (Initial Commit, 47 minutes ago). The README file is the active tab, containing the following content:

## Walkthrough: Running Agentic AI Locally on your Machine (GNU/Linux and Mac)

[Github Repo](#) [View Slides](#) [JanAI Quickstart](#)

### Getting Started

### Prerequisites for JanAI

- Minimum specs for a decent experience:

# Github Repo

<https://github.com/starship-droid/local-agentic-mcp-workshop/>

Go to README.md



# Hub



Search for models on Hugging Face...

Newest  Downloaded

**Jan-V1-4B-GGUF** 2.3 GB [Download](#)

Jan-v1 is an advanced agentic language model built on the Qwen3-4B-thinking model, optimized for complex problem-solving and integration with...

By Janhq  86716  4   Show variants

**Jan-V2-VL-High-Gguf** 4.7 GB [Download](#)

Multimodal agent for stable long-horizon task execution in software environments.

By Janhq  54723  14   Show variants

Search

+ New Chat

Projects

Recents

Hello

Hub

Settings



# How to Choose Models

Parameters		Reasoning	MCP	Vision
RAM	Model			
8GB	3B	Shows “thinking” on the screen Reasoning models can plan, loop, and achieve goals	Tools for LLM Examples: Searching on Web Filesystem Chrome Developer Tools	Model can analyse images Different models are better/worse at interpreting images
16GB	7B			
32GB	13B			

## Compatibility

- Recommended for your device

# How to Choose Models

Parameters		Reasoning	MCP	Vision
RAM	Model			
8GB	3B	Shows “thinking” on the screen Reasoning models can plan, loop, and achieve goals	Tools for LLM Examples: Searching on Web Filesystem Chrome Developer Tools	Model can analyse images Different models are better/worse at interpreting images
16GB	7B			
32GB	13B			

## Compatibility

- Recommended for your device

# Reactive vs Agentic AI

- o
- o
- o
- o
- o
- 

## Reactive

Responds only to direct prompts

Single-step execution

No goal-setting or planning

Limited or no memory

Human controls every action

## Agentic

Goal-driven and autonomous

Plans and executes multiple steps

Can decide what to do next

Uses memory and reasoning

Human supervises, not micromanages



# Start Chat

Search for models on Hugging Face...

Newest  Downloaded

Qwen3-VL-8B-Thinking-1M-GGUF 4.3 GB

Qwen3-VL-8B-Thinking is a reasoning-enhanced vision-language model for image-text-to-text tasks with 256K context.

By Unslloth 59334 1 1

Projects

No threads yet

Start a new conversation to see your thread history here.

Hub

Settings

Jan

Hi, how are you?

How can I help you today?

Ask me anything...

Qwen3-VL-8B-Thinking-1M-IQ...

Shows model downloaded and usable

Tools available: image, attachment, web search

# Enable MCP Permission

The screenshot shows the Jan Browser settings interface. The left sidebar has a 'Settings' button highlighted with a red box. The main content area shows the 'MCP Servers' section, which is also highlighted with a red box. The 'Allow All MCP Tool Permissions' toggle switch is shown with a red box around it. Below it is a 'Tool call timeout (seconds)' input field set to 30. The 'Browser Use Proactive Mode' toggle switch is also highlighted with a red box. At the bottom, there is a list of MCP servers, with the first one, 'Jan Browser MCP', highlighted with a red box. The 'Install Extension' button is visible at the bottom of this section.

Settings

General

Attachments

Interface

Privacy

Model Providers >

Assistants

Shortcuts

Hardware

MCP Servers

Local API Server

HTTPS Proxy

Extensions

MCP Servers Experimental

Find more MCP servers at [mcp.so](https://mcp.so)

Allow All MCP Tool Permissions

When enabled, all MCP tool calls will be automatically approved without showing permission dialogs. This setting applies globally to all conversations, including new chats.

Tool call timeout (seconds)

Maximum time to wait for an MCP tool response before timing out.

30

Browser Use Proactive Mode Experimental

When enabled, automatically captures browser screenshots to provide additional context for browser automation tasks. Requires Jan Browser MCP with vision and tool use model.

Jan Browser MCP Official

Transport: STDIO

Command: npx

Args: -y, search-mcp-server@latest

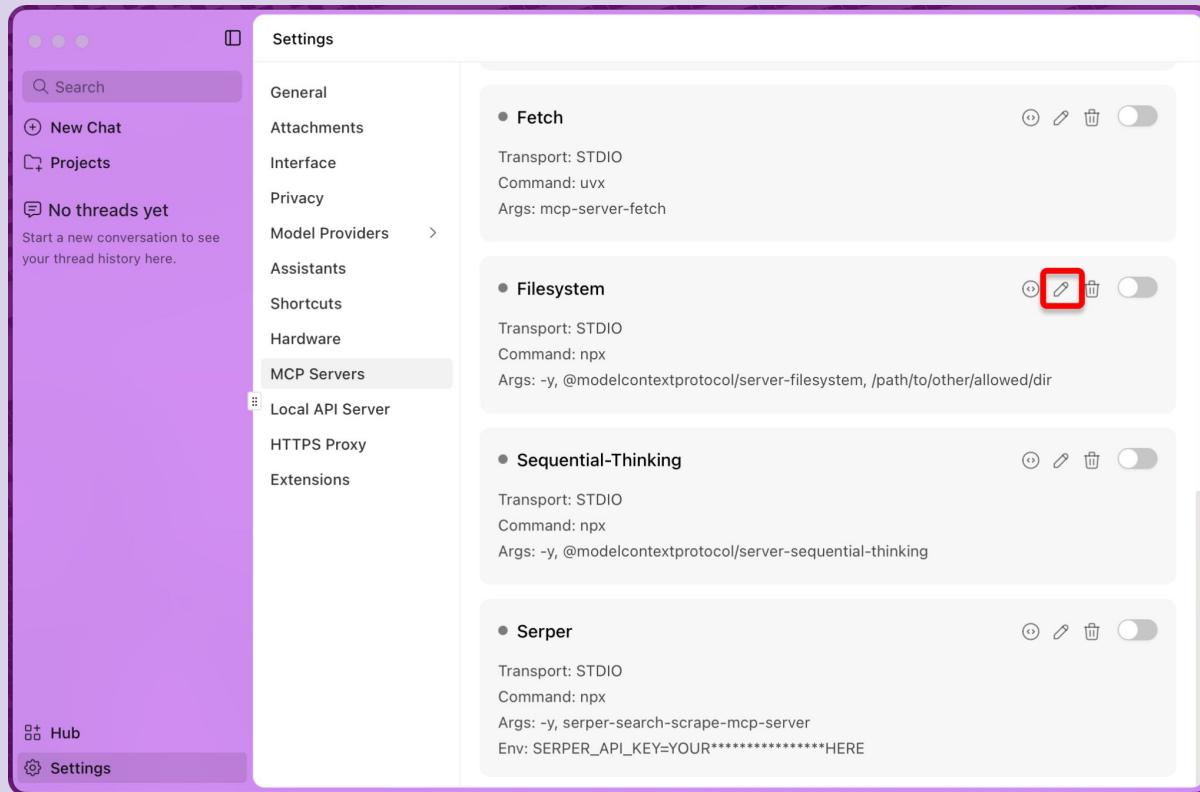
Env: BRIDGE\_HOST=127.0.1, BRIDGE\_PORT=\*\*\*\*\*

Requires Jan Browser Extension to be installed in your Chrome-based browser.

[Install Extension →](#)

Browsermem

# Edit Filesystem MCP



Settings

General

Attachments

Interface

Privacy

Model Providers >

Assistants

Shortcuts

Hardware

MCP Servers

Local API Server

HTTPS Proxy

Extensions

Fetch

Transport: STDIO

Command: uvx

Args: mcp-server-fetch

Filesystem

Transport: STDIO

Command: npx

Args: -y, @modelcontextprotocol/server-filesystem, /path/to/other/allowed/dir

Sequential-Thinking

Transport: STDIO

Command: npx

Args: -y, @modelcontextprotocol/server-sequential-thinking

Serpcer

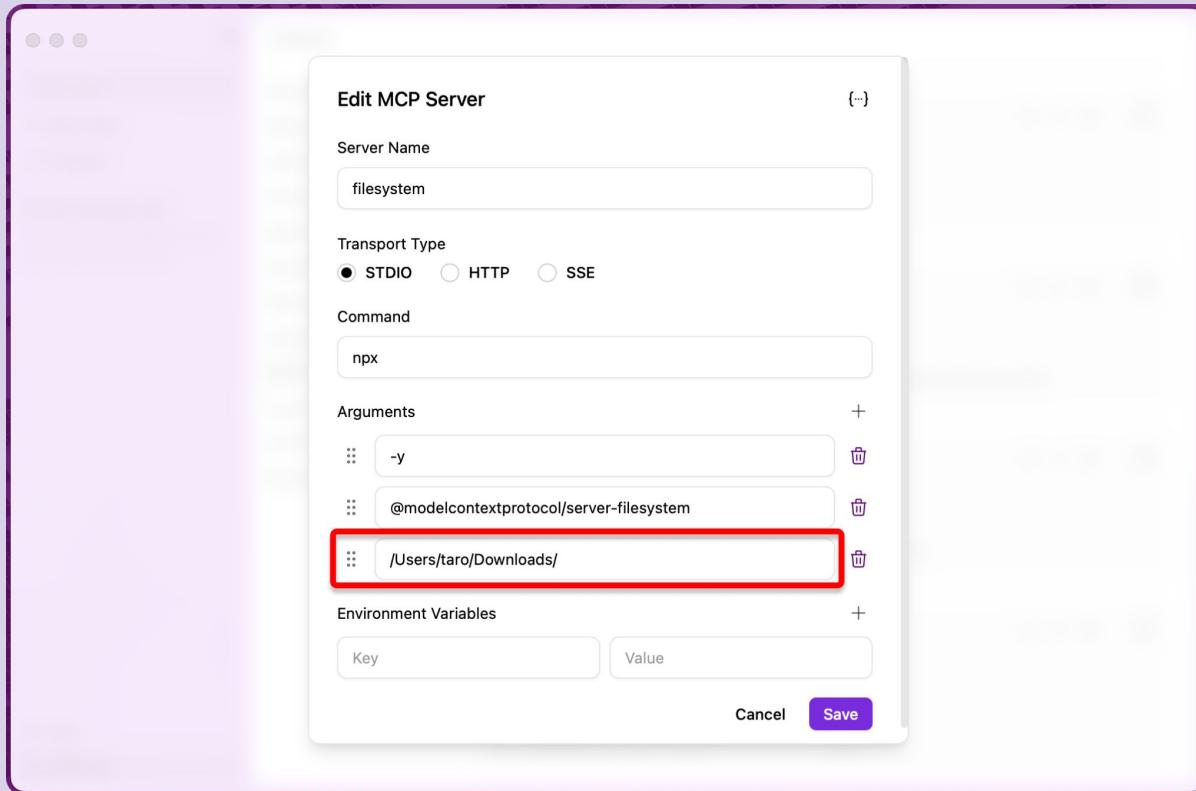
Transport: STDIO

Command: npx

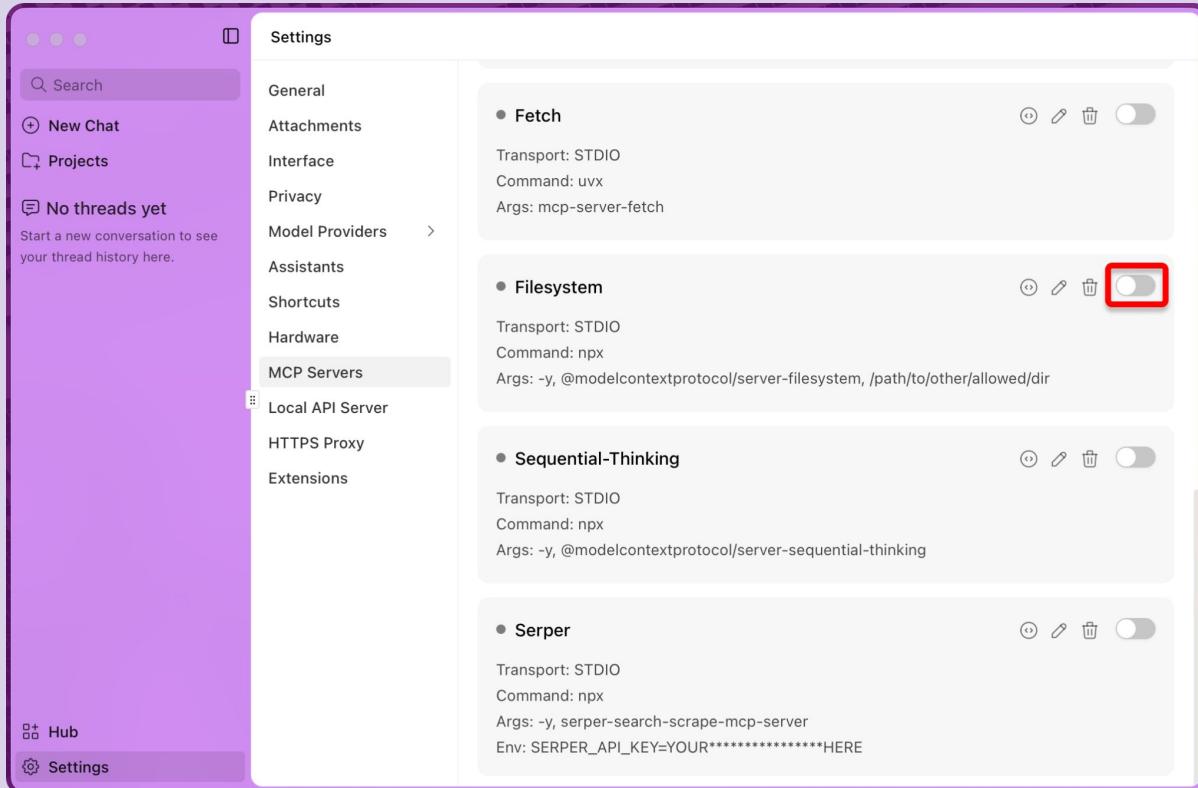
Args: -y, serpcer-search-scrape-mcp-server

Env: SERPWER\_API\_KEY=YOUR\*\*\*\*\*HERE

# Enter Allowed Filepath

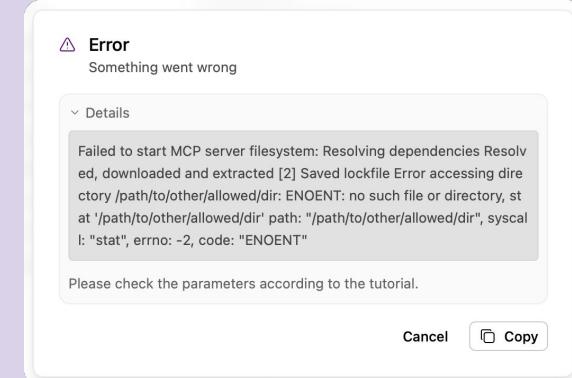


# Enable Filesystem MCP



The screenshot shows the Agentic AI Settings interface. On the left, there's a sidebar with a search bar, 'New Chat' and 'Projects' buttons, and a message 'No threads yet'. The main area is titled 'Settings' and has a 'General' tab selected. Under 'MCP Servers', there are four entries: 'Fetch', 'Filesystem' (which has a red box around its toggle switch), 'Sequential-Thinking', and 'Serper'. Each entry shows its transport (STDIO), command (uvx, npx, npx, and serper-search-scrape-mcp-server respectively), and arguments (mcp-server-fetch, -y, @modelcontextprotocol/server-filesystem, /path/to/other/allowed/dir, -y, @modelcontextprotocol/server-sequential-thinking, and -y, serper-search-scrape-mcp-server). The 'Filesystem' entry's toggle switch is highlighted with a red box.

Invalid path  
causes error



An error dialog box is displayed. It shows an 'Error' icon and the message 'Something went wrong'. Under 'Details', it says: 'Failed to start MCP server filesystem: Resolving dependencies Resolved, downloaded and extracted [2] Saved lockfile Error accessing directory /path/to/other/allowed/dir: ENOENT: no such file or directory, stat '/path/to/other/allowed/dir' path: "/path/to/other/allowed/dir", syscall: "stat", errno: -2, code: "ENOENT"'. At the bottom, it says 'Please check the parameters according to the tutorial.' with 'Cancel' and 'Copy' buttons.



# Edit MCP

Server filesystem activated successfully

Jan

Search

+ New Chat

Projects

No threads yet

Start a new conversation to see your thread history here.

Hi, how are you?

How can I help you today?

Ask me anything...

Qwen3-VL-8B-Thinking-1M-IQ4\_XS

Hub

Settings

The screenshot shows the Agentic AI MCP interface. At the top, a message indicates "Server filesystem activated successfully". Below this, the user "Jan" is listed with a gear icon. The sidebar on the left includes "Search", "+ New Chat", and "Projects". Under "Projects", it says "No threads yet" with a note to start a new conversation. The main area features a large text box with "Hi, how are you?" and "How can I help you today?". Below this is a "Ask me anything..." input field. A message from "Qwen3-VL-8B-Thinking-1M-IQ4\_XS" is shown. At the bottom of the interface, there are "Hub" and "Settings" buttons. A red box highlights the "Settings" button. The entire interface is set against a purple background with a white header and sidebar.



```
# Sprint 24 Planning & Cross-Team Sync - Meeting Minutes

**Date:** January 12, 2026
**Time:** 10:00-11:00 AM
**Facilitator:** Emma Rodriguez (Engineering Manager)

## Attendees
- Emma Rodriguez - Engineering Manager
- Alex Chen - Frontend Lead
- Priya Patel - Backend Lead
- Jordan Lee - Data Engineering
- Sofia Martinez - Data Science
- Michael Brown - UI/UX Designer
- Rachel Kim - QA Lead
- Tom Wilson - DevOps

## Agenda
1. Sprint 24 goals
2. Feature progress updates
3. Data pipeline and analytics updates
4. UI/UX review feedback
5. Risks and blockers

## Discussion Notes

### 1. Sprint 24 Goals (All Teams)
The main goal for Sprint 24 is to deliver the first beta of the user activity dashboard logic. Teams are expected to coordinate closely due to shared dependencies.

### 2. Frontend & UI/UX Updates
The frontend team has completed the initial React components for the dashboard layout. The UI/UX design currently assumes both daily and weekly aggregation views, and

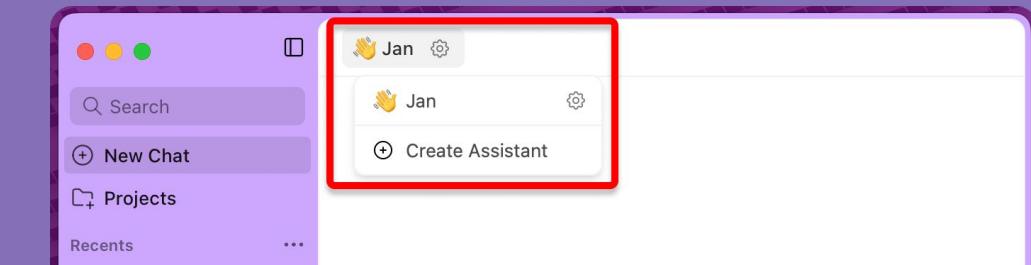
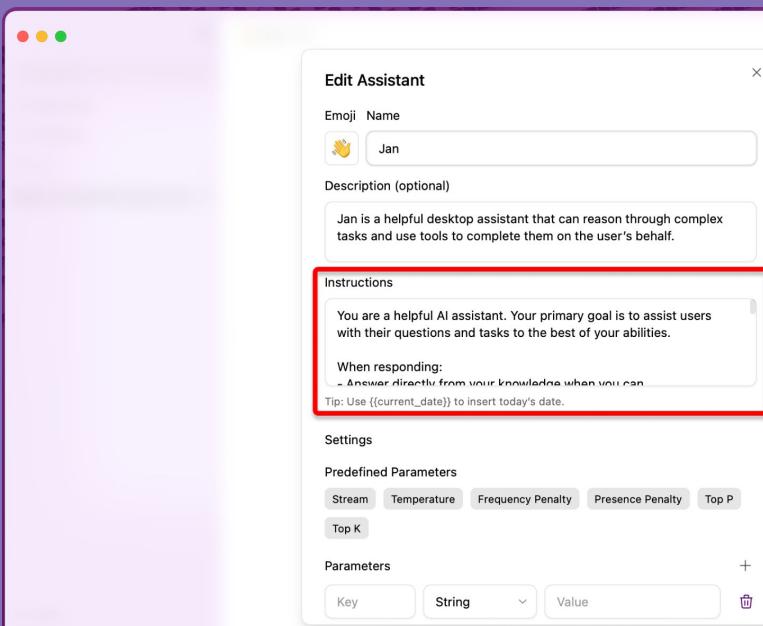
### 3. Backend & Data Pipeline Updates
Backend APIs for user activity metrics are mostly implemented but depend on changes. The data engineering team is refactoring ETL jobs to support incremental updates. Some metric definitions, such as "active user," need alignment between data science and product teams.

### 4. Analytics, Experiments, and Logging
Plans were discussed to add additional event logging to support future A/B testing. The data warehouse schema will be updated to store raw events alongside aggregated data for faster analysis.
```

# Summarise Meeting

## meeting\_mintues.md

# Customise JanAI



# MCP Servers Everywhere



Make  
your own!

Chrome Dev Tools	GitHub	AWS	Reference
Inspect DOM, CSS, and console logs  Access network requests and performance traces  Debug web pages programmatically through an agent	Browse repos, issues, PRs, discussions  Automate repository tasks and summaries  Extract code context and change history for reasoning	Query cloud resources (S3, EC2, etc.)  Manage infrastructure and retrieve logs  Enable cloud-aware agent workflows	Official MCP reference implementations for tools like Git, Memory, Fetch, and Time  <u><a href="https://github.com/modelcontext/protocol/servers">https://github.com/modelcontext/protocol/servers</a></u>

Presented By Sophie A / 2026-01-21 / Everything Open 2026

QR CODE

Thank you!

