Discussion about this video

Sean Goller

3d

I used Discord, n8n, Gemini and Postgres. I live in discord, I was trying to avoid external services and I wanted to learn about self hosting n8n anyway so that was easy, Gemini because I'm working on a different app that was using the Gemini api and flash 2.5 lite is extremely cheap, and Postgres because it's what I use in a lot of projects. I didn't even think about the community aspect at the time, but I grasped the principles of the system, and I fed the document to Claude, told it what I wanted to use instead, and we worked through it together. I learned a lot about discord bots, discord's message and threading model and n8n. This was a good prototype, but I'm probably going to redesign it from the ground up using the BMAD method to gather requirements and think about how I work and what makes sense.

Like (4)

Reply (1)

Share

Phil L

3d

Love BMAD - working to implement this at scale across my org, but Nate's discussion on principles vs frameworks has made me rethink my approach.

Like (1)

Reply

Share

Pathfinder

3d

I have my second brain setup, but my journey was the interesting part. I am using Slack, Notion, Zapier, Claude stack. However I have never used Notion or Zapier so I got frustrated and found the guide hard to follow. It wasn't until I realized that you provided the pattern to build something and I wanted step by step instructions on how use this software stack. I found success when I instead provided the instructions from the guide directly to a prompt and let it do the work for me. I still don't understand Zapier but I got the damn thing working. My V2 build will replace zapier with a node app that I can run on my own server with no additional cost. The beautiful thing is I learned how to provide this build pattern to an AI prompt for the solution, and not how to use Zapier.

Like (3)

Reply

Share

Andy Doucet

3d

I built mine with a custom Node.js web app. I set it up to use multiple models depending on the prompt. For news and whatnot, I use in Web Search with Gemini and Google via Vertex. For technical stuff, I use Sonnet 4.5. For advanced shit, I use Opus 4.5. For social media, I use Grok and XAI, and it's all configurable through the dashboard.

I'm actually building this app for a client because we can't find one that suits his needs, so it's actually very specific to how him and I think. But I think it's pretty cool that I was able to get a job right away by applying these lessons. It took me a while to put it together, reiterate, reiterate, reiterate.

I started with YAML files, then I moved to PostGres database, and then I eventually switched over to Convex. I built a lot of AI apps in the past, but I'm maintaining them now as opposed to building new apps, so this was a fun little project to do from start.

Everything's built 100% with AI. I actually launched it here: synapsepilot.app

I just subscribed, so it's pretty cool reading through everybody's implementations and how they're doing it.

Like (2)

Reply

Share

Ivo

4d

I used Slack, Granola, and Zapier to get my personal second brain working, and I just love it. Creating manual triggers for the digest and daily objectives was super easy. Loving how deep and wide this project has gone with respect to skill level and use case! It's pretty awesome. Thanks, Nate 🥳

Like (2)

Reply

Share

Cullen Steele

4d

I built mine with Telegram, n8n, Notion and Claude (and Claude was definitely my right-hand along the way) Love the top-down, principle first instruction. Thank you and keep it up!

Like (2)

Reply

Share

Benjamin Deutschle

2d

For me personally I used Siri Shortcuts, gpt, and notion. After I started using this I saw the value really quickly. I decided to build a team version using o365's. The teams I manage now all use it through the o365 architecture added a category for their personal stuff and and stuff that should be shared with the team, so everyday all the team members get a digest of their own personal stuff and also things from other team members. I also extended the board of directors idea into this where I have agents with prompts I've written looking at their second brain data and coaching them with things as well as sending me the coaching moments right away. Adoption has been a problem but my information security team took to it right away and I'm beyond amazed at how much efficiency has been added to workflow in ways I didn't expect. Sure I hoped for better communication but I've seen unexpected power. For example one team member got coaching from one of the board members telling them that team member x came across theirs and this is how they handled it.

Like (1)

Reply (1)

Share

Benjamin Deutschle

2d

Right now I'm exploring how I might be able to use this for preserving institutional knowledge and support succession planning as our staff distribution is weighted toward those near retirement. Teaching them unlearning what they know and embrace the change AI brings is challenging but when I show them how this helps their day to day, indirectly I preserve their institutional knowledge.

Like (1)

Reply (1)

Share

Jeffrey A. Wolfe

1d

Brilliant! I've been working on the Personal Knowledge Management System I need so I can access "What I know" more effectively. Seems the institutional knowledge and succession planning 'problem' has been a persistent business problem for decades.

Like

Reply

Share

Race

2d

I used a different approach centered around data sovereignty and lowest overhead. Downloaded 2 of the most efficient llms locally onto my machine, so it operates out of the terminal. I installed langgraph as the orchestration layer which also runs in the terminal and is open-sourced. Also built a custom widget as a capture point for my desktop. Using local storage for “temporary” files and obsidian as longterm storage and syncability with obsidian sync ($4 / month). To add my mobile devices to the system i dl’d obsidian on my phone and am using Obsidian as the capture point AND the storage through the phone. When obsidian sync changes the obsidian vault on my desktop and the system is running, it detects the trigger from the obsidian note, goes through the whole process and then organizes it into a separate folder back into obsidian. This was the most frictionless setup i could come up with to fit my criteria. I can also program the action button on the iphone or similar button on an andriod to bring me straight to creating a new note, hence the optimized frictionless capture point. Ty for the idea... absolutely beautiful.

Like (1)

Reply

Share

Otman Mechbal (El Capitano)

Otman Mechbal (El Capitano)3d

Really good to see us back at the roots of real human collaboration. Many people said that when ChatGPT or AI arrived, online spaces like forums, IRC, Discord, Slack, Reddit, Stack Overflow, or X threads would die, but they were wrong. Instead, it feels like we are building in the open again. 😊

Producing content and artifacts is becoming cheaper and cheaper, and trying to impose structure on things that may soon be obsolete or not adapted to our personal/business needs is like swimming in an ocean of constant noise without keeping enough space to breathe, learn, experiment, and build interesting things. What truly matters is documenting principles, decision logic, and failures with full context and nuance. That's what prevents wasted time and allows others, humans or AI, to pick up the work later. This is the kind of knowledge that actually compounds.

Like (1)

Reply (1)

Share

Otman Mechbal (El Capitano)

Otman Mechbal (El Capitano)3d

I'm sharing a structure for tools switching when I vibe code with CLI tools for instance but could have other applications for context sharing with any AI:

• AI_HANDOFF.md

The current snapshot, what is happening right now, what is next, what is blocked.

• PRD.md

What the product is supposed to do, scope boundaries, acceptance criteria.

• TASKS.md

The ordered checklist, this is the "truth list" for what to do next.

• ARCHITECTURE.md

A map of the system, where code lives, where new pieces should plug in.

• DECISIONS.md

Why things are designed that way, prevents the next tool from undoing a choice.

• RUNBOOK.md

How to run, test, verify, setup, common issues, commands.

• PROMPT_RESUME.md

The copy paste message to any tool, it tells the new tool what to read, what to run, and what not to do.

• SESSION_LOG.md

History, what happened, what commands ran, what changed.

• Makefile.md

One command interface, make dev, make test, make verify, avoids “works on my machine” drift.

I added this new markdown file by applying this:

• Principles-Based: Session state externalized to SESSION_LOG.md and AI_HANDOFF.md

I wanted to add a dedicated `GUARDRAILS.md` but it can be redundant if you are building alone. However, you should add a Safety & Resource Constraints section to your `PRINCIPLES.md`

Critical Addition: Include a “Destructive Action Protocol” (e.g., “Always use `git commit` before running any code modification tool”).

Why: This prevents “AI slop” where the tool makes rapid, unlogged changes that you cannot revert.

Benefit: Any tool reads the same context; seamless switching in <2 minutes

Leverage: Community member built “meta-agent” coordinating Claude, Codex, Gemini, Qwen, Copilot,... using shared context.

Like (1)

Reply

Share

Blaine Holt

Greet Your Limits4d

So cool to hear the community shout outs here. Super impressive what people achieved

Like (1)

Reply

Share

Ian

4d

I built it in VS Code -> Github Copilot -> Opus 4.5 because that's the only thing I can use at work :-)

Like (1)

Reply

Share

Mario Guzzi

Mario Guzzi4d

I used chatGPT project to build it. Using Nate's recommended tools so far. I now need to spend some dollars to allow GPT API connect...I was trying to build at a no extra cost beyond GPT Plus.... work in progress.

Like (1)

Reply

Share

Gravityjump

Gravityjump's Substack14h

The non-technical guy should be my new handle. I built framework and structures and processes before I even came to stack This after working with you for a number of months, this was a test of all the knowledge that you shared has been stored in my brain so thank you lol what's next for us? Non-technical people

Like

Reply

Share

Andy Wilkinson

2d

The insights I gain from Nate and this community are astounding. I got myself tripped up mentally when trying to figure out how to build this tailored to my needs.

If I have a pc desktop that I try a wok off of for life and also game dev, how do I start there with something like Claude code and obsidian. How do I organize things? PARA? Nate's methods? If I wan to add a MacBook in the future for when I'm in another room or on the go how do I work with the system when or comes to obsidian? How do I add mobile into it so I can message my AI "guy in the chair" to track and organize notes to inform projects to inform tasks to inform meeting prep, etc.?

Anyone with similar setups have thoughts? The claudesidian crowd, desktop & laptop & mobile coordination?

Like

Reply

Share

Devan B

Devan B2d

I've worked by way through this using Slack, Notion, Zapier, Openai. I've gotten stuck at the Daily Digest (Part 6.3 Query Active Project) when trying to access the content in the Notion database, since Zapier's "Find Data Source Items" functionality only seems to pull in text and data fields, it can't access Notion "pick list" properties, which are what I used to setup 'status', 'tags', etc. Has anyone figured this out?

Like

Reply

Share

Marcus Mobley

3d

I got mine to work last night after working on it for two days. I started on Zapier, got steps 1-4 working, but then realized that Zapier was going to be too expensive for me to use ongoing. So I decided to pivot to Make. This was a much messier configuration experience than Zapier because they're not partners with Notion. I got really frustrated so I decided to scrap both and went with Azure functions; mostly because it's cheap and I've used them a lot at former jobs.

So my stack is:

Slack, Grok 4.1 fast reasoning, Azure Functions (linux typescript), Notion, GitHub actions, Cursor (Sonnet 4.5) to build it.

My original approach was to try to tackle this without any sort of guide or hints on how the workflow should be laid out; but I used all the same tools Nate mentioned. I got the initial categorization working after lots of frustrating back and forth with their AI chat; being new to Zapier, I relied on this heavily at first. I quickly started discovering the shortcomings of Zapier as Nate mentioned and tried to think of ways to make this thing easier to build. To be honest, I couldn't think of a way to get the fix functionality to work, so after trying a few things I subscribed to this substack and found the guide.

It turned out my approach wasn't too far off from his, but his was still unavoidably ugly and triggered my software engineering sensibilities that I couldn't join paths back together, or easily reuse loops.

So, as I mentioned above, I tried to use Make due to cost concerns with Zapier and that was a boondoggle after an hour of trying to get Slack and Notion configured. And so I decided to drop these SaaS middlemen and went with something I knew: Azure Functions.

That wasn't as smooth as I had hoped it would be due to Claude Sonnet using the wrong deploy command burning 3 hours trying to get it running, but eventually it all worked and it's way better than Zapier. It's much faster: it reliably replies in < 5 seconds and it's a lot cheaper. Almost $0 using consumption plan function app.

What's great about it is that it's all in typescript and so it's super easy to tweak or expand using Cursor vs trying to fumble my way through Zapier's UI. Also changes are deployed within 2 min using Github actions.

Here's my stack in Azure:

Azure function app

slack-events - http trigger

capture-worker - queue trigger

fix-worker - queue trigger

daily-digest-timer - timer

weekly-review-timer - timer

weekly-review-worker - queue trigger

storage account

queues:

capture-jobs

digest-jobs

fix-jobs

review-jobs

azure app config - stores the various LLM prompts for each update

app insights for debugging

Anyhow, thanks for the inspiration Nate. It pushed me to learn some new tools and overall pretty enjoyable.

Like

Reply

Share

AI Lifestyle Lawyer

3d

Yaaaaassss ... I think I might be the only female mid-life mama posting on Nate's stack (lemme know if not) 🤣 I read and watched Part I - for YEARS I've been waiting / wishing / hoping the tech would arrive that would allow real building of a 2nd brain (no shade on Tiago but it needed to be active not static). And I tried really hard to love Thomas Frank's templates in Notion for a 2nd brain but alas ... I firmly believe if you don't build it yourself, you won't use it long term. Building one of these with the right tools and principles for myself, and then hopefully showing other working mamas.

Like

Reply (1)

Share

Irene Garvin

2d

Hey fellow mid-life mama!! I agree if you don’t build it perhaps you won’t use it long term but I am so curious how we will spend time on other implementations.

Like (1)

Reply (1)

Share

AI Lifestyle Lawyer

1d

Hi Irene!! So glad to see another midlife mama on here!! 😎🤩

Like (1)

Reply

Share

Kevin Rank

3d

My ultimate goal, is to combine this with Karakeep. I want to have a more universal bookmark repository with maybe some notes/context. I want that to exist along side my second brain to integrate the bookmarks into notes, ideas, links to things I want to save. Basically, Karakeep would be context sparse bookmarks, where the second brain would be more context rich. I also want Karakeep to be more shareable.

Like

Reply

Share

Peter Meng

Letters From Your Future3d

This was a game (life?) changer project. When I watched the first video I was psyched. Took my dog for a walk to think about it and realized I could build the whole app in my favorite platform Base44 with Claude 4.5 Opus integrated. About 10 hours later I have an app my friends are saying I should take to market. Check it out - https://my3rdhalf.base44.app.

BUT - this current video is even more mind blowing as you've provided (as usual!) a clear framework for extending this exponentially. Keep it coming. Amazing stuff! Wild times!!

Like

Reply (1)

Share

Ulrich

3d

Brian, just tried out your app. Awesome! A tutorial would be great (would pay for it). What do you think?

Best, Ulrich

Like (1)

Reply (1)

Share

Peter Meng

Letters From Your Future2d

Thanks Ulrich. I'll add a 1 minute video soon into the app that takes the user through the process.

Just figured out how to wrap the web app for mobile. Waiting on my Apple and Android Developer accounts to be approved (Why does it take a week????) so I can post to the stores. Until then enjoy the app and suggest any changes and report bugs. You've earned a fee copy (of a $5.00 app)! ;-)

Like

Reply