



● A PLAYBOOK BY DAVID, BUILDING CONTEXTBOLT

How one person ships a SaaS *every day*.

My Claude Code + MCP + Obsidian workflow for writing blog posts, running SEO, drafting tweets, and shipping every day. In 10 hours a week.

● THE SETUP

I work 10 hours a week. Still shipping *every day*.

I have a full-time job. I build ContextBolt in the evenings and at weekends.
No team. No funding. No moat except speed.

The only way the maths works is if I automate 80% of the work. This is how.

2 WEEKS OF OUTPUT

25 Blog posts
written & published

40 pSEO pages
shipped & indexed

5 Daily automations
running on cron

60+ Deploys
to production

Most solo founders get stuck in the *wrong loop*.

They spend their limited hours building features, fixing bugs, and wrestling with marketing tools. The work expands to fill the time. Shipping slows. Momentum dies.

01

Build

Every hour goes into code. The product gets longer, not better.

02

Fix

Bugs eat the rest. Every ship unlocks two more problems.

03

Market later

Distribution gets pushed to "after launch." After launch never comes.

Five layers. *One keyboard.*

Every layer does one job. Together they turn a 10-hour founder into a 40-hour team. I built it in three weekends.

01 **THE VAULT**

Obsidian. Single source of truth. Every decision lives in markdown.

02 **SKILLS**

7 custom Claude Code skills. Named commands for every workflow.

03 **SCHEDULED**

Claude runs while I sleep. Four daily tasks, one weekly deep-dive.

04 **MCP SERVERS**

8 live data + tool plugs. Claude queries GA4, GSC, Brevo, Canva, Chrome.

05 **DEPLOY**

Commit to live in 48 seconds. GitHub Actions + Cloudflare Pages.

Obsidian is *the brain*.

Every decision, metric, commit, strategy doc, skill, and memory lives in plain markdown. Claude reads the vault before every session. Rewrites it after.

“If it isn't in the vault, it didn't happen.”

The only rule. Claude enforces it on every shutdown by rewriting the relevant files so they stay coherent. Not a changelog. A working document.

/CLAUDE_VAULT

00 - 0S/

instructions.md · Dashboard.md · Playbook.md · skills/

05 - ContextBolt/

GTM/ · Operations/ · Technical/ · reports/ · code/

analytics/

current.md · YYYY-MM-DD.md (auto-generated daily)

memory/

MEMORY.md + 35 persistent facts across sessions

.claude/

skills/ · scheduled-tasks/ · settings.json

7 skills I call *by name*.

Skills are re-usable playbooks. Type `/analytics` and Claude pulls GA4 + GSC. Type `/weekly-review` and Claude sets Monday priorities. Every complex workflow becomes one command.

`/vault-session`

Loads instructions, Dashboard, Memory, Playbook. Runs at every start and shutdown.

`/analytics`

Pulls GA4 + GSC, writes daily report to `/analytics`, updates Dashboard, flags quick wins.

`/weekly-review`

Monday priorities, Wednesday check-in, Friday wrap-up. Escalates stale blockers.

`/social-posts`

Reads strategy, blog, analytics. Drafts 5 posts. Runs anti-slop + One Thing tests.

`/reply-engine`

Drafts 15 to 20 X replies per day across tiered target accounts. Golden-rule tested.

`/contextbolt-copy`

Writes landing pages, emails, product copy in Harry Dry voice. No em dashes. Ever.

`/audit`

Technical quality sweep. Accessibility, performance, theming, anti-patterns. PO-P3 scored with a full fix plan.



Claude works *while I sleep*.

Four tasks run every day on a cron. One deeper pass runs every Monday. The output always lands in the vault as markdown I can skim on my phone on the train.

07:00

Daily X post drafts

5 varied posts. 2 expertise, 2 authority, 1 lead-gen. Passed through anti-slop checks.

08:00

Daily site pulse

Health checks, GSC quick wins, indexing status. One markdown report. Silent on pass.

10:00

Daily blog post

Writes, generates SVG illustration, commits, pushes. Live in under 2 minutes.

18:00

X reply queue

Searches target accounts, drafts 15 to 20 replies, lands in /X Replies/queue-YYYY-MM-DD.md.

MON

Weekly SEO deep-dive

Performance, content freshness, AEO/GEO scoring. TLDR with priority actions on top.



Eight plugs. *One brain.*

MCP (Model Context Protocol) is how Claude talks to my actual tools. Not "ask the user to paste in GA4 data." Claude queries Search Console live. Drafts a Brevo campaign. Opens Chrome. Creates a Canva design. All mid-conversation.

contextbolt-ga4

Live GA4 data: users, events, page views, funnels.

contextbolt-gsc

Search Console queries, CTR, positions, quick wins.

brevo

Contact lists, campaigns, transactional email in one API.

canva

Generate, edit, export designs programmatically.

claude-in-chrome

Drive a real browser. Click, fill, read, screenshot.

claude-preview

Local dev server with logs, snapshots, network inspect.

mcp-registry

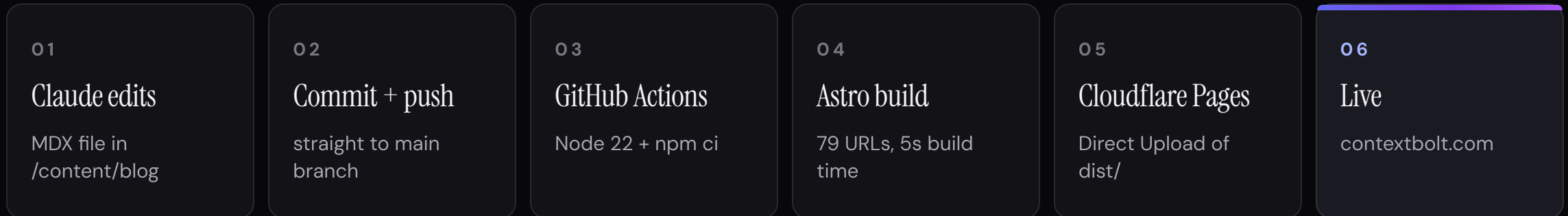
Search and discover new MCP connectors for any tool.

scheduled-tasks

Cron-style automated Claude sessions with full context.

Commit to live in *48 seconds*.

Claude writes an MDX file, runs `git commit`, pushes to GitHub. GitHub Actions does the rest. No wrangler on my laptop. No API tokens locally. No "works on my machine."



Average deploy time: *48 seconds* · 60+ deploys since the migration on 7 April.

A CMS for *manual overrides*.

Claude handles 90% of content. For the other 10% I open `/admin` on my own site. Decap CMS reads straight from my MDX files. Edit a headline, hit save, it commits and deploys. No database, no lock-in, no extra service.

No database

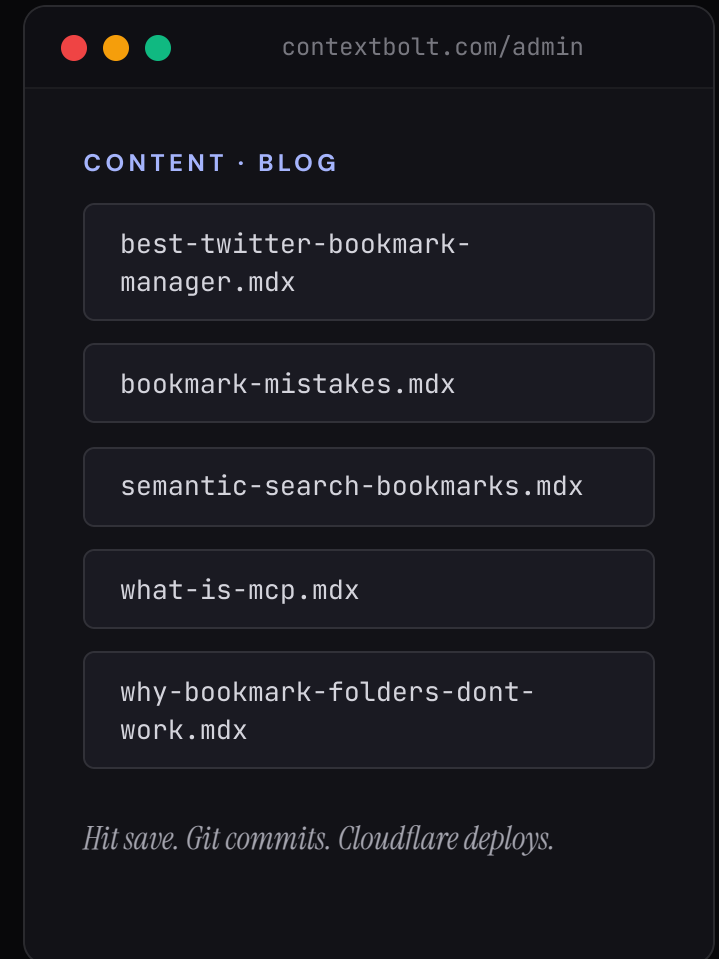
Every edit is a git commit. Every rollback is a git revert. Content lives on disk.

No paid tier

Decap CMS is open source. Runs off a single static HTML file at `/admin`.

No lock-in

The content is plain MDX. I can ditch the CMS tomorrow and nothing changes.



This is what *Tuesday* actually looks like.

I don't touch a keyboard before 6pm. Every output below was created or prepared by Claude. I read, edit, approve, post. Total active time: under 60 minutes.

- 07:00 **CLAUDE** Drops 5 X post drafts into the vault. 2 expertise, 2 authority, 1 lead-gen.
- 08:00 **CLAUDE** Writes the site pulse report. Flags 2 quick-win pages for meta rewrites.
- 10:00 **CLAUDE** Ships a new blog post. Live in 48 seconds. Auto-submitted to Google.
- 12:30 **ME** Skim the post drafts over lunch. Pick one to publish. 4 minutes.
- 18:00 **CLAUDE** Queues 18 X replies from target accounts. Each under 24 hours old.
- 19:30 **ME** Post 12 good replies. Push the chosen blog post. 22 minutes total.
- 22:00 **ME** Ask Claude to update the Dashboard and log the day. 1 minute.

How a blog post *writes itself*.

- 1 Claude reads the SEO SOP, blog strategy, and the last 10 posts. 10s
- 2 Checks Search Console for under-served keywords and gaps in my coverage. 4s
- 3 Drafts title, meta description, structure, FAQ, and a 1,500 to 2,000 word body. 45s
- 4 Generates a custom SVG illustration for the blog card and saves it to the asset file. 6s
- 5 Validates frontmatter against the Zod schema. Flags anything off before the commit. 2s
- 6 `git commit -m "new post" && git push`. GitHub Actions takes over. 48s
- 7 Post is live. Sitemap updated. RSS refreshed. Done before I finish my coffee. -

Total active time from me: *0 minutes*. I read the post on my phone later and tweak the headline if it lands wrong.

15 quality replies in *10 minutes*.

Every good reply puts you in front of someone else's audience. Problem: quality replies take ages. Fix: Claude drafts them against a golden rule and a tiered target list. I pick. I post.

- 1 Reads the reply-engine skill and target accounts file.
- 2 Searches X for posts under 24 hours old from those accounts.
- 3 Drafts 15 to 20 replies across 7 templates.
- 4 Runs each through the "useful to a stranger" golden rule.
- 5 Outputs a queue file in the vault for me to skim and post.

THE TIER SYSTEM

Tier 1 **40% OF REPLIES**

10K+ followers · Reach + algorithmic boost

Tier 2 **40% OF REPLIES**

1K to 10K followers · Higher follow-back rate

Tier 3 **20% OF REPLIES**

Under 1K followers · Goodwill. They remember.

2 weeks. *One founder.*

Every number below is a real output from this stack since I rebuilt the site on Astro in early April.

25

Blog posts
live on the site

40

pSEO pages
(comparisons, glossary, use
cases)

7

Custom Claude Code
skills I call by name

8

MCP servers wired
into my workflow

5

Daily or weekly
automated tasks

48_s

Commit-to-live
deploy time

1,502

Google impressions
in the last 7 days

10_{hrs}

Active time from me
per week



Three principles *everything rests on.*

Copy the stack if you want. These three ideas matter more than any tool.

01

The vault is the brain.

Everything Claude needs to make a decision lives in markdown. Nothing important lives in the conversation. Sessions die. Markdown does not.

02

Skills are the hands.

Every recurring workflow becomes a named skill. "Do analytics." "Review this week." "Draft replies." The command is the product.

03

MCP is the nervous system.

Give Claude direct plugs into the real tools you use. It stops asking you to paste data. It queries the source.



Build this in *one weekend*.

The whole stack above. Four steps. No complicated tooling. No paid tier needed to begin.

01

Spin up a vault

Obsidian + a /CLAUDE_VAULT folder. Add a Dashboard.md and an instructions.md. That's it.

02

Install Claude Code

Write your first skill. Mine was /vault-session. One markdown file per skill. Start small.

03

Add one MCP server

Start with Google Search Console or your own data source. Immediate, visible feedback.

04

Wire a deploy

Astro + GitHub Actions + Cloudflare Pages. Free tier covers it. 48-second ships from day one.



ContextBolt

AI-powered recall for saved content.

Built by one person. · Follow for more.

contextbolt.com