Internal Tools Workspace

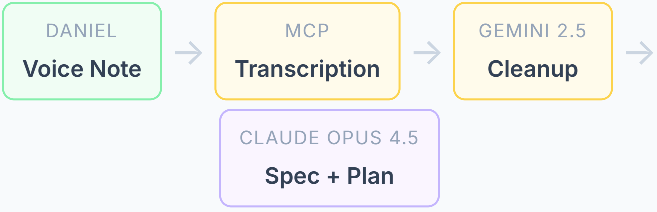
From Voice Note to Implementation Plan

Daniel Rosehill

24 December 2025

About This Document

This document demonstrates a voice-to-implementation workflow. Starting from a long-form voice note, the process uses AI tools to transcribe, format, and generate actionable specifications and recommendations.



Tools: Gemini Transcription MCP, Gemini 2.5 Flash, Claude Code (Opus 4.5)

Step 1: Input DANIEL

The Idea (Cleaned-Up Voice Note)

Transcribed using Gemini Transcription MCP and processed with Gemini 2.5 Flash. Lightly edited for readability. Original verbatim transcript in Appendix.

> The purpose of this voice note is to provide context about a workflow I'd really like to roll out properly in the coming year, 2026. I'll describe what I'm trying to achieve and what I haven't figured out yet.
>
> With the rapid rise of AI, there are so many different workflows that are potentially helpful. I'll give an example of one that I just thought about. I was working on my whiteboard—one of those days where I feel I need to jot down what I need to get done. Mostly very boring stuff: filing tax returns, NFC tags I need to set up, cleaning.
>
> So here's a workflow I thought about. I've done a few very effective ones with Nano Banana that have basically involved cleaning up a whiteboard. The use case was sketches for product ideas. But what I was thinking for this workflow—a catchy name might be something like "A Whiteboard a Day" or "My Daily Whiteboard."
>
> I can clean up the whiteboard as I've done before, and that works really well. Not only that, it would create a Google Doc with my daily whiteboard. The photos go up to Gemini, which cleans them up firstly, and secondly extracts the data—basically a vision workflow kind of like OCR. Maybe do even a little bit of organization because whiteboards show connections. It can infer prioritization, sort the tasks. Today I have a mixture of business and personal tasks here. If it gave me a list like "24th of December: Personal tasks, bullet points; Business tasks, bullet points" and generated a Google Doc in a folder—that would be a great workflow.
>
> ### The Problem of Workflow Deployment
>
> Over the course of last year, I found quite a few workflows like this and a few have been very useful. Another one I want to roll out is an end-of-day log. It's a practice I really want to start doing because I like the idea of Kaizen—daily incremental improvement. I tend to sometimes say "I really want to finish this, I'm going to burn the midnight oil"—and it's a habit I want to get out of.
>
> The driver for me is usually that if I don't do it this way, I won't remember where I left off. So I thought: record an end-of-day diary when I'm about to step away from the workstation—"this is where we're up to, these are the blockers, these are the projects"—generate a Google Doc from that, and in the morning I can look at my retrospective and pick up.
>
> These are two workflow ideas intended as little assistants for my daily life and productivity. But neither is so monumental that I want it to be its own website or app.

What I'm Getting At

There's a lot of technology that's come to market in the last year. I love the Google website builder they rolled out in AI Studio. It's easier than ever with vibe coding to say "I want this, let's make it." The only question becomes deployment.

If I create a bunch of different utilities, the commonality as internal tools is probably authentication—because it's using Gemini API credits. I don't want anyone on the internet using my whiteboard cleaner and getting a bill for the whole internet using Gemini on my dime.

Authentication is a requirement to protect the API. The other thing is cross-platform. My whiteboard app makes more sense to use from my phone where I can take photos, upload, hit send. The backend architecture is pretty straightforward. It's purely a problem of frontend.

I frequently find myself asking: we can build all these utilities, they need to be authenticated, but I want a workspace. An app that is my AI assistant productivity center. Not a conversational app—bundling together standalone utilities. I can go into "Daniel's AI Workspace" or my company's "Internal Productivity Zone."

I have a VPS, Vercel, a home server. It's not that I'm lacking for places to put stuff. I'm trying to figure out the most sensible way. These are piecemeal apps—I might come up with eight over the course of the year. I want something I can add to. I work on these as modules—spend an hour getting the whiteboard thing working, then integrate it into my workspace. A wrapper I can add elements to.

The Need for an AI Glue Framework

What I'm looking for is not the flashy tool or framework—it's the glue. A framework whose purpose is to bundle together different AI tools. Not something like Divi where you're tied into using their system. I'd like to build these things custom. When you use Google AI Studio, you tend to get React or Vue frameworks. That's fine.

I just want a private environment where I can drop in my own modules. Most of the challenge is actually UI/UX. If I had a workspace on my phone with these different modules—the end-of-day thing, the whiteboard thing—all you need is a menu. A sidebar. You just need that navigation core to wrap around your elements, common authentication, and support for dropping in and deploying additional modules.

That's pretty much the spec I have in mind.

Step 2: Specification CLAUDE OPUS 4.5

Formalized Requirements

Problem Statement

Multiple standalone AI-powered utilities have been (or will be) developed independently. These tools:

- Are too small to warrant individual apps/websites
- Share common needs: auth, API key management, cross-platform access
- Currently lack a unified deployment and access strategy

What's needed: A framework/shell that bundles these existing utilities into a cohesive authenticated workspace.

Core Requirements

1. Authentication Layer

- Single login protects all modules
- Prevents unauthorized API credit usage
- Method: Google OAuth (simple, familiar)

2. Module Shell / Navigation

- Desktop: Persistent sidebar
- Mobile (Android): Hamburger menu
- Modules render within the shell's content area
- Smooth transitions between modules

3. Shared Environment Variables

- `GEMINI_API_KEY`
- `OPENROUTER_API_KEY`
- `REPLICATE_API_KEY`
- Google API credentials

Modules inherit these without individual configuration.

4. Cross-Platform Support

- **Desktop:** Browser-based

- **Mobile:** Android phone via browser (responsive)
- **PWA:** Installable on Android home screen

5. Module Integration Pattern

- Self-contained module directories
- Standard interface for mounting into the shell
- Module isolation (one failure doesn't crash others)

Non-Requirements

- Not building the AI utilities themselves (already handled)
- Not a no-code/low-code platform
- Not a conversational AI interface
- Not deployed locally/self-hosted

Success Criteria

1. New module integrates in <1 hour
2. One login, all modules accessible
3. Works on desktop browser and Android
4. Environment variables shared without per-module setup

Step 3: Recommendation CLAUDE OPUS 4.5

Implementation Approach

Stack

Framework	Next.js 14+ (App Router)
Auth	NextAuth.js with Google OAuth
Styling	Tailwind CSS
PWA	next-pwa
Deployment	Vercel
Domain	tools.dsrholdings.cloud

How It Works

Authentication

NextAuth with Google OAuth, restricted to your email only. Middleware protects all routes—unauthenticated users see nothing.

Shell Layout

A shared layout wraps all pages:

- Desktop: 256px sidebar with module links
- Mobile: Hamburger menu that slides out
- Content area renders the active module

Adding a Module

1. Create folder: `src/app/new-module/page.tsx`
2. Add to navigation array in Sidebar component
3. Implement the module UI
4. Push to GitHub → Vercel auto-deploys

No routing config, no auth setup, no environment variable wiring.

PWA for Android

With `next-pwa` and a manifest.json:

- User visits site in Chrome
- Prompted to "Add to Home Screen"
- App installs with icon
- Launches full-screen (no browser chrome)
- Auth session persists

File Structure

```
dsr-tools/
  src/
    app/
      api/auth/[...nextauth]/route.ts
      whiteboard/page.tsx
      eod-log/page.tsx
      layout.tsx
      page.tsx
    components/
      Sidebar.tsx
      MobileNav.tsx
      SessionProvider.tsx
    middleware.ts
  public/
    manifest.json
    icons/
  next.config.js
  package.json
```

Deployment

1. Push to GitHub
2. Import to Vercel
3. Set environment variables once
4. Add custom domain
5. DNS: `CNAME tools cname.vercel-dns.com`

Result

- Single authenticated workspace at `tools.dsrholdings.cloud`
- Google login restricted to your email
- Responsive sidebar (desktop) / hamburger (Android)

- PWA installable on Android home screen
- Modules are just folders under `/app`
- All API keys available via `process.env`
- Git push = deployed

Appendix DANIEL

Raw Voice Transcript

This is the verbatim transcript before cleanup, showing the natural speech patterns including filler words, false starts, and corrections. Transcribed using Gemini Transcription MCP. Included to demonstrate the voice-to-document workflow.

Okay, so the purpose of this uh voice note, um, that I will transcribe as usual using uh Whisper or Gemini is to uh provide a bit of context about um a, I guess workflow, work process that I'd really like to roll out uh properly in the coming year, 2026. And I'll describe sort of what I'm trying to achieve um, and what I haven't figured out yet. So with the rapid rise of AI and all that, um, there's just so many different workflows that are potentially helpful. And I'll give an example of one that I just thought about. I was working on a whiteboard, working on my whiteboard. It's one of those days where I feel like I need to jot down what I need to get done. Mostly very boring stuff like uh filing tax returns. Um, what else do I have up there at the moment? And then a few more interesting things about uh NFC tags I need to set up. Uh looking over now. Uh cleaning of course, another another scintillating task. So I have a few things and it really does help me to kind of have those just like written out on a whiteboard even if it's just on my office. I can kind of check on it. Um, so here's the workflow that I thought about. I've done a few um very effective ones with Nano Banana, um, that have basically involved cleaning up a whiteboard. So you take a whiteboard, you know, it's kind of a bit messy and cleaning it up. The use case for that was sketches for product ideas. Um, but what I was thinking in this work in this workflow is I take I was thinking of a name, a catchy name might be something like uh a whiteboard a day or my daily whiteboard or something like that. And I can um clean up the whiteboard as I've done before and that works really well. It's a it's a it's a very effective um workflow. And but not only that, it would create a Google doc um with my daily whiteboard. So in other words, the whiteboard or whiteboards, if I might occasionally um do a few of them. I take photos on my phone. Uh this is what I this is what I want to do today or even this week for that matter. Uh, the photos go up to Gemini, which uses Nano Banana to clean them up firstly. Um, and secondly to extract the data. So basically a vision workflow kind of like OCR, I guess. Really, uh a good workflow for Gemini because it's multimodal. Um, and maybe do even a little bit of organization because whiteboards show connections. Um, it can infer prioritization, it could sort the whiteboards. Like today for example, I have a mixture of business and personal tasks here. The business ones are the bureaucracy. Uh the personal ones are the the household stuff. Um, and if even if it gave me a list like today's date, 24th of December, personal tasks, task, bullet points, business tasks, task, bullet points and generated a Google Doc in a folder. That would be a great workflow. Now, here's what I'm, here's my question. Over the course of last year, I found quite a few workflows like this. Um, and a few of them have been very, very uh useful. Another one that I kind of want to roll out uh for my own sort of just work life balance, productivity planning, all the rest of it is an end of day log. It's a practice I really want to to start doing um because I really like the idea of Kaizen or daily incremental improvement and I tend to sometimes say I really want to finish this. I'm gonna go for it at all costs. I'm gonna have another cup of coffee. I'm gonna, you know, burn the midnight oil and it's really bad. And it's an hab, it's a habit I want to get out of. And what I've realized that um the driver for me is usually if I don't if I don't um do it this way, I won't remember where I left off, what the project was, blah, blah, blah. And therefore I've thought as a sort of hack for my own uh brain would be to record an end of day diary with just when I'm about to step away from the workstation for the day and say, okay, this is where we're up to. These are the blockers, these are the projects. Um, and then generate, record that as a voice memo maybe, generate a Google Doc from that. And then in the morning, I can look at my retrospective and say, okay, I'm ready to pick up. So these are two ideas of workflows that are um intended as, I guess, little assistance for me in my daily life and my productivity. But neither is so monumental that I want it to be like its own website or its own app. So what I'm getting to is this. Um, there's a lot of technology that's come to market in the last year. Um, I love the Google website builder and they rolled it out in AI Studio. It's easier than ever especially with Vibe coding to actually say, I want this. Let's make it. Let's do it. Great. Um, and the only question that I have becomes if I create deployment basically. Um, if I create a bunch of different utilities, the, they need to be, the commonality is as they're internal tools is probably authentication. Like, and that's only because it's using Gemini API credits. So I don't want anyone on the internet to be able to use my whiteboard cleaner and uh, you know, I'll get a bill for the whole of the internet using Gemini on my dime. Um, so I have authent, authentication is a requirement just to protect the the, you know, the API and the integrations. Um, but the most elegant and the other thing is cross platform. Uh, my whiteboard, for example, that app actually probably makes a lot more sense to I'd use that from my phone where I can just take photos, upload it, hit send and send it off to the, to the queue. Um, the whiteboard app is a good example actually, I think of an N8N workflow. Like I can see it being quite fairly straightforward to, to do that. Uh, a form capturing images, send, and then send that off to Gemini, uh, and then send, you know, create the Google Doc. So, I've done a lot of that stuff. Like the back end, um, architecture is is pretty straightforward or you can get, or you can vibe code your way to it. Um, so it's purely a problem in my mind, um, of front end in that I haven't really seen a, and I don't even know, like Streamlit's another example. Streamlit and Gradio, I've done a lot of tools using those things. And I frequently find myself asking, great, we can build all these utilities, they want to be authenticated, but I need, I want a workspace. I want like an app that is my AI assistant productivity center. It's not a conversational app. It's bundling together all these standalone utilities. So I can just go into my let's call it Daniel's AI workspace or my company's called yes or holdings. It could be yes or holdings um, I don't know, internal productivity zone or these are terrible names. But that's the idea. Um, and in terms of deployment, you know, I have a VPS, I have Vercel, I have a home server. Um, it's not that I'm lacking for places to put stuff. I'm just trying to figure out what's the most sensible way. Like what I want to do because these are kind of piecemeal apps, like uh, you know, I might have come up with eight of these over the course of the year. So it's not like a gigantic, you know, spiral of different things. Um, but it's more that I want to have something that I can create a new. I tend to work on these as modules, right? So I might spend an hour today, um, getting this workflow, the whiteboard thing working. Um, and then what I'd like to do is integrate that into my workspace. So it's kind of I'd like to have that kind of wrapper um, that I can add elements as I wish. And then the other thing that makes a lot of sense to me because these are would be my own internal tools, whatever. Um, I know in Versell, for example, you can share environment variables. So the ones I'm always using are Gemini, Open Router, um, Replicate, you know, probably five or six of them for different AI things. So that makes a lot of sense to me because if I could have like this space, I could share the variables. Um, I guess what I'm looking for to be more more concrete is there's a million and one great projects on GitHub and the internet and open source. And what I'm looking for is is not the, not the um, not the flashy tool or framework. It's the it's the glue. It's a framework that's whose the purpose is to bundle together different AI tools. And that isn't something like uh let's say, I'm trying to think of an example. Divi, I think is it called Divi? Oh, the names sound similar after a while. Um, where it's kind of you are tied into using their system to build it. No, I'd like to build these things are probably best custom built. Um, when you use Google AI Studio, you tend to get kind of React or Vue frameworks. That's fine for me. Um, but I again, I just want the sort of private environment where I have the ability to drop in my own uh modules, have a, and again, most of the challenge here is actually UI UX or it just simple things like that. Like if I had a workspace on my phone, and I have these different modules, we the end of day thing I talked about, the whiteboard thing, all you need is a menu basically. Like can be a sidebar and then on the desktop it can be another also a menu. You just need that navigation core um to wrap around your elements. Um, common authentication, and support for the ability to drop in and deploy additional modules. I would that's pretty much the spec that I have in mind here.