

[Open in app](#)



The Micro-SaaS Co... · [Follow publication](#)

Member-only story

# This Reddit Thread Is a Blueprint for a \$100K Business:

How to Find Untapped SaaS Ideas by Solving “Papercut” Problems

5 min read · Oct 26, 2025



Mihai Sandu

[Follow](#)

Listen

Share

More

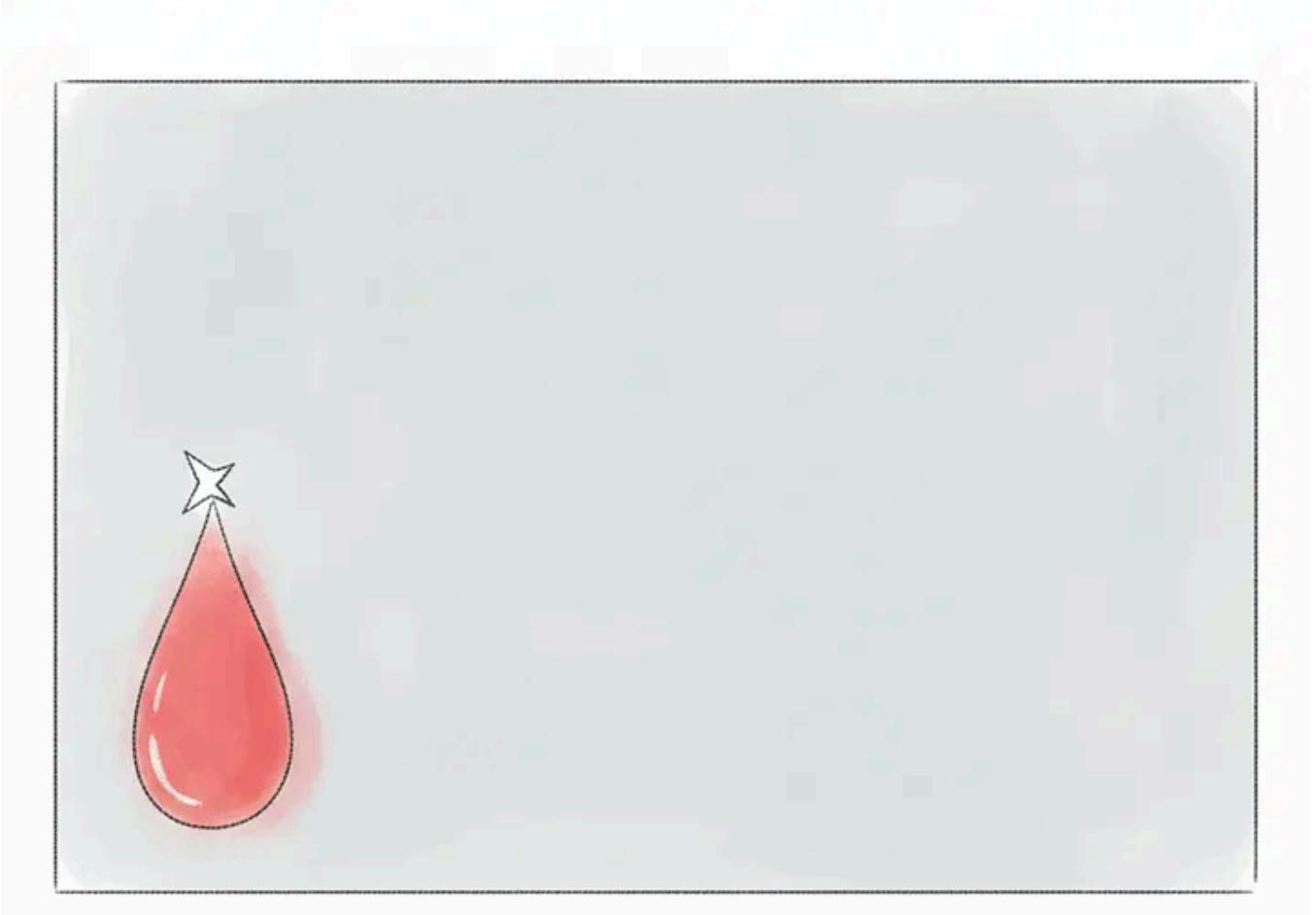


Image Generated with AI

Reddit isn't just a place for memes; it's a treasure map for business ideas. No gurus here. Just a healthy obsession with fixing things.

This week, we're venturing into the world of professional video editors.

I was scrolling through the r/editors subreddit and stumbled upon a star. The post starts with an editor celebrating 12+ years in the business and finally automating the boring stuff away.

It's a love letter to a tool called Keyboard Maestro, which lets you build little macros to automate clicks. The author talks about saving 5–15 seconds on tasks they do 20+ times a day — creating a new folder with today's date, managing their clipboard, or resizing windows.

It sounds small, but in a world where you're paid by the hour or by the project, those seconds add up to real money and sanity.

The post isn't a complaint. It's a celebration. But hidden inside that celebration is a massive opportunity.

Don't have a medium account? You can read this post on my [Substack](#).

## The Evidence Is in the Comments

The real magic was the comments. The entire community lit up, sharing the incredibly specific, sometimes complex “micro-tools” they had built for themselves.

They weren't just making folders. They were building custom software to patch holes in their professional, multi-thousand-dollar editing programs.

Check out this comment:

I really fell down the KM rabbit hole a few years ago. I was always annoyed that Avid could only do “AutoSequence” based on the Start TC but not based on an Aux TC. So I built a script that can patch a bin full of clips onto an existing sycmap based on their AuxTC.

This is fascinating. This person's super-expensive software, Avid, was missing a feature they needed. So, they just built it themselves. This isn't just about saving time; it's about adding a capability that doesn't exist. The frustration is with the limits of their primary tool.

Then there was this one:

Figured out how to get it to parse CSVs which has been very useful for anytime a producer gives me any sort of spreadsheet referring to the footage.

This is a huge pain point. An editor gets a spreadsheet from a producer with notes or timecodes. To use that info, they have to manually copy and paste it into their editing software. This person built a bridge between a boring spreadsheet and a complex video timeline. That's a powerful, high-value task.

The evidence is clear: Editors are highly technical people who are already solving their own problems with duct-taped solutions. They are willing to spend hours building a script to save minutes.

What if we could give them back those hours?

## The Solution: An Editor's Swiss Army Knife

This isn't about building a better Keyboard Maestro. That tool is amazing, but it's a blank canvas for tinkerers.

Most editors don't want to be programmers. *They want to edit.*

Let's build EditKit.

EditKit is a dead-simple app that offers a library of one-click automations for video editors.

No scripting. No complex setups. Just a clean dashboard with buttons that say things like "Clean Up My File Names" or "Create Project from CSV."

For a solo founder, this is perfect. You're not trying to build the next Adobe Premiere. You're building a small, sharp tool that solves the "papercut" problems — the tiny, annoying tasks that drive creative professionals crazy. You can build and release these automations one by one. The product gets more valuable with every new "tool" you add to the kit.

It's buildable, maintainable, and you can market it directly to the communities where people are already talking about these exact problems.

## The Actionable Playbook

Okay, enough dreaming. Here's how you could actually build this.

### The MVP (Minimum Viable Product)

We need to be ruthless. What's the absolute minimum to solve a real, money-wasting problem? Based on the Reddit thread, here are the first three tools for EditKit:

1. The CSV-to-Timeline Tool: This is the killer feature. A simple web page where you upload a CSV file, map the columns ("Clip Name," "Timecode," "Marker Text"), and it generates an XML file that editors can drag right into Premiere Pro, Final Cut, or DaVinci Resolve. This could save someone hours on a single project.
2. The Smart Project Folder Creator: A simple form where you enter a client and project name. It generates a perfectly structured set of folders (`01_PROJECT`, `02_FOOTAGE`, `03_AUDIO`, etc.) and downloads it as a ZIP file. Simple, but it saves clicks and keeps projects organized.
3. The File Name Cleaner: A page where you drag a folder of media files. The tool automatically removes weird characters (`!@#$%^&*`), replaces spaces with

underscores, and makes everything web-safe and edit-friendly. This prevents a ton of technical headaches.

That's it. Three tools. Each one solves a distinct, nagging problem mentioned in the thread.

### **The Real Challenge (It's Not the Code)**

The biggest hurdle isn't building these tools. A decent developer could build the MVP in a few weeks.

The main boss battle is trust and integration.

Video editors are protective of their workflows. They won't use a tool if they think it might corrupt a file or mess up their project. Your app has to feel solid, reliable, and professional. The marketing can't be about flashy features; it has to be about safety and dependability. The phrase "*EditKit never touches your original files*" should be written everywhere.

### **The “No-Brainer” Pricing**

Keep it simple.

- \$12 per month.

That's it. An editor's hourly rate is anywhere from \$40 to \$150+. If EditKit saves them just 15 minutes of tedious work *in an entire month*, it has already paid for itself. It's an impulse buy anchored to the value of their time, not the complexity of your code. 🎉

### **The Pre-Flight Check (Don’t Build a Thing... Yet)**

We have the idea. But we don't write a single line of code until we validate it. This is the fun part.

#### Step 0: The “Fake It ’Til You Make It” Test

1. Go to a site like Carrd or Webflow and build a simple one-page website.
2. The headline: “Stop Wasting Time on Repetitive Editing Tasks.”
3. Clearly describe the three MVP tools. Use simple animated GIFs to *show* them working (you can fake these with screen recording software).
4. Have one button on the page: “Get Early Access.”

5. When someone clicks it, just collect their email in a Google Form.

This entire test costs less than \$20 and a weekend of your time.

## Next Steps

- Go back to the source. Take your new landing page and post it in r/editors, r/postproduction, and other editor communities. Be honest: “Hey everyone, I saw a thread here a while back about custom macros. I’m exploring an idea for a simple tool to handle this stuff without any coding. Would this be useful to you?” The feedback will be brutal and invaluable.
- Talk to 5 people. Email everyone who signs up and ask for a 15-minute chat. Ask them one question: “Can you show me the most boring, repetitive task you did this week?” Their pain is your product roadmap.

• • •

**Thanks for reading. If you want more in-depth analysis like this, I invite you to join my community over on Substack.**

👉 [Click here to subscribe for free.](#)

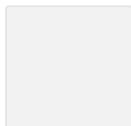
SaaS

Business

Software Development

Entrepreneurship

Side Hustle



Follow

## Published in The Micro-SaaS Corner

249 followers · Last published Oct 28, 2025

Your weekly dose of user validated Micro-SaaS ideas. We uncover niche problems and underserved markets, giving you the blueprint for your next business. For indie hackers, developers, and aspiring founders ready to build.

[Follow](#)

## Written by Mihai Sandu

517 followers · 95 following

A software developer interested in writing about programming, SaaS, investing and self-development.

Twitter @mihais77

## Responses (1)



Bgerby

What are your thoughts?



james co

5 days ago (edited)

...

software as a services is a best choice for the best now working ideas.

<https://bathroomremodelingknoxvilletn.com/>



-- [Reply](#)