

Set up ELU.

ELU is a session-replay tool that surfaces the exact moments where users get stuck. This guide installs it and walks through the two prompts from the video. Three steps, about five minutes.

STEP 01 Sign up at [eLu.dev](https://elu.dev)

Create an account and copy your API key from the dashboard.

STEP 02 Install with the prompt on the next page

Paste it into Claude Code, Cursor, or any coding agent. Add your API key. It detects your framework and handles the rest.

STEP 03 Connect your GitHub repo

In the ELU dashboard, link your repo. This activates the MCP so Claude can read your live friction data directly in conversation.

Give this to your coding agent.

Paste into Claude Code, Cursor, or any coding agent. Replace `[YOUR_ELU_API_KEY]` with your key from elu.dev/dashboard. It detects your framework and handles everything else.

INSTALL PROMPT

I signed up for ELU Analytics (<https://elu.dev>) – a product analytics SDK, similar to PostHog or Plausible. Please install it in this project.

Add the snippet below to the `<head>` of every page. In framework projects (Next.js, Remix, SvelteKit, Astro, etc.), put it in the root layout or app shell so it runs on every route. Skip if it is already present, and leave any other analytics tags alone.

```
<script async
  src="https://elu.dev/v1/[YOUR_ELU_API_KEY].js"
></script>
```

Copy verbatim — only swap `[YOUR_ELU_API_KEY]`

Find your unreasonable moment.

Run this inside **Claude with ELU MCP connected**. Claude pulls your real friction data and brainstorms the most surprising thing you could build.

PROMPT 01

Use ELU MCP to pull my top 3 friction points from the last 7 days.

For each one:

1. What the user was trying to do and exactly where they stopped
2. The obvious fix most teams would ship
3. The unreasonable version – if this were a 5-star hotel, what would they do before the guest even asked?

Don't constrain it to one pattern. Could be proactive, personalised, auto-configured, anticipated – whatever makes the user think: **"I can't believe they did that."**

Rank all 3 by wow factor x build speed.
Tell me which one to build and why.

Now build it.

Send this as a **follow-up in the same conversation** — no copy-pasting required. Claude writes the full build prompt for a coding agent.

PROMPT 02

Take the top recommendation above.

Write a complete prompt for a coding model to build this feature end-to-end. The prompt must include:

- What to build and why
- The exact trigger and detection logic
- The tools and APIs to use
- Step-by-step build instructions
- End with: **"Start with step 1. Show me the code."**

Write it so the coding model can execute without asking me anything.

Send as a follow-up in the same chat • No pasting needed