

O TEU CLAUDE CODE DESMONTA QUALQUER *app*

O REA (Reverse Engineer Anything) é uma ferramenta grátis e de código aberto que deixa o teu agente de programação abrir um app por dentro, mostrar como uma função foi feita e te ajudar a criar a tua versão. Aqui tá o passo a passo pra instalar e os pedidos prontos pra tu copiar e colar.

PASSO 01 / 04

Confere se tu tem o Node.js

GRÁTIS

O REA precisa do Node.js, que é o programa que roda ferramentas como essa no teu computador. Abre o terminal (ou pede pro próprio Claude Code) e roda o comando abaixo. Se aparecer v22.19 ou mais novo (ou v24.11, ou v26), tá pronto. Se não aparecer nada, baixa a versão LTS em nodejs.org.

COMANDO

```
node -v
```

github.com/morluto/rea · "Node.js 22.x (≥22.19), 24.x (≥24.11), or 26+, plus npm"

PASSO 02 / 04

Instala com um pedido só

OFICIAL

Cola este pedido no Claude Code (funciona igual no Codex, no Cursor, no Gemini CLI e no Grok Build). Ele mostra o que vai mudar antes de mexer em qualquer coisa. Tu lê, aprova e depois fecha e abre o Claude Code de novo.

PEDIDO PRONTO (DO SITE OFICIAL)

```
Instale o REA e conecte-o a este agente de programação usando  
npx rea-agents@latest setup  
Mostre primeiro as alterações propostas e espere minha aprovação. Depois,  
verifique a instalação.
```

OU, SE TU PREFERIR, DIRETO NO TERMINAL

```
npx rea-agents@latest setup
```

rea.tools/pt-BR · guia passo a passo: rea.tools/pt-BR/get-started

PASSO 03 / 04

Pergunta como uma função *funciona* OFICIAL

Escolhe um app que tu usa e uma função que tu queria ter no teu produto. Troca "Notes" pelo nome do app e "a busca" pela função.

PEDIDO PRONTO (DO README EM PORTUGUÊS)

Investigue como a busca funciona no aplicativo Notes, mostre as evidências e crie uma funcionalidade semelhante para o meu projeto.

PRA TESTAR COM UM EXEMPLO REAL (O MESMO DO VÍDEO)

Use REA para rastrear como Notion copia texto e HTML para a área de transferência do sistema. Mostre o código em cada etapa.

rea.tools/pt-BR/showcase/notion · estudo de caso oficial

PASSO 04 / 04

Cria a tua versão DICA NOSSA

Depois que ele explicar, pede a tua versão. Pedir o plano antes evita que ele saia mexendo no teu projeto inteiro.

PEDIDO PRONTO

Agora me explica em português simples, passo a passo, como essa função funciona. Depois cria uma versão parecida no meu projeto, com as ferramentas que eu já uso. Antes de mexer em qualquer arquivo, me mostra o plano e espera a minha aprovação.

POR ONDE COMEÇAR

Começa por app de computador feito em JavaScript

Apps como o Notion (feito em Electron) e sites são o caminho mais fácil: essa análise não precisa de nenhum programa extra. Pra programas nativos, o REA usa o Hopper, o Ghidra ou o IDA, e a instalação pode instalar o Hopper se tu aprovar.

github.com/morluto/rea · FAQ "Do I need Hopper, Ghidra or IDA?"

ANTES DE USAR

Três cuidados

- 1. Usa em app que tu tem direito de analisar e respeita os termos de uso de cada um. Entender como uma função funciona pra fazer a tua é uma coisa; copiar o produto dos outros é outra.
- 2. A análise roda no teu computador e o REA não sobe o teu app. Só que os resultados vão pro teu agente, e a empresa da IA (Anthropic, OpenAI, Google) tem a própria regra de dados.
- 3. O REA muda rápido. De vez em quando, atualiza com o comando abaixo e reinicia o agente.

```
npx rea-agents@latest setup
```

O QUE A GENTE CONFERIU EM 16/10/2026	FONTE
Repositório oficial, licença MIT (grátis e código aberto)	github.com/morluto/rea
283.896 estrelas no GitHub	github.com/morluto/rea
Funciona com Claude Code, Codex, Cursor, Gemini CLI e Grok Build	README
Analisa programas do computador, apps de Android, sites, apps JavaScript e Electron e .NET	README, "What you can analyze"
Inspeciona o app sem o código-fonte, explica, mostra as evidências e cria uma versão pro teu projeto	README em português
A análise acontece no teu computador	README, FAQ "Does REA upload my app?"

Quer aprender a usar tudo isso do zero, em português? **Entra na lista da comunidade Prosperando com IA:** aliceallgayer.com.br/lista?de=prosperando