

Founder Skill no Claude: como instalar e usar as 11 skills

Founder Skill no Claude: como instalar as 11 skills que testam um negócio, o que cada uma faz, a ordem certa de rodar e o que é simulação.

1. Em 1 minuto
2. Faça agora
3. As 11 skills
4. O exemplo da cafeteria
5. O que é simulação e o que não é
6. Deu problema?
7. Para quem quer ir além
8. Fontes

01 Em 1 minuto

O QUE É

Um pacote de 11 skills para o Claude que testam um negócio antes do lançamento.

O QUE VOCÊ GANHA

Conselho, concorrentes, painel de clientes, preço, financeiro, marketing e um plano final com veredito.

QUANTO CUSTA

Nada. É gratuito, de código aberto, sem cadastro, sem chave de API e sem nada para conectar.

Palavra difícil: Skill é um conjunto de instruções que ensina o Claude a fazer uma tarefa de um jeito fixo. Cada skill aqui faz o papel de uma função de empresa.

02 Faça agora

Se você nunca instalou uma skill, comece pelo Passo 1. Não precisa do terminal.

PASSO 1

Escreva a sua ideia

Descreva o negócio em poucas linhas: o que é, para quem e o preço.

A primeira skill parte do que você contar. Sem isso, ela não tem por onde começar.

PASSO 2

Cole o link no Claude

Abra o Claude. Cole o link do projeto e, embaixo, escreva `install skill`:

```
https://github.com/Jakeschincariol/founder-skill

install skill
```

Esse é o jeito mais simples. O outro jeito (como plugin, no Claude Code) está em "Para quem quer ir além".

PASSO 3

Rode o conselho

Comece pela primeira skill, a do conselho de administração:

```
/founder-board
```

Três conselheiros dão nota à sua ideia, apontam o que pode matar o negócio e votam.

Dica: Use sempre a mesma pasta de projeto. As skills guardam o que escrevem numa pasta `founder/` dentro dela.

PASSO 4

Siga a ordem das skills

Depois do conselho, rode as outras nesta sequência:

```
board -> competitors -> consumer -> pricing -
> offer -> cfo -> marketing -> brand -> ops -
> launch -> plan
```

Cada skill lê o que a anterior escreveu. Por isso a ordem muda o resultado.

Atenção: O financeiro, o painel, o preço e o plano fazem contas. Para isso, precisam do Python 3.8 ou mais novo no computador. Veja como conferir em "Para quem quer ir além".

PASSO 5

Leia o veredito do plano

A última skill, `/founder-plan`, junta tudo num plano só. No fim, ela dá um veredito calculado pelos números:

- **Profitable:** dá lucro.
- **Not yet:** ainda não.
- **Incomplete:** incompleto.

03 As 11 skills

Cada linha é um comando. Você digita o comando no Claude e a skill começa.

Comando	O que faz
`/founder-board`	**Conselho de administração.** Três conselheiros, cada um com um método publicado (oferta, de $100M Offers; monopólio, de Zero to One; produto, da biografia do Steve Jobs por Isaacson), dão nota à ideia, apontam o que pode matar o negócio e votam.
`/founder-competitors`	**Pesquisa de concorrentes.** Mapeia concorrentes diretos, indiretos e substitutos em fontes públicas, com preço, posicionamento e do que os clientes deles reclamam. Cada fato vem com link.
`/founder-consumer`	**Painel de consumidores.** Cria compradores simulados (100 por padrão) a partir do seu cliente ideal, cada um com renda, hábito e objeção próprios, e conta quem compra, quem não compra e por quê.
`/founder-pricing`	**Preço.** Transforma as respostas de preço do painel numa faixa aceitável, compara com os concorrentes e a sua margem, e escolhe o que testar.
`/founder-offer`	**Oferta.** Monta a oferta (bônus, garantia, urgência real, nome) que responde às objeções do painel e testa de novo no painel.
`/founder-cfo`	**Diretor financeiro.** Quanto sobra por venda, margem real, vendas por dia para empatar, o ano 1 mês a mês, quanto dinheiro precisa, quando o investimento volta e simulações de "e se".

Comando	O que faz
`/founder-marketing`	**Diretor de marketing.** Posicionamento, os canais que os seus compradores usam, uma campanha de lançamento de 30 dias com datas, dez ganchos e o máximo que dá para pagar por cliente.
`/founder-brand`	**Marca.** Opções de nome com as checagens de marca registrada, domínio e perfil a fazer, tom de voz, promessa e um briefing do visual.
`/founder-ops`	**Operação.** Fornecedores, equipe, rotina do primeiro dia, ferramentas, licenças a conferir e lista de riscos. Os custos reais voltam para o financeiro.
`/founder-launch`	**Lançamento.** Primeiro um teste barato no mundo real, depois a contagem regressiva com datas, o roteiro do dia do lançamento e os primeiros 30 dias.
`/founder-plan`	**Plano de negócio.** Junta tudo num plano só, com um veredito calculado pelos números: Profitable (dá lucro), Not yet (ainda não) ou Incomplete (incompleto).

Ideia ainda no começo?

O painel tem um modo rápido. Ele usa 20 compradores em vez de 100:

```
/founder-consumer --quick
```

Dica: Dá para rodar qualquer skill sozinha. Mas na ordem, uma aproveita o trabalho da outra.

04 O exemplo da cafeteria

O projeto traz um exemplo completo: uma cafeteria de matcha em Toronto. Ela vende matcha latte gelado a $6,50 o copo. Todos os dados estão nos arquivos de exemplo do projeto.

O que cada skill fez

1. **Conselho:** avalia a ideia. Cada conselheiro dá nota, lista os riscos e vota em investir, investir com condição ou passar.
2. **Concorrentes:** mapeia os bares de matcha e cafeterias a uma distância a pé. Mostra quanto cobram pelo latte e do que as avaliações reclamam.
3. **Painel:** monta o cliente (profissionais urbanos de 24 a 38 anos que compram café quase todo dia). Cria 100 compradores e devolve quem compra, quem passa e os motivos com as palavras deles, como "o meu café é mais perto" e "$6,50 é caro para uma bebida".
4. **Preço:** confere os $6,50 contra a faixa aceitável do painel e contra os concorrentes.
5. **Oferta:** responde às principais objeções com preço de semana de lançamento e cartão fidelidade. Depois roda um painel rápido para ver se a taxa de compra muda.
6. **Financeiro:** chega nos números da tabela abaixo.
7. **Marketing, marca, operação e lançamento:** viram a campanha de 30 dias, o nome e o visual, os fornecedores e a rotina. O lançamento tem datas e começa com uma loja temporária para testar compradores reais.
8. **Plano:** junta tudo e calcula o veredito.

Os números do financeiro

Conta	Resultado
Quanto sobra de cada copo depois dos custos dele	$4,76
Vendas por dia para empatar	186 copos
Margem de lucro com os 383 copos por dia planejados	38%
Resultado da operação no ano 1, com começo lento	$93.926
Investimento inicial	$25.280, recuperado no mês 8
Dinheiro em caixa que o dono precisa ter antes de se pagar	$34.092

Quando sai "Profitable"

Só sai Profitable se as quatro coisas forem verdade:

- Cada venda dá dinheiro.
- O ano 1 tem lucro.
- O ponto de empate cabe na capacidade da loja.
- Uma parte suficiente do painel compra.

05 O que é simulação e o que não é

Os avisos abaixo são do próprio projeto.

Os conselheiros não são essas pessoas

O conselho usa métodos publicados. As três lentes são resumos, com palavras do projeto, de ideias de três livros:

- $100M Offers (Alex Hormozi).
- Zero to One (Peter Thiel com Blake Masters).
- A biografia do Steve Jobs por Walter Isaacson.

O conselho nunca fala como eles nem diz que eles apoiam nada. O projeto não tem ligação com nenhum deles.

O painel não são clientes de verdade

São compradores simulados. Servem para achar objeções, grupos de cliente e palavras antes de gastar.

Não é previsão. Modelos de linguagem tendem a concordar, então a taxa de compra é o teto.

> **Atenção:** O projeto recomenda confirmar as decisões grandes com pessoas reais. O `/founder-launch` começa por esse teste. E não use resposta do painel como depoimento.

Os números dependem dos seus dados

O financeiro marca toda estimativa como estimativa e pede orçamento real.

- Custos: confirme com fornecedores.
- Estrutura, folha e impostos: confirme com um contador.

A pesquisa usa fontes públicas

Tudo vem com link. Nada de concorrente, preço, avaliação ou citação inventados.

Não é consultoria

Não é consultoria financeira, jurídica nem tributária. Licenças, marca registrada e o que se pode prometer dependem do lugar. As skills dizem o que conferir e onde. Quem confirma é um profissional.

06 Deu problema?

O que aconteceu	O que fazer
`/founder-board` **não existe**	Instalado como plugin, o nome é `/founder-skill:founder-board`. Para o nome curto, copie as pastas (veja "Quer os comandos sem o prefixo?" em "Para quem quer ir além").
O comando `/plugin` **não existe**	É do Claude Code. No app do Claude, use o Passo 2 (colar o link e `install skill`).
Erro ao rodar as contas do financeiro ou do painel	As ferramentas pedem Python 3.8 ou mais novo. Confira com `python3 --version`.
Uma skill não acha o que a outra fez	Rode na mesma pasta de projeto e na ordem do guia. Tudo fica em `founder/`.
O painel saiu otimista demais	O projeto avisa que modelos tendem a concordar. A taxa de compra é o teto, não a previsão.

Pronto, o principal acabou aqui. Daqui pra baixo é a parte técnica, pra quem quer entender os detalhes.

07 Para quem quer ir além

Quem fez e a licença

O Founder Skill foi criado por Jake Schincariol. É de código aberto, com licença MIT. Tinha 2.159 estrelas no GitHub em 29/10/2026.

As 11 skills cobrem: conselho de administração, pesquisa de concorrentes, painel de consumidores, preço, oferta, financeiro, marketing, marca, operação, lançamento e plano de negócio.

O que você precisa ter

- [ ] **Claude ou Claude Code.** A instalação mais simples é colando o link no Claude. Como plugin, precisa do Claude Code.
- [ ] **Python 3.8 ou mais novo**, só para as ferramentas de cálculo (financeiro, painel, preço e plano).
- [ ] **Uma pasta de projeto.** As skills guardam o que escrevem numa pasta `founder/` dentro dela.
- [ ] **A ideia do negócio descrita em poucas linhas:** o que é, para quem, o preço.

Conferir o Python

São quatro ferramentas de cálculo, só com a biblioteca padrão do Python. Não tem nada para instalar com `pip`, e nenhuma acessa a internet.

> **Palavra difícil:** Terminal é a janela onde você digita comandos para o computador, em vez de clicar.

Teste no terminal:

```
python3 --version
```

Instalar como plugin, no Claude Code

São dois comandos, um de cada vez, digitados dentro do Claude Code:

```
/plugin marketplace add Jakeschincariol/founder-skill
```

```
/plugin install founder-skill@founder-skill
```

Instalado como plugin, o Claude Code coloca o nome do pacote na frente de cada comando. Em vez de `/founder-board`, fica `/founder-skill:founder-board`, e assim com todas.

Quer os comandos sem o prefixo?

O projeto mostra outro caminho: baixar o repositório e copiar as pastas das skills. Rode no terminal, um de cada vez:

```
git clone https://github.com/Jakeschincariol/founder-skill.git
```

```
cp -r founder-skill/skills/founder-* ~/.claude/skills/
```

Para valer só num projeto, copie as mesmas pastas para `.claude/skills/` dentro dele.

Sem Claude Code

Dá para colar o conteúdo de um `SKILL.md` no começo de uma conversa. Ele roda como um modo. Segundo o projeto, você perde os subagentes e as ferramentas em Python, mas o método funciona.

Como uma skill passa trabalho para a outra

Cada skill lê o que as anteriores escreveram na pasta `founder/` do projeto. Exemplos do próprio projeto:

- O `/founder-board` escreve `founder/idea.md`. As condições dos conselheiros viram pendências para as próximas skills.
- O `/founder-consumer` usa as reclamações que o `/founder-competitors` achou. O preço e a oferta partem das respostas e objeções do painel.
- O `/founder-plan` junta tudo em `founder/business-plan.md`.

08 Fontes

Conferido em 29/10/2026, direto no repositório oficial:

- Founder Skill (Jakeschincariol/founder-skill)
- Skill do painel de consumidores (founder-consumer)
- Site do autor, Jake Schincariol

Tem guia novo a cada vídeo.

Todos em marraads.online · siga @marraads