

# Build your own CRM instead of renting one

You describe the system your business actually needs, in plain English, and it gets built on your own computer.

## WHAT IT IS

### A tool that writes the software for you

Claude Code is made by Anthropic. You talk to it in ordinary sentences and it builds real, working software on your machine — it writes the files, sets up the database, and runs the result so you can click through it. You never open a code editor and never write a line of code yourself.

## WHAT IT DOES

### Builds the real thing, not a mockup

- A working pipeline board with your own stages, deal values, and drag-to-move
- Contacts and companies that link to each other and keep a full history
- A real database, so nothing disappears when you close the window
- Any change you ask for in plain English — new fields, new stages, a different look

## WHAT IT REPLACES

### The subscription you never stop paying

Commercial CRMs bill per person, per month, forever — and you still have to run your business the way their software expects. Building your own inverts that: the software fits your process, you own it, and the only running cost is the Claude plan you need to build it.

## BEFORE YOU START

### Two things, and how to check you have them

→ **A Mac on macOS 13 or later, or a Windows 10 / 11 PC.**

**On Mac:** click the Apple logo (top-left of your screen) → **About This Mac**. The version is at the top. If it is below 13, run **System Settings** → **General** → **Software Update** first. **On Windows:** press the **Windows key**, type "winver", press Enter.

→ **A paid Claude plan — Pro or Max.**

The free Claude plan does not include Claude Code. Pro is \$20 per month, or \$17 per month paid yearly. Max starts at \$100 and is only worth it once you build every day. **To check:** [claude.com](https://claude.com) → sign in → **Settings** → **Billing**. Start on Pro.

## STEP 1

### Install Claude Code

#### 1 Open Terminal.

On Mac: press **⌘ + Space**, type "Terminal", press Enter. On Windows: press the **Windows key**, type "PowerShell", press Enter. A plain window with nothing but text appears — that is correct, it is just somewhere to paste commands.

#### 2 Paste this in and press Enter. On a Mac:

```
curl -fsSL https://claude.ai/install.sh | bash
```

On Windows PowerShell, use this instead:

```
irm https://claude.ai/install.ps1 | iex
```

It takes about a minute and may sit there showing nothing. That is normal — leave it until your cursor comes back.

#### 3 Check it worked.

```
claude --version
```

A working install prints a version number, something like **2.1.173 (Claude Code)**. If it says "command not found", close the window completely, open a new one and try again.

Prefer not to touch a terminal at all? Anthropic also publishes a Claude Code desktop app you install by double-clicking, like any other program. Same tool, same result — get it at [claude.com/download](https://claude.com/download).

## STEP 2

### Make a folder and start

#### 1 Create a folder for the project and move into it.

```
mkdir ~/my-crm && cd ~/my-crm
```

This makes an empty folder called "my-crm" in your home directory. Everything built lands here, and you can open it in Finder like any other folder.

#### 2 Start it.

```
claude
```

The first time, a browser window opens asking you to sign in to your Claude account. Approve it, then come back to the Terminal window. After that it starts straight away.

## STEP 3

### Ask for a plan before anything gets built

This step decides whether you end up with a real system or a mess, and it is the one most people skip. Describe your actual business, then explicitly ask for a plan — not code.

```
I run a small agency. I want a CRM: companies, contacts, and deals
moving through stages I choose. Don't write any code yet. Ask me what
you need to know, then show me a plan broken into phases I can
approve one at a time.
```

It will ask you questions back — what your pipeline stages are called, what you track on a deal. Answer them properly. Every answer here is something you don't have to correct later.

## STEP 4

### Build it one phase at a time

#### 1 Approve one phase, not the whole plan.

Say: "Build phase 1 only. Then run it and show me it working before you go further." A tool built and checked in pieces stays fixable. One built in a single pass rarely does.

#### 2 Make it prove each phase before moving on.

Ask for a screenshot of the working screen every time. If it says something is finished but cannot show you, treat it as unfinished and say so.

#### 3 Change anything by just saying it.

"Rename the Qualified stage to Discovery." "Add a phone number field to contacts." Plain sentences are the whole interface. That's it.

## STEP 5

### Open it and use it

When it runs the app it prints an address like **http://localhost:3000**. Paste that into your browser. "Localhost" just means the app is running on your own computer rather than out on the internet — it behaves like any website, only privately.

#### GOOD TO KNOW

**Set up an undo button on day one.** Ask it to "set up Git and commit after every phase". Git keeps a snapshot of every version, so any change can be reversed by asking. Without it, a bad edit is permanent.

**It runs on your computer, not the internet.** That is fine for one person. Putting it online so a team can log in is a separate job, and worth doing only once the thing works.

**Vague requests produce vague software.** "Make it better" gets you noise. "The deal cards are too tall, show four per screen" gets you the change you wanted.

**Start narrower than you think.** Contacts and a pipeline first. Invoicing, reporting and automations are easy to add later and expensive to get wrong early.