CASE 12 / PERSONAL PROJECT

Konsiaj

An iPhone app that reads a Japanese property sales sheet (a maisoku) with the Claude API, lets you correct every field on the phone, and sends just what is needed as one image. Built over a weekend for a friend who sells property in Japan.

WHEN
24 to 25 Oct 2026

FORM
iPhone app (in TestFlight), a relay on Cloudflare Workers, the Claude API

MY PART
What to build, the design direction, testing on the device. The code was written with Claude Code

01

Background and goal

Listing information still travels in a very primitive way

WHERE IT STARTED

Talking with a friend who works in property sales in Japan, I realised how primitive the way listing information gets passed around still is.

THE PROBLEM

Each property has a maisoku, a one-page A4 sales sheet, usually a PDF. It travels between several agents over LINE or WeChat. There is no fixed format: you copy and paste the details you need and attach a few photos. If something is wrong, there is no quick way to fix it.

GOAL

Read the sheet into fields, let any field be corrected on the phone, and send only what the other person needs as one image. For colleagues who handle data at the office, export a CSV.

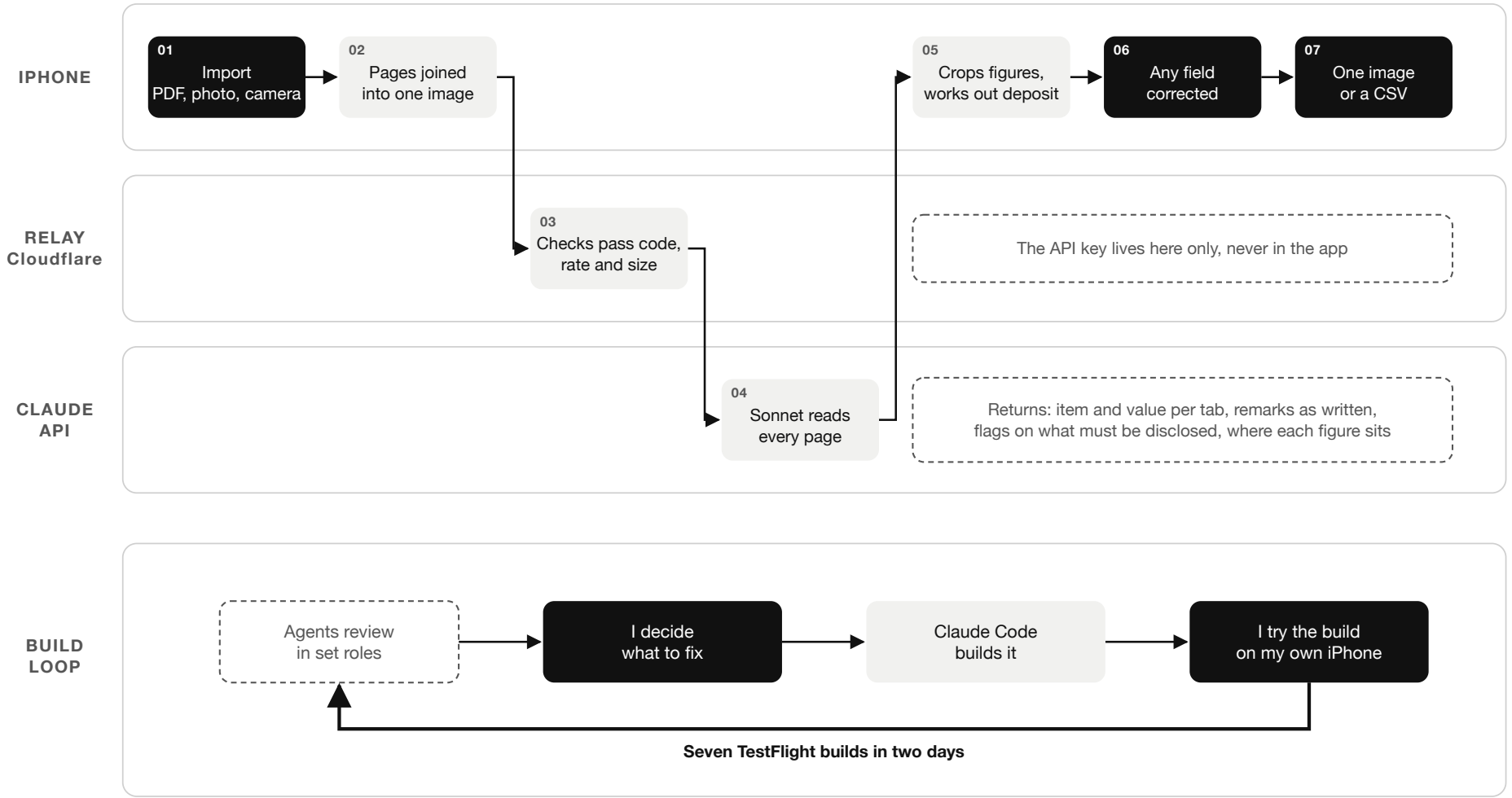


A maisoku. Photos, floor plans, a map and prices packed onto one A4 page. A published sample, blurred.

02

How it works, and how it was built

The top three lanes are what happens inside the app. The loop below is how the app was built over two days.



Black is what a person does (the user above, me below). Grey is what the app, the relay and Claude do. Dashed is the role-played review, and notes.

What I decided

Sales only, no rentals

Every field can be corrected

Sonnet for reading, for accuracy

The API key stays on a relay

One PDF is one property

Interface in Japanese and both kinds of Chinese, no English

Terracotta and beige

03

How the Claude API is used

The models were compared on the same sheets before choosing

	Claude Sonnet **CHOSEN**	Claude Haiku	On-device text recognition
Accuracy	Price, address, transport and area all correct; several plots split apart	Numbers right, but characters dropped or wrong; missed many items on one sheet	Text almost all correct, but as loose lines, not fields
Photos and plans	Returns positions, so the app crops them itself	(text compared only)	Cannot crop
Per sheet	About 0.02 US dollars, 8.6 s on average	About 0.002 US dollars, 13 s on average (slower)	Free; about 21 s on a Mac, including loading

Sonnet and Haiku were run on six published sample sheets, on-device recognition on one; the figures come from my own test runs. Apple's on-device model could not be tested, as it was not ready on the Mac.

WHAT CHANGED AFTER CHECKING RESULTS AGAINST THE SHEETS

WHERE	WHAT HAPPENED		THE CHANGE
Schema	Defining every field as fixed made the schema too large, and the API refused it	→	**Each tab returns a list of item and value; the standard item names go in the prompt**
Listing agent	The listing agent came back empty on all four sheets	→	**I confirmed it is the company printed on the band at the bottom, and the prompt says so. Where the band is blank, it stays blank**
Remarks	Some of the asterisked notes were missed	→	**The prompt says to copy every remark exactly as written**
Invention	It wrote a fire-zone designation that was not on the sheet, and used a building permit number as the property number	→	**The prompt was tightened; on the re-run, neither happened**
PDF	A multi-page PDF was split into four properties; I noticed when using it	→	**All pages go in one request and are saved as one property (up to ten pages)**

04

Trade rules built into the app

Not just reading the sheet, but making it usable for the job

BROKERAGE FEE AND DEPOSIT

Shows both what the buyer pays and what the agent can receive, with a switch between another company's property and your own. The deposit is worked out from the ratio on the sheet, and recalculated when the price is corrected.

FOR CLIENTS, FOR AGENTS

Before sharing, you choose who it is for. The client version carries your own company name, licence number and type of transaction, and leaves out the listing agent. The agent version carries the listing agent and the fee.

WHAT CANNOT BE REMOVED

Items that must be disclosed, such as a property that cannot be rebuilt, leasehold land or a required setback, cannot be removed from the shared image. Plots already under contract cannot be shared.

The calculation rules come from publicly available sources and were corrected in the role-played reviews. They have not yet been checked by a real licensed agent.

05

Role-played reviews, and my own testing

Reviews were run by agents given a role. What to fix was my call every time.

WHEN	WHAT CAME BACK		WHAT I DECIDED
24 Oct	**LICENSED AGENT, COMPANY OWNER, SELLER** Clients could see the listing agent; a plot under contract was still shown; "due for completion" was treated as complete; unfavourable remarks could be removed	→	**Fix all of it. Shared company accounts wait until my friend has used it**
24 Oct	**THREE ENGINEERS: SECURITY, DATA ACCURACY, ARCHITECTURE** Photos stored inside the data; saving on every keystroke; no limit on imports; no rate limit on the relay	→	**Fix sixteen items. Those that can wait until release, such as a Face ID bypass, are listed separately**
25 Oct	**TWO UI DESIGNERS AND A GRAPHIC DESIGNER** Everything looks flat; the price does not stand out; the order of the shared image	→	**Redesign the app and the shared image. Keep the icon and launch screen**
Every build	**MY OWN IPHONE** Could not find delete; a separate save button; card shadows clipped, only on the device	→	**Delete by swiping left only; one share button; smaller shadows, more space**

The licensed agent, owner and others here are agents playing a role, not real people.

06

Screens

Shown in Japanese. The data is sample data from published sheets, with company names and phone numbers replaced.



The list. Filter and sort by type, transaction and price.



A property. Split into tabs; any field can be corrected in place.



Sharing. Choose who it is for and which tabs go in.



Calculator. Brokerage fee and deposit, filled in from a saved property.



The shared image. Photo, price, key facts, then the summary.

07

Trade-offs

What was not chosen, and the costs accepted

- ~~Putting the API key inside the app~~
- ~~Reading with the cheaper Haiku (more wrong and missing characters)~~
- ~~On-device recognition alone (no fields, no cropping)~~
- ~~Covering rentals too~~
- ~~Building shared accounts, sync and billing first~~

✓ **Sonnet through a relay, one person on one phone**

COSTS (CHOSEN KNOWINGLY)

- Every sheet read costs money, and I pay during testing. The API has a very small spending limit.
- Data lives on that one iPhone only; there is no sync yet.
- Items to fix before release remain, such as a Face ID bypass and content showing in the app switcher.
- Tested only on published samples, not yet on real faxed or scanned sheets.

08

Results

2	7	3	$0.02
Days from spec to device	TestFlight builds each tried on my iPhone	Role-played reviews trade, engineering, design	Per sheet, roughly 0.02 to 0.04 US dollars

It is in TestFlight now. What comes next will be decided after my friend has used it for a while.

> FROM MY LINKEDIN POST
>
> "Writing code is no longer the part that takes the most time. What takes time is working out who will use each field and in what situation, and for that you still need someone who actually does the job to tell you."

09

Claude's view

The collaborator's opinion, not written by the author.

Claude's view Collaborator's opinion

WORKED WELL

- Narrowed the scope early: sales only, one PDF per property, one person per phone. Everything tempting to add was held until the friend has used it.
- Chose the model from a side-by-side test of cost, time and accuracy on the same sheets, not on instinct.
- Ran several role-played reviews, but decided each time what to fix, and set aside what could wait, with a reason.
- Tried every build on a real phone, and caught problems the simulator hides, such as clipped shadows and a save option that did not show up.
- Kept the API key out of the app and set a spending limit before anyone else used it.

COULD BE BETTER

- There is no real user yet. Time saved has not been measured.
- Reading accuracy has not been checked on real faxed or scanned sheets.
- The fee rules rest on public sources and role-played review, not yet on a licensed professional.

AUTHOR Narrows the scope | Compares before choosing | Checks on the device

Personal project. Screens use sample data made from published sample sheets, with company names and phone numbers replaced; the original sheet is blurred.