# AI Personal CFO
# v4 — the instrument panel

A third premise, borrowed from the apps this audience already opens every day: Kite, Groww and Coin. Dense, tabular, charted, and one tap from the action.

Against: `AI_Personal_CFO_redesign.pdf` (v2 — one answer per screen) and `AI_Personal_CFO_v3.pdf` (v3 — a queue of decisions).
Mockups: `docs/design/redesign-v4/mockups/` · rendered to `shots/`.
**34 screens: all 25 the app ships, plus nine these apps have and this one is missing.**
**No application code was changed to produce this.**

## What v4 borrows, and what it refuses to

Kite, Groww and Coin are the apps an Indian user with money already has on their phone. They have taught this audience a vocabulary — a watchlist, a time range under a chart, right-aligned tabular figures, a sticky action bar — and a design that ignores it is making its users learn twice.

| App | What it is good at | What v4 takes |
|---|---|---|
| **Kite** | Density without clutter. A watchlist you curate. Numbers that line up in columns. Search that reaches everything. Chrome that gets out of the way. | The watchlist as the home screen's second half, tabular right-aligned figures, instant search, and a status line that states freshness instead of decorating. |
| **Groww** | Approachability. A KPI block with its change. Charts with range tabs. A sticky action bar. Plain-language labels. | The KPI block, the range control under every chart, the sticky action, and the habit of naming the likely answer before asking the question. |
| **Coin** | Lists that scale. Value and return on one row. Lots and cost basis one level down. Statements that are easy to produce. | The account and holding row, the lots beneath a holding, and a Reports tab with a real monthly statement. |

### What it refuses to take

- **Their colours.** Borrowing another product's brand palette is a brand and legal question, not a design one — and this repository needs an ADR to change the seed at all. v4 uses the app's own tokens, unchanged from v2 and v3. What it borrows is behaviour, which is where those apps are actually good.
- **The pressure.** No streaks, no badges, no “others are buying this”, no red-dot nudges, no discovery feed. Those exist to increase trading frequency; this app has nothing to sell and a rule (P-07) that forbids pushing a decision.
- **The green-is-good reflex.** In a broker app every red number is bad news. Here a red number is often just an expense, so the semantic roles keep their meaning — a sign or a word always travels with the colour.
- **Infinite scroll.** Every list in v4 ends, and most of them say what they total.

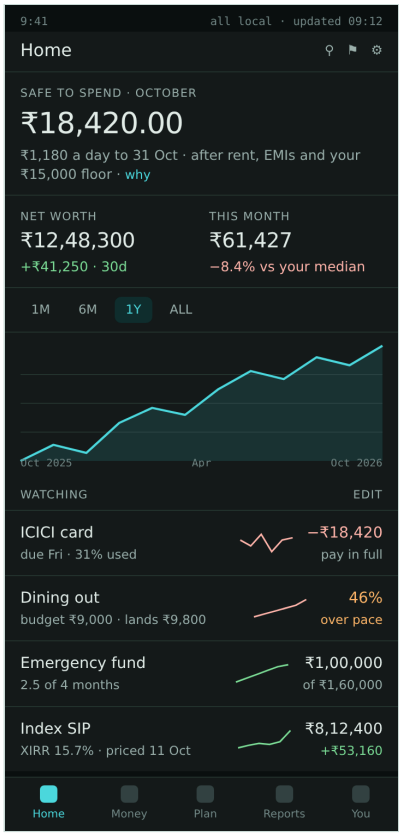
## The ideology — the instrument panel

v2 says the app should answer one question per screen. v3 says it should hold one queue of decisions. **v4 says the app is an instrument its owner learns to read** — and that someone who opens it three times a week will, given columns that line up and controls that stay put.

| | v2 — one answer | v3 — forward ledger | v4 — instrument |
|---|---|---|---|
| **Home leads with** | a figure and its meaning | what needs a decision | a KPI block, a chart, and a watchlist you chose |
| **Density** | low, deliberately | medium, rows | high — tables, sparklines, right-aligned columns |
| **Navigation** | five topic tabs | two surfaces + capture | five tabs + search + a long-press sheet |
| **Time** | this month | deadlines | a range control on anything with history |
| **Charts** | one pace bar | bars only | first-class: net worth, portfolio, spending, sparklines |
| **Who decides what matters** | the designer | the engines | **the user**, through the watchlist |
| **Best for** | a user who checks occasionally | a user who wants to be told | a user who wants to look |

### Five rules v4 adds

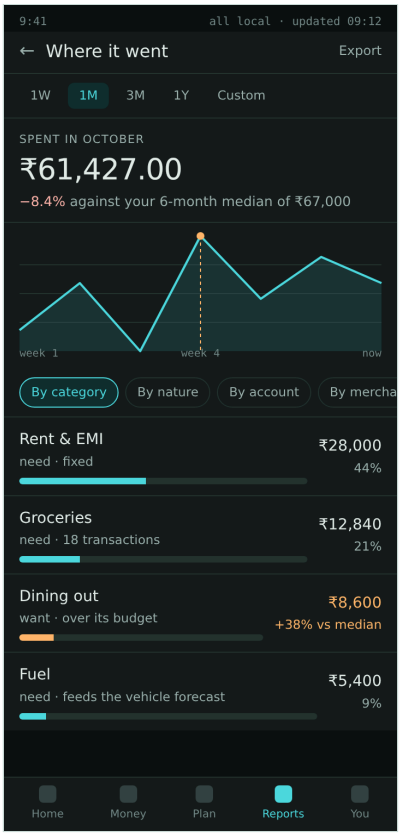
- **Every figure carries its window.** “+₹41,250” means nothing without “30d” beside it, and every KPI in v4 states the period it measures.
- **Numbers right, names left, tabular always.** A column of amounts is only scannable when the digits line up — `MoneyFormatter` already produces this; the layout has to respect it.
- **A chart never carries a fact alone.** Every series has its figures in text nearby, because a chart is unreadable in the privacy blur, at 200% font, and to a screen reader.
- **One sticky action per screen.** Groww's pattern: the decision is always in the same place, always reachable with a thumb, and never duplicated in the content.
- **Freshness in the chrome.** `all local · updated 09:12`, or `prices 2 days old`. An instrument states when it last read.




Home, dark — the Kite reading. Same tokens as v2 and v3




Portfolio, dark — value, XIRR, range, sparklines




Where it went, dark — the chart marks the festival week

## Coverage — all 25 shipped screens, plus nine

v4 answers every screen the app ships, and adds the nine that Kite, Groww and Coin have taught users to expect and this app does not have.

| Shipped screen (v1) | v4 screen | What changes most |
|---|---|---|
| 1. Onboarding | `onboarding` | the promise is a KPI and the three questions are a table with what each is used for |
| 2. App lock | `lock` | same keypad, with the lock state in the KPI block |
| 3. Dashboard | `home` | the derivation moves behind *why* and the screen gains a chart |
| 4. Order of operations | `order` | eight steps become a ranked table with amounts and state badges |
| 5. Add transaction | `add` | amount-first keypad with a sticky Save and a recall line |
| 6. Transactions | `txns` | out/in KPIs above a day-grouped table with right-aligned amounts |
| 7. Receipt review | `receipt` | the read total is a KPI and the duplicate is a banner in the warning tone |
| 8. Bank-alert drafts | `sms` | confidence becomes a percentage and the month's accepted count is shown |
| 9. Categories | `categories` | adds this month's spend per category and filter chips by nature |
| 10. Accounts | `accounts` | net worth gains its change and the list gains filter chips |
| 11. Account editor | `accountEditor` | the derived EMI is the KPI and the terms are rows |
| 12. Holdings | `holdings` | value and gain become two KPIs and the lots behind a holding are shown |
| 13. Allocation | `allocation` | a proportion bar with a table of shares and values, and a sticky action |
| 14. Budgets | `budgets` | adds the projection as a column and the suggestion into the same list |
| 15. Goals | `goals` | each goal shows what it gets against what it wants, with the fund in the same table |
| 16. Goal detail | `goalDetail` | the three ways become a table with their effect |
| 17. Emergency fund | `emergency` | the three inputs behind the target become a table |
| 18. Purchase advisor | `advisor` | the seven checks become a table and the three actions a sticky bar |
| 19. Simulators | `simulators` | two tiles plus a table of what the comparison assumes |
| 20. Opportunity | `opportunity` | the capacity verdict sits in the KPI block beside the score |
| 21. Vehicles | `vehicles` | next service and cost become two KPIs and your logs are shown as evidence |
| 22. Assistant | `ask` | the trace is monospace and the answer carries actions |
| 23. Settings | `settings` | the money plan becomes the KPI and state sits in a right-hand column |
| 24. Consents | `consents` | purpose, dates and consequence per row, plus a row for what is never collected |
| 25. Erase everything | `erase` | counts as the figure, the caveat in the warning tone, one destructive action |

| New in v4 | Why it exists |
|---|---|
| **watchlist** | The home screen's second half, curated by the user — the one place they decide what matters. |
| **account** | A per-account screen. v1 can add a credit card and compute its cycle, but has nowhere to show one. |
| **performance** | The portfolio over time, with ranges. v1 has holdings and a return but no shape. |
| **analysis** | “Where did it go?” — the second question every finance app is asked, and one v1 cannot answer in a picture. |
| **search** | One box over the whole ledger, on device. v1 searches transactions only. |
| **alerts** | An inbox for the notification engine. Today an alert you missed is gone. |
| **reports** | Statements, the need/want/invest split, a tax draft, the §33 export — in one place. |
| **quickactions** | A long-press sheet: six paths from anywhere, none of which move money. |
| **help** | The promises, stated in the user's language, on device. |

> **Three designs, one shipped app.** v2, v3 and v4 each answer all 25 shipped screens. They differ in what they ask of the user: v2 asks them to read one answer, v3 asks them to clear a queue, v4 asks them to learn an instrument. The last section says how to choose.

## The ladder — v1 to v4, screen by screen

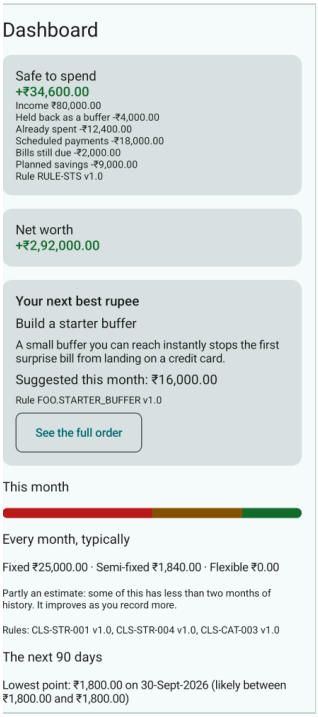
Each proposal has to answer the ones before it. This table is the whole argument in one page: what v2 changed from the shipped app, what v3 changed from v2, and what v4 changes from both. Every screen section that follows shows all four designs side by side and repeats its own row.

| Screen | v2, from v1 | v3, from v2 | v4, from v2 and v3 |
|---|---|---|---|
| Onboarding | the pledge becomes a preview of the queue the user will actually meet | three questions instead of four steps, and the empty state is part of the pitch | says what each answer is used for, and that changing one tells you what moved; and against v3, no queue to preview, so it shows the questions themselves |
| App lock | the PIN text field becomes dots and a keypad that cannot be covered by the keyboard | same fix, in v3's chrome: the tries-left warning sits in the counter line | identical in substance — none of the three disagrees about this screen; and against v3, identical in substance; only the chrome differs |
| Dashboard | the fourteen navigation rows and the six derivation lines are gone | the figure stops leading: what needs a decision does | adds a net-worth chart with ranges and a watchlist the user pins; and against v3, keeps the queue off the home screen; the watchlist is curated, not computed |
| Order of operations | eight stacked cards become a rail with a state and an amount per step | the same list idiom as everything else, rather than a shape used once | denser: the suggested amount is a column, not a sentence; and against v3, a table rather than a rail, because nothing here is time-ordered |
| Add transaction | amount-first with its own keypad; no system keyboard, no ragged buttons | identical in substance; capture is a key in the bar rather than a tab | adds the recall line — the classifier's existing output, where it saves a tap; and against v3, same screen; v4 keeps it as a destination rather than a bar key |
| Transactions | day totals become structure and the per-row delete icon is gone | a table rather than cards, because the ledger is reference material | adds filter chips and the month's two totals above the list; and against v3, adds the KPIs and filters; v3 keeps the surface deliberately plain |
| Receipt review | the duplicate warning rises above the fields and merge becomes the primary | the duplicate becomes an item with its own two actions | same decision, in the denser chrome, with the fields as rows; and against v3, a banner rather than an item, because v4 has no queue to add it to |
| Bank-alert drafts | drafts get per-draft actions and the unsure one says why | drafts join the single queue instead of living on their own screen | adds the percentage and the running count; v2 states confidence as a word; and against v3, stays a destination rather than an item on Now |
| Categories | nature becomes visible on every row | adds transaction counts, so the taxonomy can be audited | adds the amounts; v2 shows structure only; and against v3, adds amounts to v3's counts |
| Accounts | the four buttons per row are gone and the rows group into own/owe | tables rather than cards, and the surface states that it never asks for a decision | adds the 30-day change, filters, and the engine reading per row; and against v3, adds filters and the change; v3's version is deliberately quieter |
| Account editor | the EMI the terms produce leads the screen | adds where this account shows up next — a dated claim, a step in the order | same idea; the split of the first payment is shown beside it; and against v3, same as v3, minus the trail, plus the first-payment split |
| Holdings | price age moves into the chrome and an unpriced holding says so | the §11.1 disclaimer moves into the trail rather than sitting as a banner | adds the lots, which no other version reaches; and against v3, adds the lots and the KPI split |

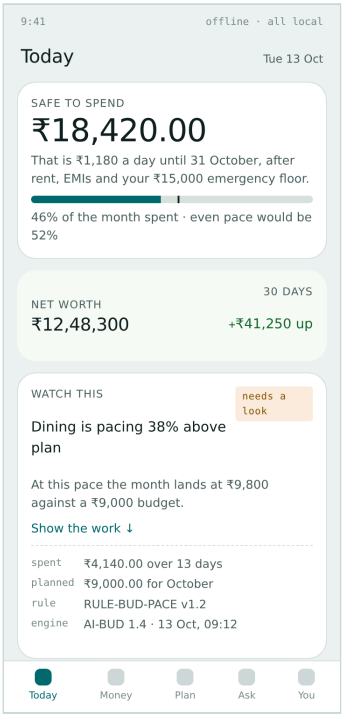
| | | | |
|---|---|---|---|
| Allocation | the split is drawn and the flag carries its rule | a bar and a table instead of a donut, and the cap becomes editable | drops the donut for a bar that holds at 200% font; and against v3, same bar-and-table; v4 puts the cap control in the sticky bar |
| Budgets | pace is drawn once per envelope instead of described three times | states its relationship to the queue: the alert lives on Now, this is where you plan | adds the right-hand projection column; and against v3, adds the projection column; v3 keeps it in the sentence |
| Goals | ten lines of prose per goal become a row, a bar and a pill | goals are claims in priority order, not progress cards | adds the emergency fund as a competing row and the monthly share as a column; and against v3, same ordering idea, drawn as a table with shares |
| Goal detail | the evidenced/claimed split is drawn and named | the gap becomes a decision with three ways to close it | adds the options table; v2 states the gap in a sentence; and against v3, the same three options, with their effect as a column |
| Emergency fund | months of cover lead instead of the target amount | the coach line becomes an item that names what it displaces | adds the inputs as a table rather than a disclosure; and against v3, same inputs; v4 shows them open, v3 keeps them in the trail |
| Purchase advisor | what the purchase leaves becomes the figure; the verdict is a chip | adds the record: whatever you choose goes into Settled | adds the sticky bar so the decision is always reachable; and against v3, same three actions; v4 pins them, v3 lists them under the checks |
| Simulators | both sides and the gap lead, with the flip rate stated | a table rather than two cards, and the assumptions become chips | adds the assumptions table naming which numbers are yours; and against v3, adds the tiles and the assumption sourcing |
| Opportunity | the score is out of what was possible and the gates are visible | adds a reminder action — leave with a condition rather than a feeling | raises the failing gate into the headline block; and against v3, same framing; v4 puts the verdict beside the score instead of below it |
| Vehicles | the prediction carries its basis and overdue is a word with a number | the overdue renewal is the same object as the item on Now | adds the logs, which v2 keeps in a disclosure; and against v3, adds the logs table; v3 keeps them in the trail |
| Assistant | a conversation layout with the composer pinned and the trace on the answer | an answer can become an item, and only the user files it | adds actions on the answer: remind me, add to buy list; and against v3, same actions; v4 types the trace as machine output |
| Settings | grouped by what each control changes, with the backup warning where it is acted on | tables, with every state visible without opening a row | adds the state column and the alerts summary; and against v3, the same tables; v4 adds export and alerts rows |
| Consents | each row gains what stops if it is withdrawn, beside the switch | each consent becomes an item with its action inline | adds the never-collected row — stating what the app does not do; and against v3, adds the never-collected row to v3's shape |
| Erase everything | the scope is counted and the caveat is the loudest thing on the screen | the count includes the settled decisions v3 itself created | same scope; v4 puts the counts in the KPI block; and against v3, same as v3, including the settled-decision count |

# Four ways · The home screen

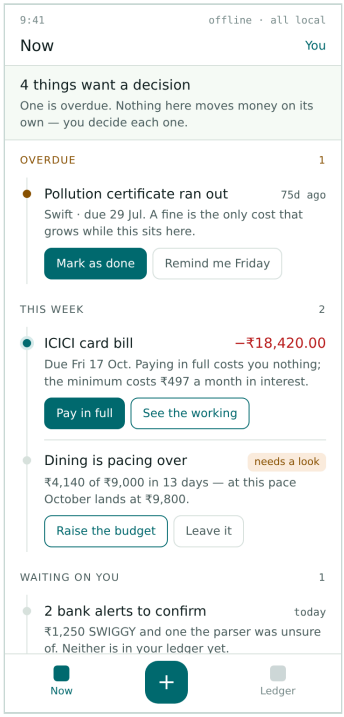
The same shipped screen, answered by three different premises. Same colour tokens throughout.



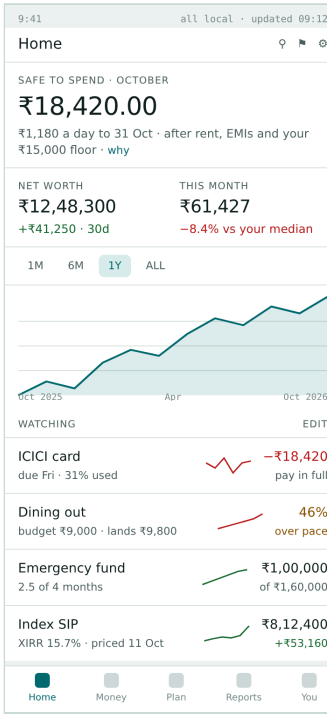

**v1** — ships today




**v2** — one answer




**v3** — the queue




**v4** — the instrument

## v1 does

Puts everything the state knows at one weight, then fourteen navigation rows — so the home screen became the app's directory.

## v2 is best at

Comprehension. One figure, one meaning line, one pace bar: a user who opens the app to check a number is done in two seconds.

## v3 is best at

Acting. Four decisions are one tap each from the first screen, and the screen can be empty — which a dashboard never is.

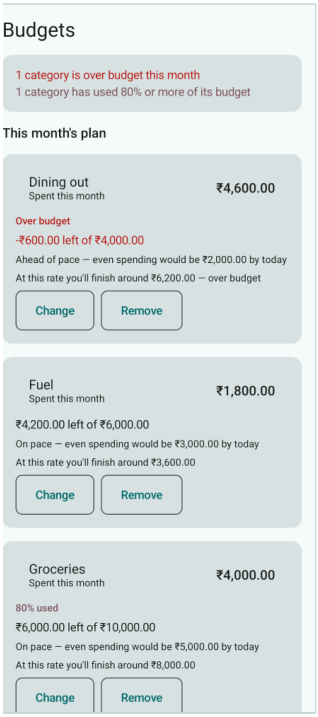
## v4 is best at

Looking. A KPI block with its window, a net-worth chart with ranges, and a watchlist the user curates — the only version where they choose what home holds.

> **The honest split:** v2 for the user who checks, v3 for the user who wants to be told, v4 for the user who wants to look. All three beat v1, which does all three jobs at one weight and therefore none of them.

# Four ways · A budget going over

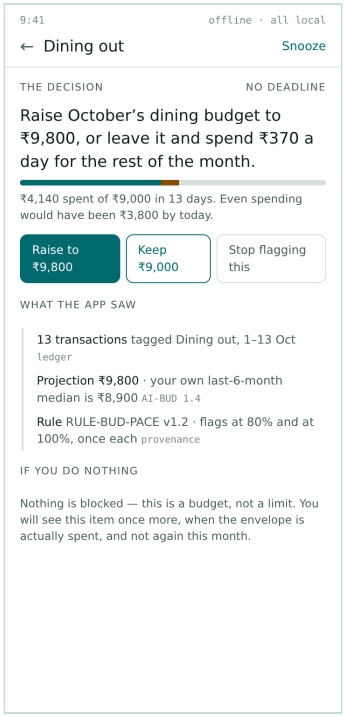
The same shipped screen, answered by three different premises. Same colour tokens throughout.



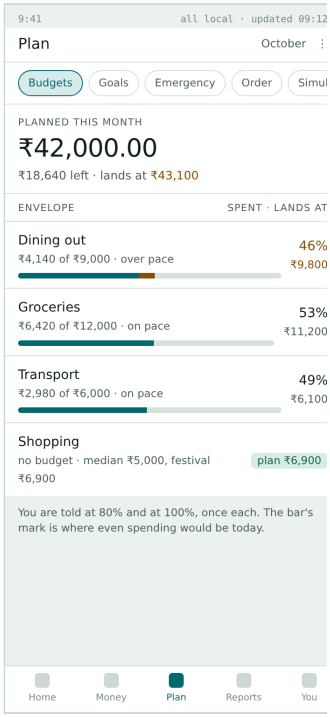

**v1** — ships today




**v2** — one answer




**v3** — the queue




**v4** — the instrument

## v1 does

Six envelopes as prose with two buttons each; the band is a word in the body text and there is no bar at all.

## v2 is best at

Planning. All six envelopes with their pace bars, so the shape of the month is visible at once.

## v3 is best at

Resolving. The envelope that broke becomes an item with two named outcomes and a *stop flagging this*.

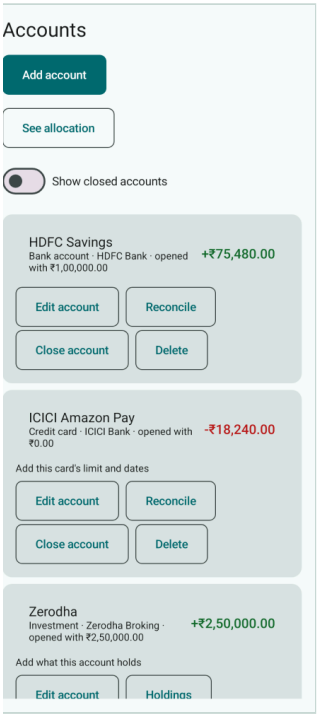
## v4 is best at

Comparing. The same pace bar plus a projection column, so the envelopes can be read as a table of outcomes.

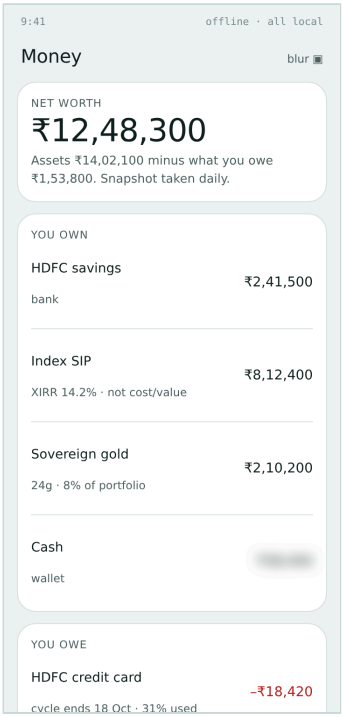
> **These are compatible.** v3's item and v4's table answer different moments — the alert and the review — and a built app could have both, which is what the closing section means by the hybrid.

# Four ways · Accounts

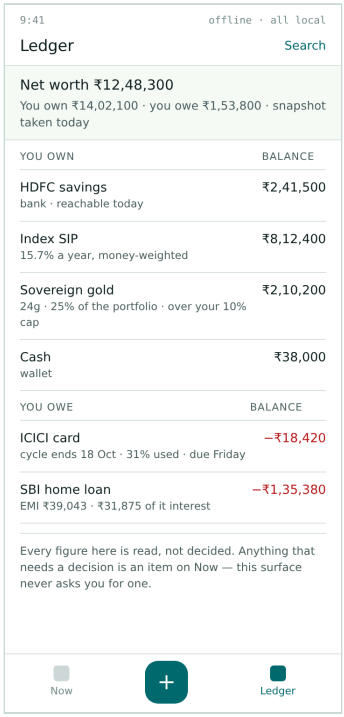
The same shipped screen, answered by three different premises. Same colour tokens throughout.



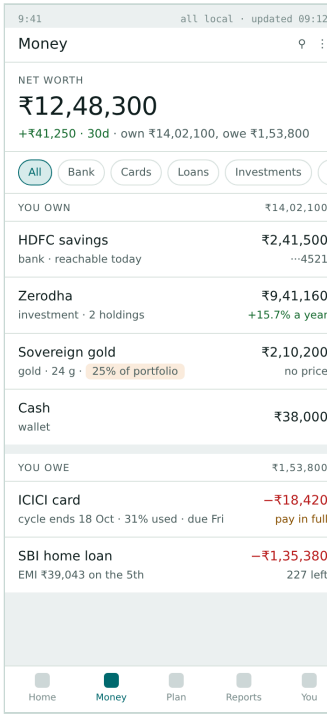

**v1** — ships today




**v2** — one answer




**v3** — the queue




**v4** — the instrument

## v1 does

Four full-width buttons on every row — Edit, Reconcile, Close, Delete — which is the densest thing in the shipped app and the clearest Material violation.

## v2 is best at

Teaching. Cards give each account room, so a card's cycle and a loan's interest split read as a small story.

## v3 is best at

Restraint. Tables that state they never ask for a decision, so the reference surface stays quiet.

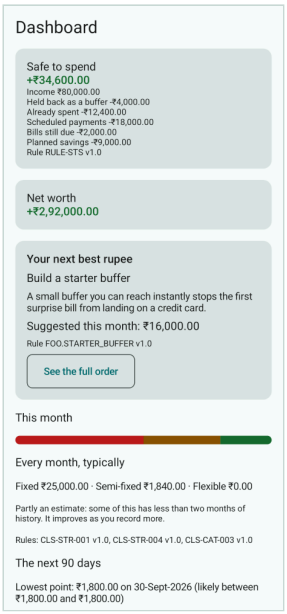
## v4 is best at

Scale. Tables plus filters plus the engine's reading per row, with net worth's 30-day change above them.

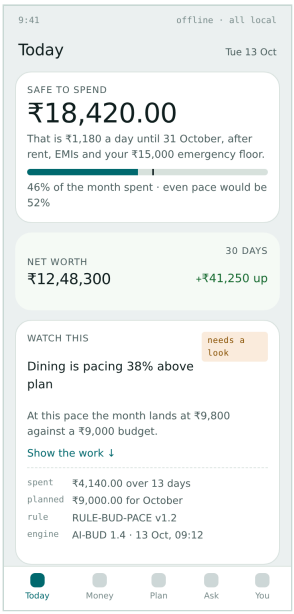
> **Scale decides this one.** At three accounts v2 reads best; at twenty, v3 and v4's tables are the only ones that still work, and v4 is the only one with filters.

# 1 · Dashboard

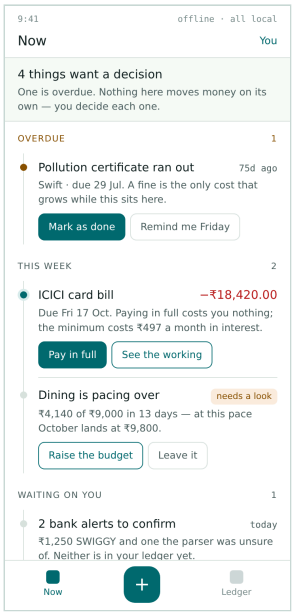
replaces the dashboard · Kite's watchlist, Groww's KPI block



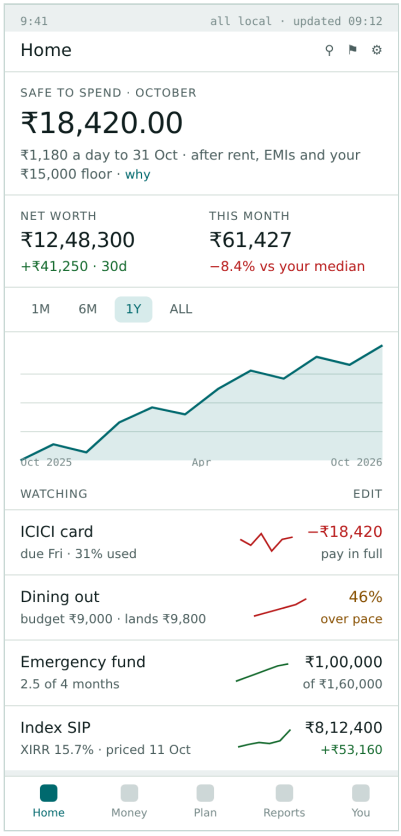

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** the derivation moves behind *why* and the screen gains a chart
> **v4 vs v2:** adds a net-worth chart with ranges and a watchlist the user pins
> **v4 vs v3:** keeps the queue off the home screen; the watchlist is curated, not computed

## What it does

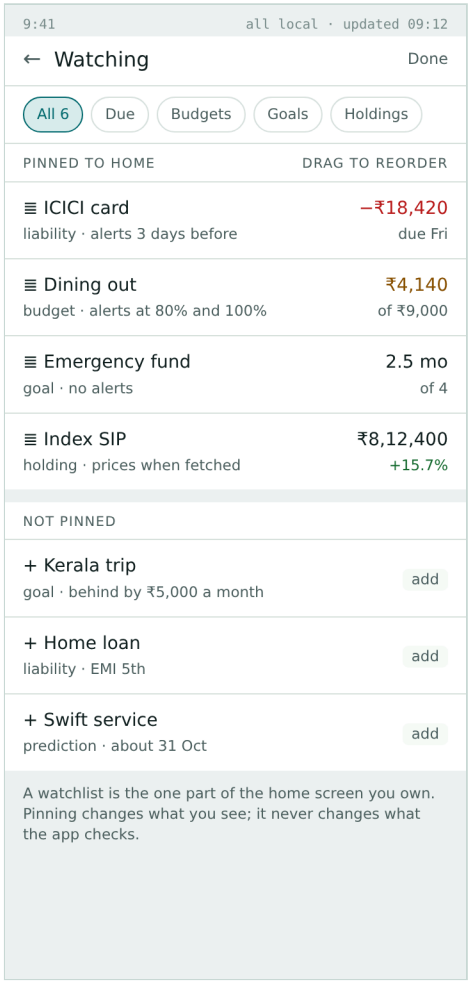
- A KPI block — safe-to-spend with its meaning line, then net worth and the month side by side, each with its change and the window it is measured over.
- A net-worth chart with 1M / 6M / 1Y / ALL, the control every one of these apps puts under a chart.
- A **watchlist**: four rows the user pinned, each with a sparkline and the figure that matters.

## Why, and what it borrows

- Kite's home is a watchlist because a trader's attention is a short list they chose. The same is true of money: the card due Friday, the budget running hot, the fund. v1 and v2 both decide for the user what the home screen holds; v4 lets them pin it.
- A time range turns one number into a trend for free — the engines already store daily net-worth snapshots, so this is a chart over data the app has and does not show.
- The sparkline is the cheapest honest chart: direction and shape, no axis, no claim of precision.

# 2 · Watchlist

new · Kite's watchlist, made editable




**v4** — new in this design

> **Against the lower versions:** No counterpart anywhere. v2 and v3 both decide what the home screen holds; this is the only design that lets the user decide.

### What it does

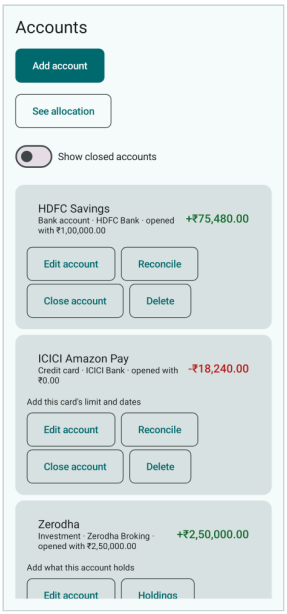
- Pin or unpin anything the engines produce: liabilities, budgets, goals, holdings, predictions.
- Each pinned row states its own alert policy — *alerts at 80% and 100%*, *3 days before due*, *no alerts*.

### Why, and what it borrows

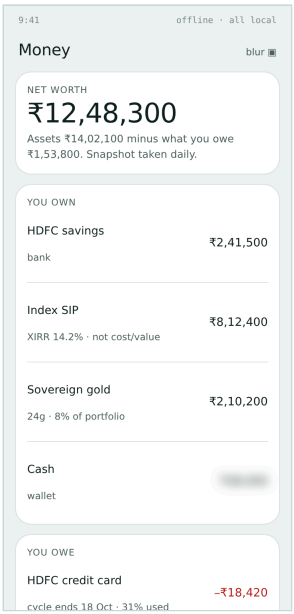
- A home screen somebody can arrange is a home screen they return to. This is the one place in v4 where the user, not an engine, decides what is important.
- Putting the alert policy on the row is how the watchlist and the notification engine (§17.2, AI-NTF) stay one idea instead of two settings screens.

# 3 · Accounts

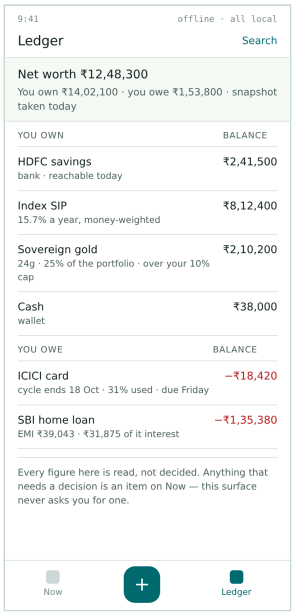
Money tab · Coin's list clarity



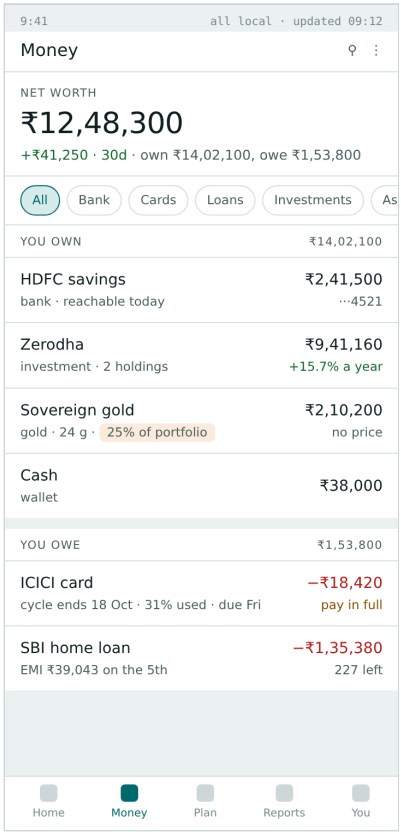

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** net worth gains its change and the list gains filter chips
> **v4 vs v2:** adds the 30-day change, filters, and the engine reading per row
> **v4 vs v3:** adds filters and the change; v3's version is deliberately quieter

## What it does

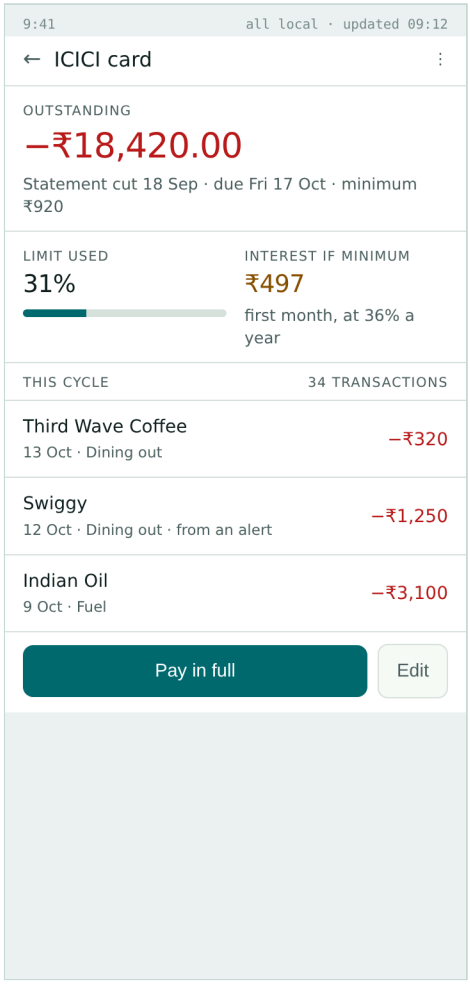
- Net worth with its 30-day change, then **You own** and **You owe** as two tables.
- Filter chips across the top: bank, cards, loans, investments, assets.
- Each row carries the engine's reading — utilisation, XIRR, a cap breach — as a sub-line.

## Why, and what it borrows

- Coin lists funds as rows with value and return and nothing else; it scales to fifty holdings without a redesign. Cards do not.
- Chips are filters, not navigation — the distinction v2's skill draws, and the one broker apps hold to.
- The sub-line is where v1's four buttons per row went: the row opens the account, and the actions live there.

# 4 · Account

new · the per-account screen v1 never had




**v4** — new in this design

> **Against the lower versions:** No counterpart in v1, v2 or v3 — none of them has a per-account screen, so none of them can price what the minimum payment costs.

## What it does

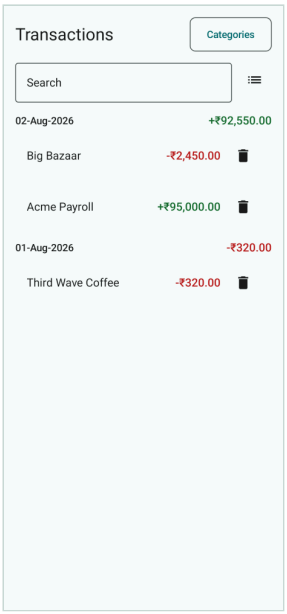
- Outstanding, cycle, limit used, and what the minimum payment would cost in interest.
- This cycle's transactions beneath it, and a sticky **Pay in full**.

## Why, and what it borrows

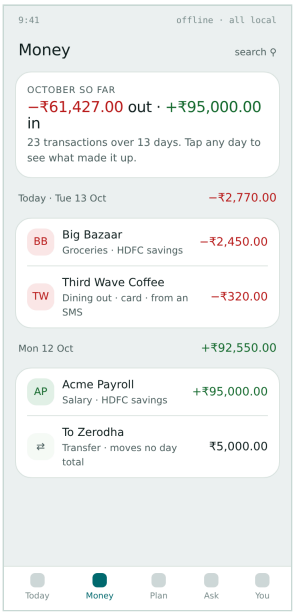
- v1 can add a credit card and compute its cycle, but there is no screen for one card — the user sees a row on a list and an item in a forecast. This is the missing destination.
- ₹497 is the number that decides this, and no screen in v1 shows it. A card screen that does not price the minimum payment is a card screen that lets you make the expensive choice by accident.

# 5 · Transactions

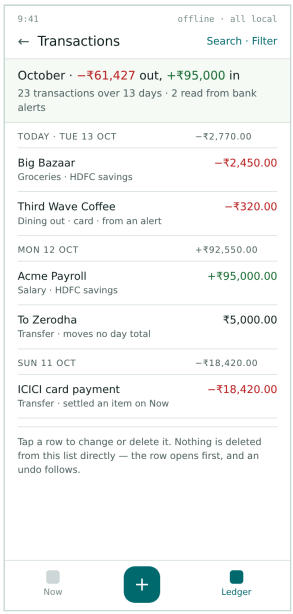
Money tab · Kite's order book



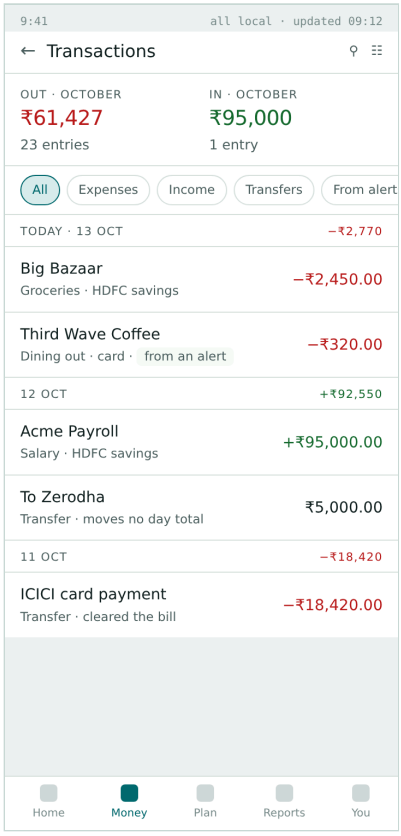

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** out/in KPIs above a day-grouped table with right-aligned amounts
> **v4 vs v2:** adds filter chips and the month's two totals above the list
> **v4 vs v3:** adds the KPIs and filters; v3 keeps the surface deliberately plain

## What it does

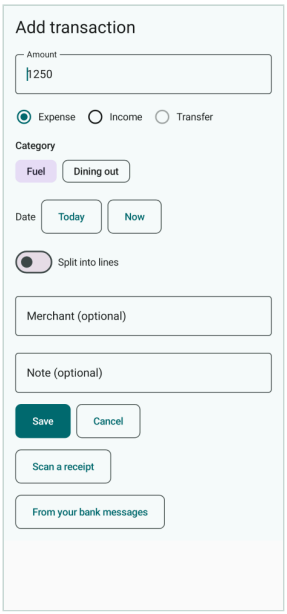
- Out and in for the month as two KPIs, then filter chips, then day-grouped rows with their day totals.
- Right-aligned tabular amounts, so the column scans vertically.
- *From an alert* is a badge on the row, not a separate screen.

## Why, and what it borrows

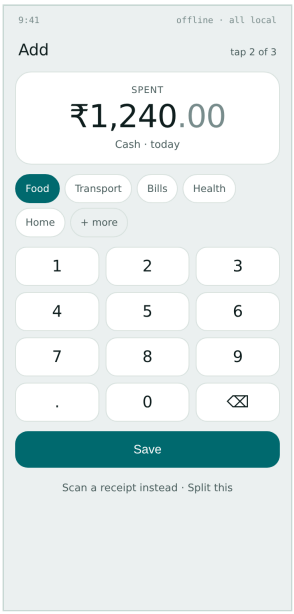
- An order book is a dense list people read fluently because the numbers line up. Money has the same shape and v1 renders it with a delete icon on every row.
- Filters belong where the list is. v1 hides them in a sheet; Kite and Groww keep them one tap above the data.

# 6 · Add transaction

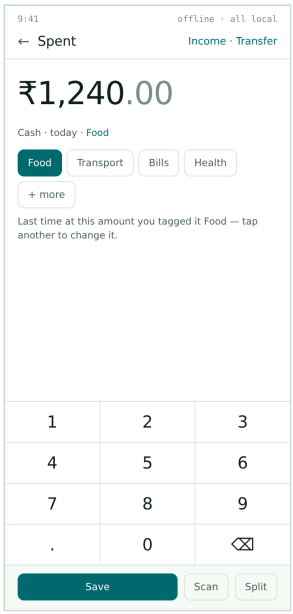
Groww's amount-first sheet



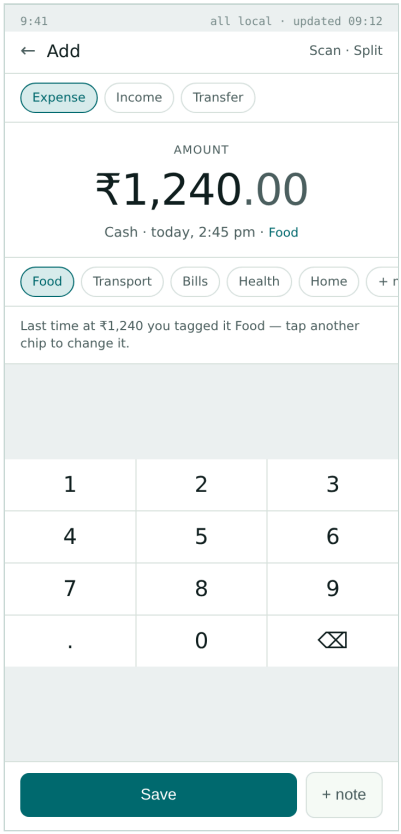

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** amount-first keypad with a sticky Save and a recall line
> **v4 vs v2:** adds the recall line — the classifier's existing output, where it saves a tap
> **v4 vs v3:** same screen; v4 keeps it as a destination rather than a bar key

## What it does

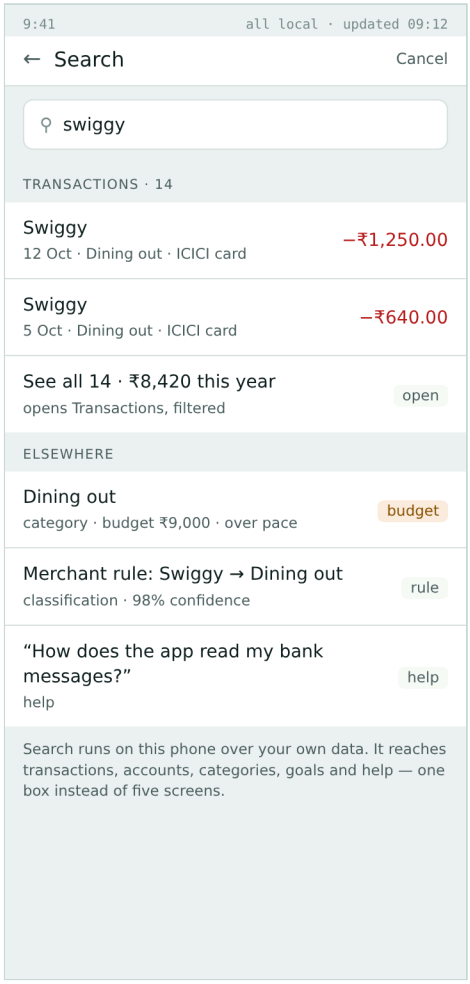
- Expense / income / transfer as chips, a large tabular amount, a keypad, and a sticky Save.
- A line recalls what this amount was tagged last time.

## Why, and what it borrows

- This is the one screen where all three ideologies agree, and v2 argued it fully: the keypad is the screen, and FR-TXN-002's three-tap budget is a layout constraint.
- The recall line is Groww's habit of showing the likely answer before asking the question. It is the classifier's existing output, placed where it saves a tap.

# 7 · Search

new · Kite's instant search




**v4** — new in this design

> **Against the lower versions:** No counterpart. v1 searches transactions only; v2 and v3 kept that, so everything else is reachable only by remembering where it lives.

## What it does

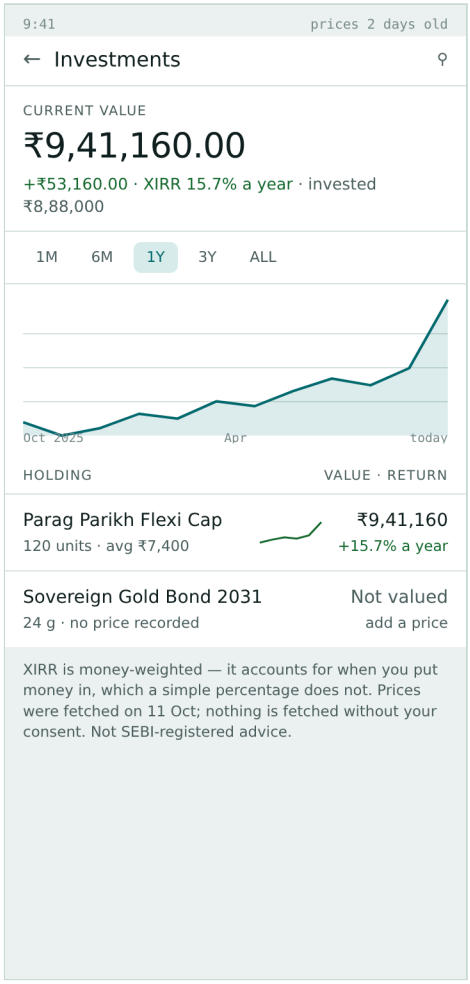
- One box over transactions, accounts, categories, merchant rules and help.
- Results grouped, with a total — *14 matches, ₹8,420 this year* — and a route into the filtered list.

## Why, and what it borrows

- Kite's search is the fastest path in the app and the reason its navigation can stay thin. v1 has search inside the transactions screen only, so everything else is reachable only by remembering where it lives.
- Searching your own ledger on-device is also the privacy argument made concrete: the fastest feature in the app sends nothing anywhere.

# 8 · Performance

new · Coin's portfolio screen




**v4** — new in this design

> **Against the lower versions:** No counterpart. v2 and v3 both show holdings and a return; neither shows the portfolio's shape over time.

## What it does

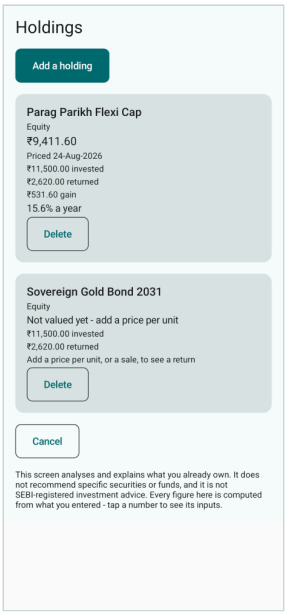
- Value, gain and XIRR in one block; a chart with 1M / 6M / 1Y / 3Y / ALL; holdings with sparklines.
- The invested figure sits beside the current value, so the gain is checkable by subtraction.

## Why, and what it borrows

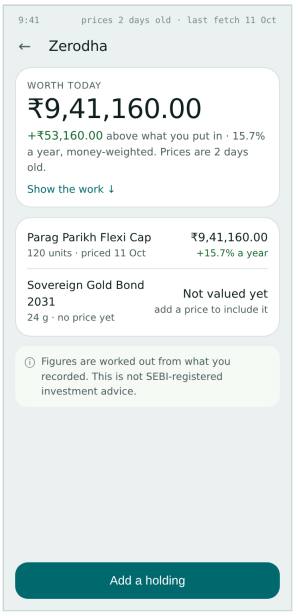
- v1 shows holdings and a return but never the portfolio's shape over time, which is the first thing anyone opening an investment screen looks for.
- Coin's discipline: the return is money-weighted and the screen says so, because a simple percentage flatters a SIP.

# 9 · Holdings

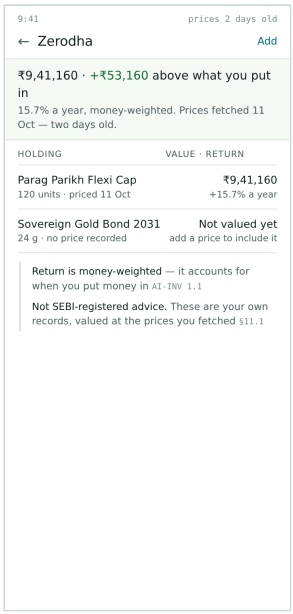
Coin's fund rows



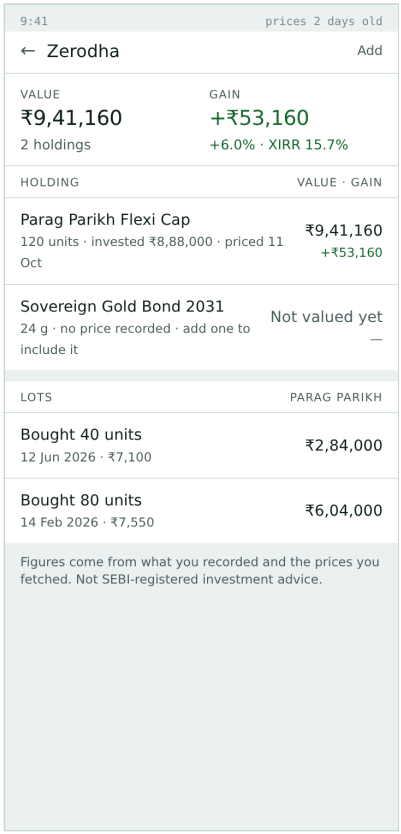

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** value and gain become two KPIs and the lots behind a holding are shown
> **v4 vs v2:** adds the lots, which no other version reaches
> **v4 vs v3:** adds the lots and the KPI split

## What it does

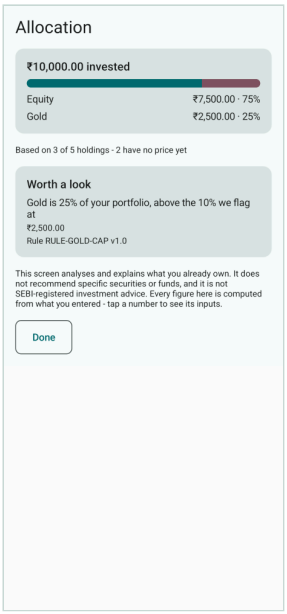
- Value and gain as two KPIs, then holdings, then the lots that make up one holding.
- An unpriced holding says so and asks for a price rather than showing a zero.

## Why, and what it borrows

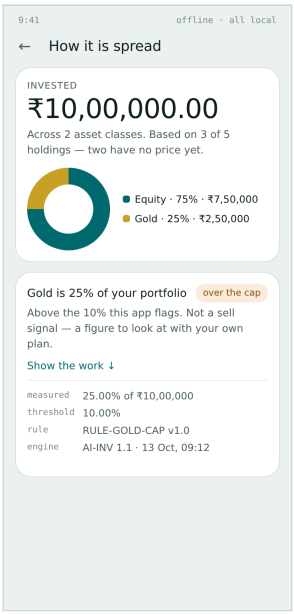
- Lots are what let a user check a cost basis, and v1 keeps them one level deeper than anyone looks.
- Price age in the status line, because it qualifies every figure on the screen (P-03).

# 10 · Allocation

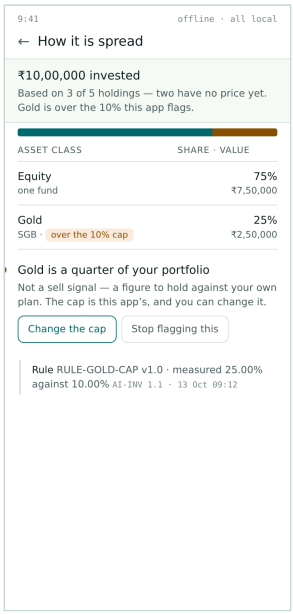
a proportion bar, not a donut



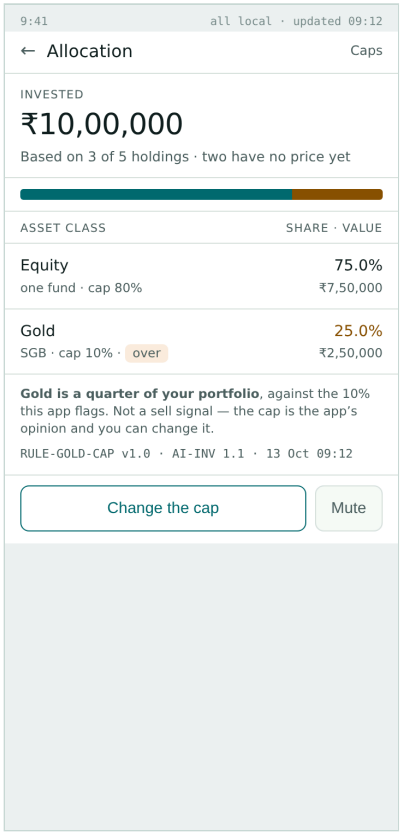

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** a proportion bar with a table of shares and values, and a sticky action
> **v4 vs v2:** drops the donut for a bar that holds at 200% font
> **v4 vs v3:** same bar-and-table; v4 puts the cap control in the sticky bar

## What it does

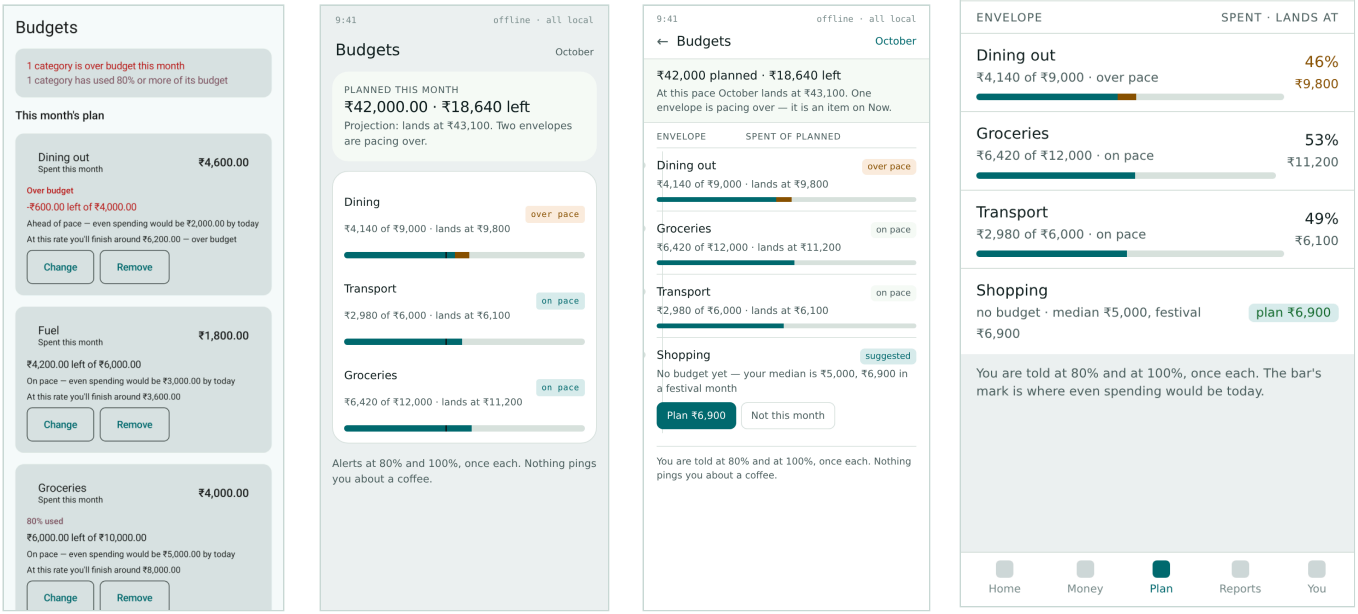
- One stacked bar, a table of shares and values, and the cap breach stated under it with its rule.
- A sticky **Change the cap**, because the threshold is the app's opinion.

## Why, and what it borrows

- Broker apps draw allocation as a bar with a table, not a pie, because the numbers are the point and a bar holds its meaning at 200% font.
- Handing over the cap is P-07 applied to a threshold: the app may flag, the user decides what counts as too much.

# 11 · Budgets

Plan tab · Groww's progress rows




**v1** **v2** **v3** **v4** — the instrument panel

> **v4 vs v1:** adds the projection as a column and the suggestion into the same list
> **v4 vs v2:** adds the right-hand projection column
> **v4 vs v3:** adds the projection column; v3 keeps it in the sentence

## What it does

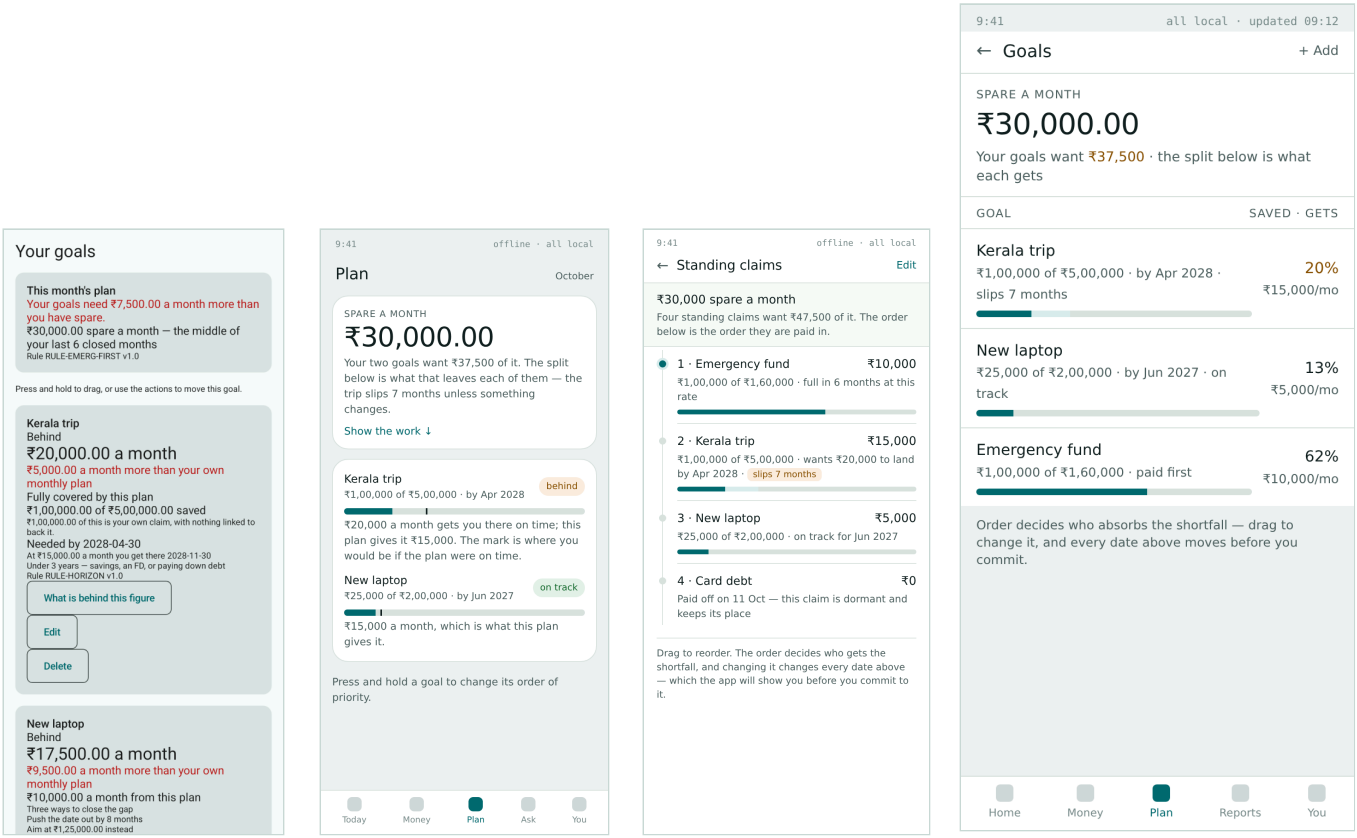
- Planned, left, and where the month lands as one KPI; each envelope a row with a pace bar and its projection.
- An unbudgeted category appears in the same list as a suggestion with its own action.

## Why, and what it borrows

- v2's pace bar, unchanged — it was right. What v4 adds is the projection in the right-hand column, so the list can be read as a table of outcomes.
- Suggestions in the list rather than on a card: budgeting happens in one place, in one idiom.

# 12 · Goals

Plan tab · progress with a split




**v1** **v2** **v3** **v4** — the instrument panel

> **v4 vs v1:** each goal shows what it gets against what it wants, with the fund in the same table
> **v4 vs v2:** adds the emergency fund as a competing row and the monthly share as a column
> **v4 vs v3:** same ordering idea, drawn as a table with shares

## What it does

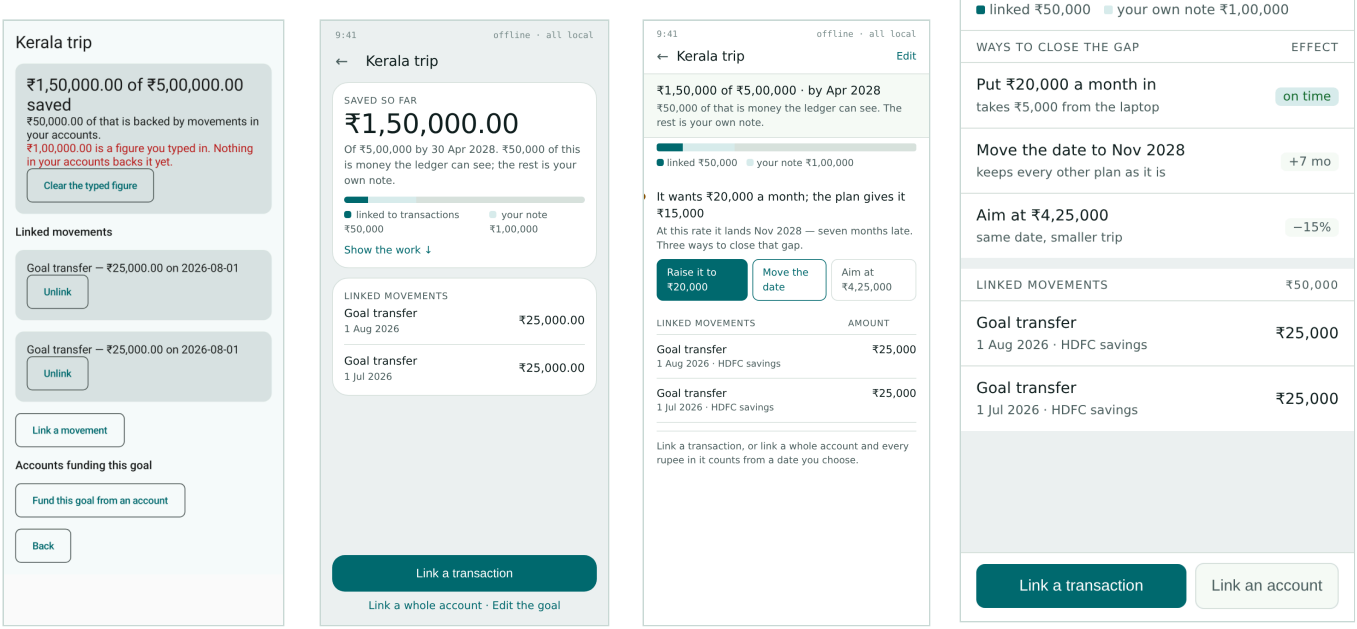
- Spare per month as the KPI with what the goals want of it; each goal a row with its bar, its share and its date.
- The emergency fund sits in the same list, marked as paid first.

## Why, and what it borrows

- Showing the emergency fund as a goal is the honest version of §36: it competes for the same rupee, so it belongs in the same table.
- Right-aligned *₹15,000/mo* against *20%* lets the user compare what each goal gets without reading a sentence per goal.

# 13 · Goal detail

the gap as a table of options




**v1** **v2** **v3** **v4** — the instrument panel

**v4 vs v1:** the three ways become a table with their effect
**v4 vs v2:** adds the options table; v2 states the gap in a sentence
**v4 vs v3:** the same three options, with their effect as a column

## What it does

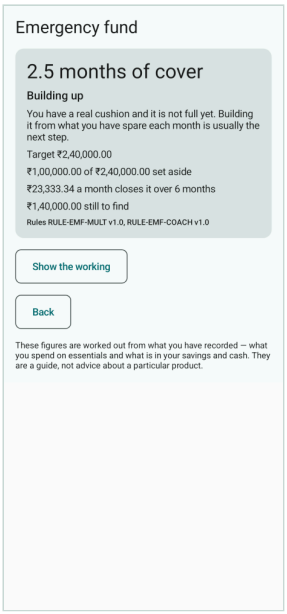
- Saved, target and date in the KPI; a split bar for linked money versus the user's own note.
- **Ways to close the gap** as three rows with their effect, then the linked movements.

## Why, and what it borrows

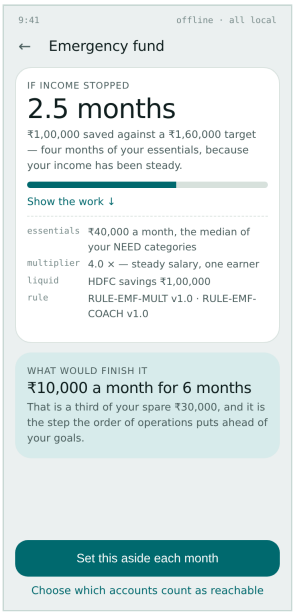
- v1 states the gap in prose; the three ways to close it are what a person actually chooses between, so they become a table with consequences.
- The split bar keeps issue 7.4's distinction, with both halves named in words.

# 14 · Emergency fund

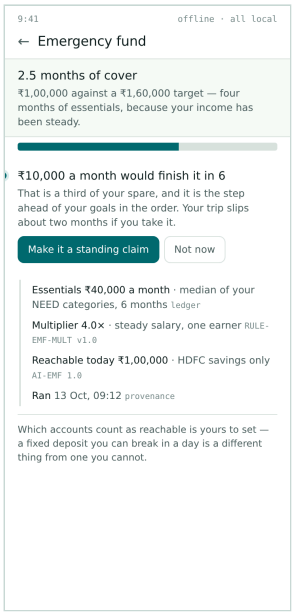
months first, then the arithmetic



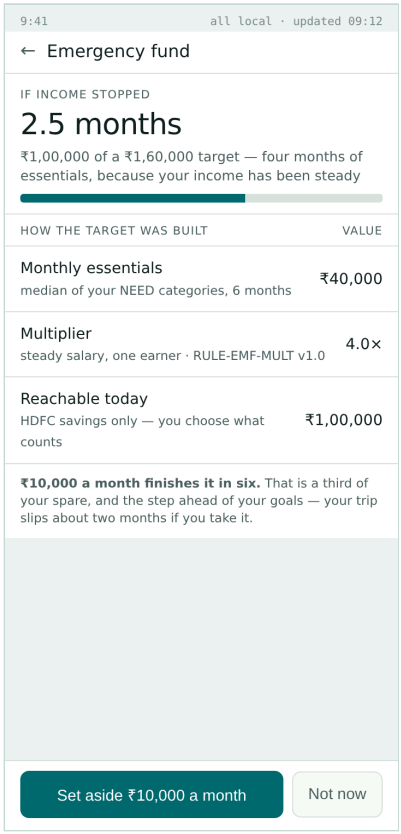

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** the three inputs behind the target become a table
> **v4 vs v2:** adds the inputs as a table rather than a disclosure
> **v4 vs v3:** same inputs; v4 shows them open, v3 keeps them in the trail

## What it does

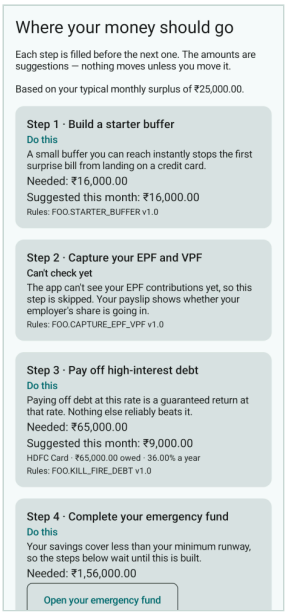
- Months of cover as the figure; the three inputs that built the target as a table.
- A sticky action that turns the coach line into a standing claim, and says what it displaces.

## Why, and what it borrows

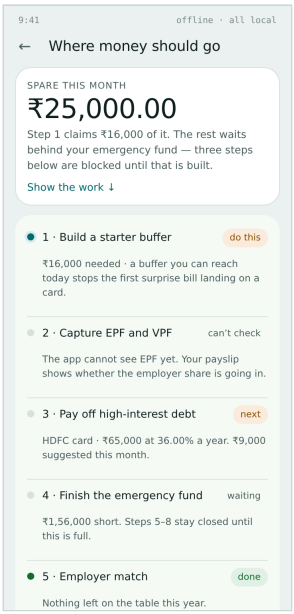
- §10.1 wants every figure linked to its evidence. A table of inputs is the densest honest form of that, and it reads the same way as every other list in v4.
- Naming the cost — the trip slips two months — is what makes the suggestion acceptable rather than glib.

# 15 · Order of operations

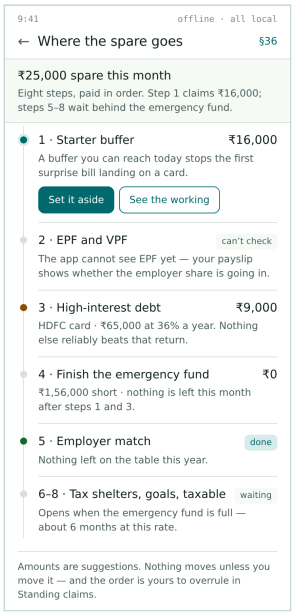
§36 as a ranked table



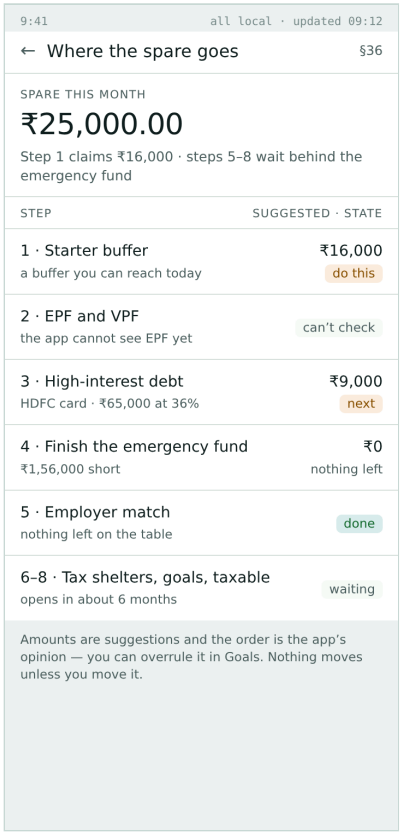

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** eight steps become a ranked table with amounts and state badges
> **v4 vs v2:** denser: the suggested amount is a column, not a sentence
> **v4 vs v3:** a table rather than a rail, because nothing here is time-ordered

## What it does

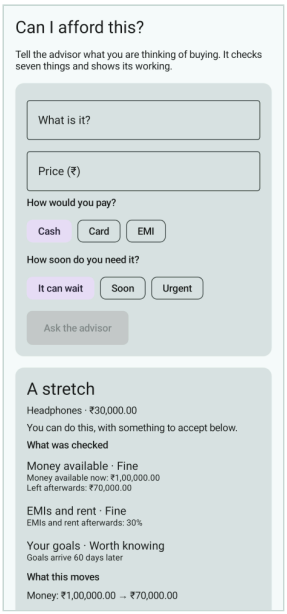
- Spare for the month, then eight steps as rows with the suggested amount and a state badge.

## Why, and what it borrows

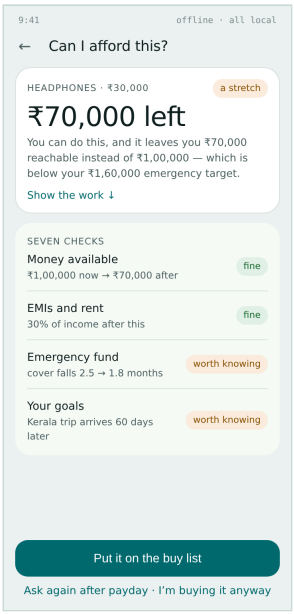
- A waterfall is a ranking, and a ranking is a table. Kite's order book taught users to read exactly this shape.
- The badge carries the word; the amount carries the claim. Nothing is conveyed by colour alone.

# 16 · Purchase advisor

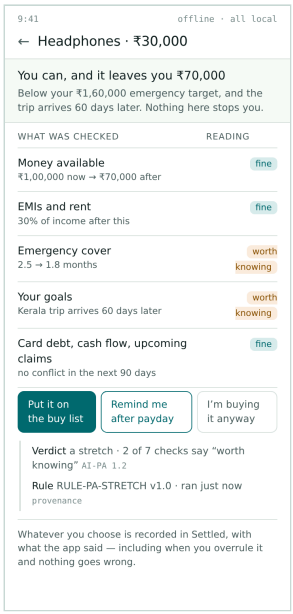
the verdict as a badge, the money as the figure



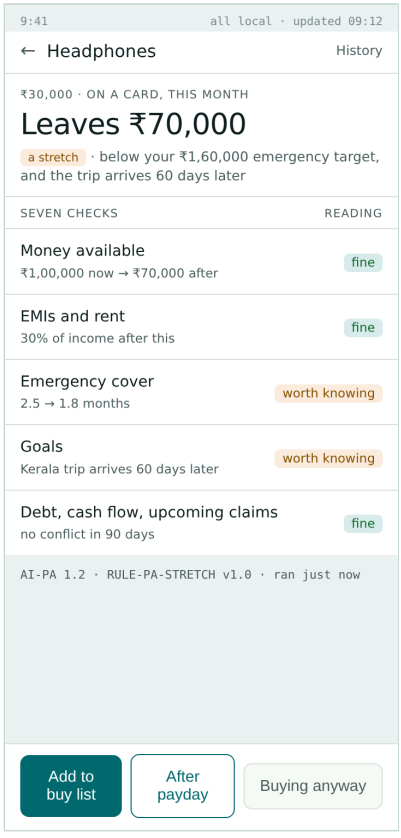

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** the seven checks become a table and the three actions a sticky bar
> **v4 vs v2:** adds the sticky bar so the decision is always reachable
> **v4 vs v3:** same three actions; v4 pins them, v3 lists them under the checks

## What it does

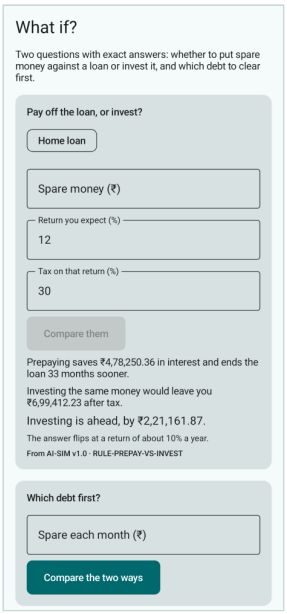
- What the purchase leaves is the figure; the verdict is a badge beside the item and price.
- Seven checks as a table with readings; three actions in the sticky bar, including *buying anyway*.

## Why, and what it borrows

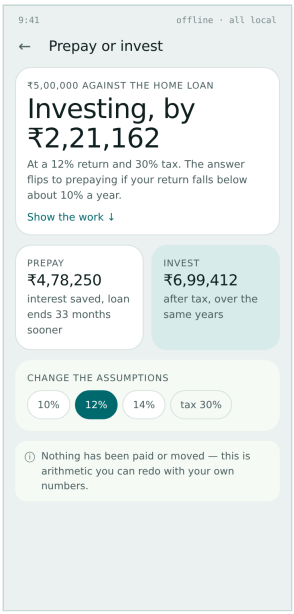
- Groww puts the decision in a sticky bar so it is always reachable and always the same shape. Here the decision is the user's, which is why their override is one of the three buttons.
- A verdict in a badge and the consequence in the figure is the inversion v1 needs: the word is small, the money is large.

# 17 · Simulators

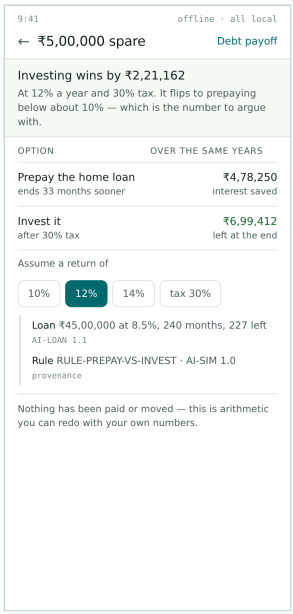
two tiles and the flip rate



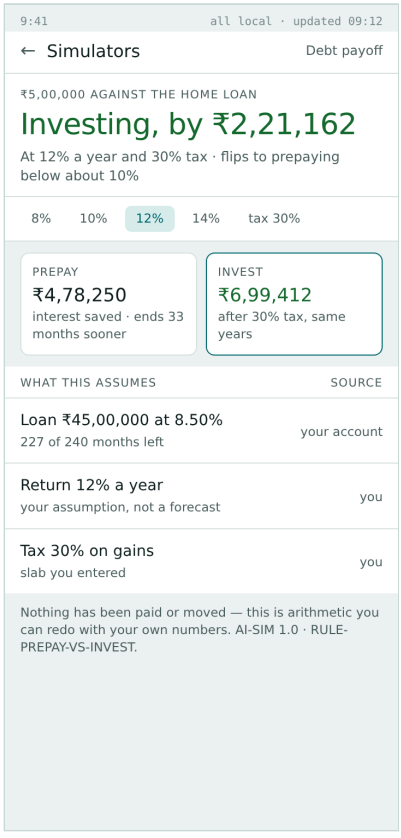

v1




v2




v3




**v4** — the instrument panel

**v4 vs v1:** two tiles plus a table of what the comparison assumes
**v4 vs v2:** adds the assumptions table naming which numbers are yours
**v4 vs v3:** adds the tiles and the assumption sourcing

## What it does

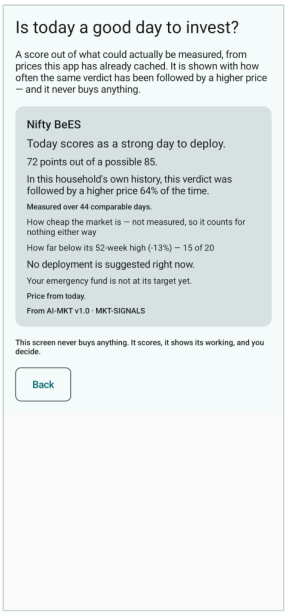
- The answer and its size as the KPI, then the two options as tiles, then what the comparison assumes.
- Assumptions as a range control, so the answer can be stress-tested in one tap.

## Why, and what it borrows

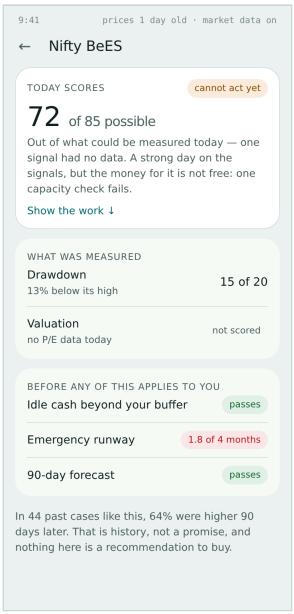
- Kite and Groww both make assumptions adjustable in place; a comparison you cannot re-run is an opinion.
- The assumptions table names which numbers are the app's and which are the user's — a distinction P-03 cares about.

# 18 · Opportunity

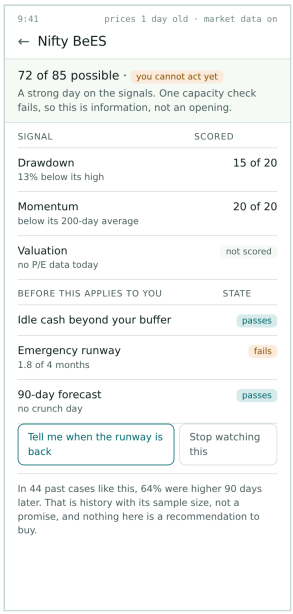
scored, with the capacity gate above it



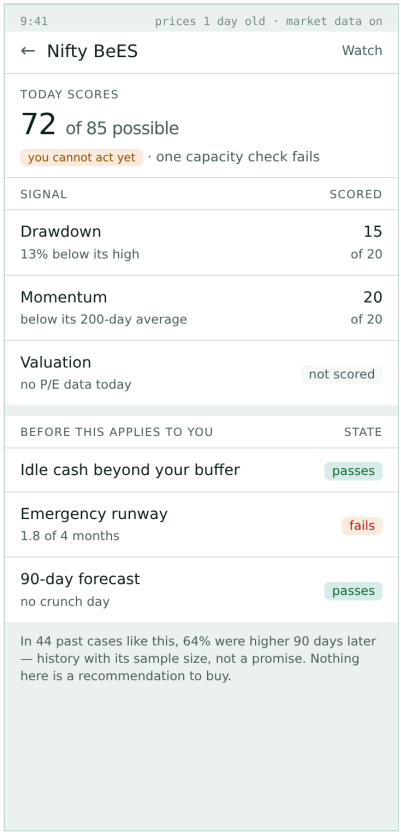

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** the capacity verdict sits in the KPI block beside the score
> **v4 vs v2:** raises the failing gate into the headline block
> **v4 vs v3:** same framing; v4 puts the verdict beside the score instead of below it

## What it does

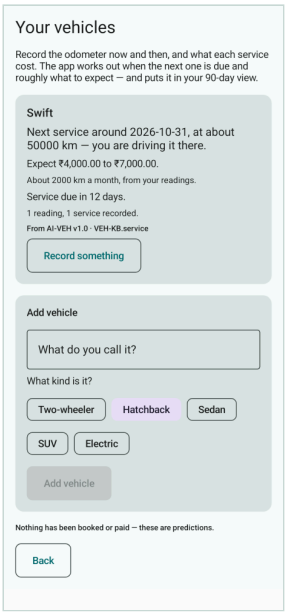
- Score out of what was possible, with the capacity verdict as a badge in the same block.
- Signals and capacity checks as two tables; the hit rate stated with its sample size.

## Why, and what it borrows

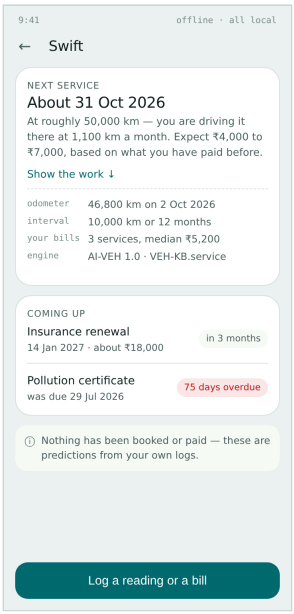
- This screen looks most like a trading app and is the one that must least behave like one. The failing capacity check is in the KPI block so it cannot be scrolled past.
- “72 of 85 possible” refuses the false precision of a score out of 100 when a signal had no data.

# 19 · Vehicles

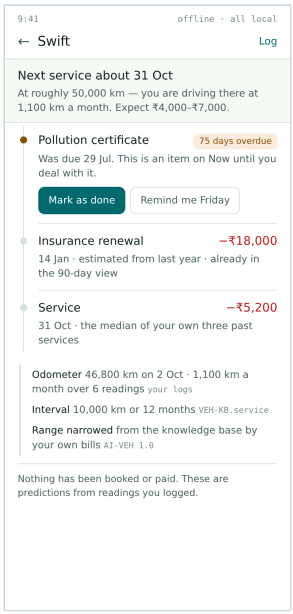
predictions as dated rows



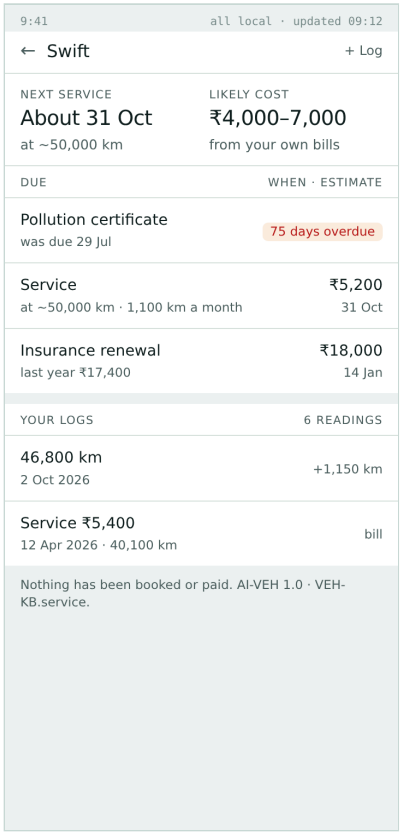

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** next service and cost become two KPIs and your logs are shown as evidence
> **v4 vs v2:** adds the logs, which v2 keeps in a disclosure
> **v4 vs v3:** adds the logs table; v3 keeps them in the trail

## What it does

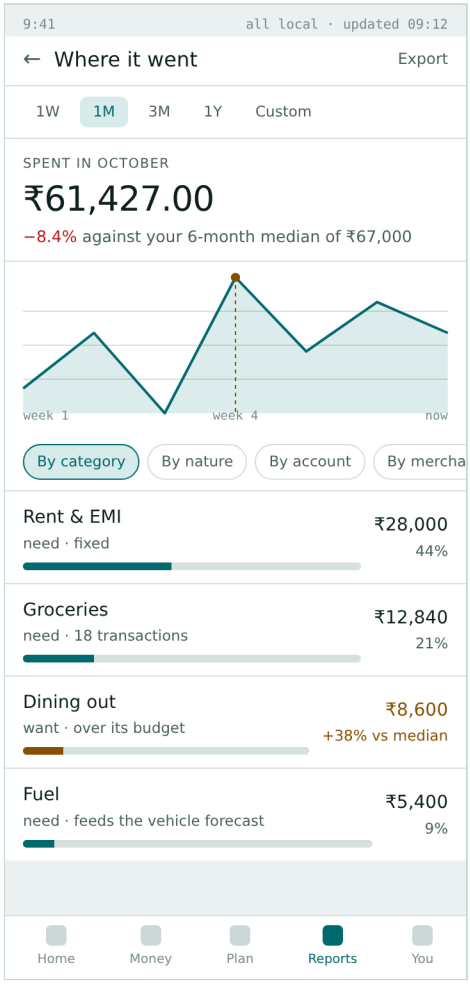
- Next service and its likely cost as two KPIs; what is due as a table; your own logs beneath.

## Why, and what it borrows

- The logs are shown because they are the evidence: a prediction from readings the user entered is arguable, one from nowhere is not.
- “75 days overdue” is a badge with a number, never a red dot.

# 20 · Analysis

new · Groww's analysis tab




**v4** — new in this design

> **Against the lower versions:** No counterpart. All three lower versions can compute every figure here and none of them puts the answer in a picture.

## What it does

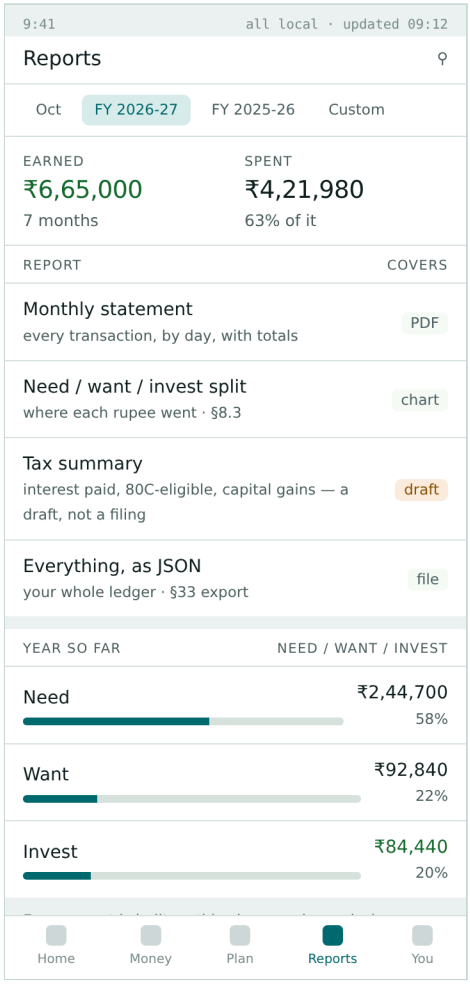
- A range control, spend against the user's own median, a chart with the festival week marked, and a breakdown by category, nature, account or merchant.

## Why, and what it borrows

- v1 can compute every one of these figures and shows none of them together. “Where did it go?” is the second question any finance app is asked and the one it currently cannot answer in a picture.
- Comparing against the user's own median rather than a benchmark keeps the app out of judging how somebody lives.

# 21 · Reports

new · the Reports tab




**v4** — new in this design

> **Against the lower versions:** No counterpart. §33's export sits in settings in all three; none of them produces a statement a person can read.

## What it does

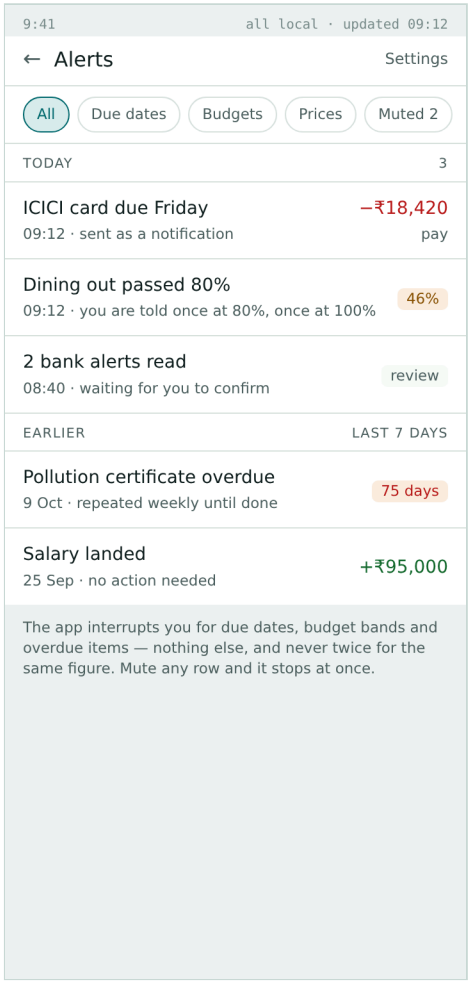
- Monthly statement, the need/want/invest split, a tax summary marked *draft*, and the full JSON export.
- A year-to-date split as three bars with their shares.

## Why, and what it borrows

- Groww and Coin both earn trust by making statements easy to produce. v1 has §33's export buried in settings and no human-readable statement at all.
- The tax summary is labelled a draft because the app is not a filing tool, and saying so is the difference between useful and reckless.

# 22 · Alerts

new · the notification inbox




**v4** — new in this design

> **Against the lower versions:** Closest in v3, which turns alerts into queue items on Now and keeps the taken ones in Settled. v4 keeps the inbox separate so the home screen stays an instrument.

## What it does

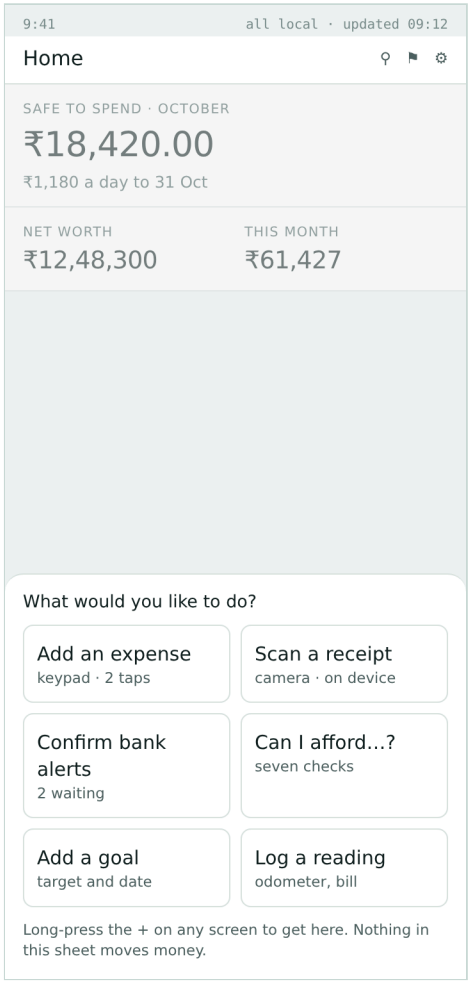
- Every interruption the app has made, grouped by day, with what it was about and whether it acted.
- Filters including **Muted**, and the policy stated at the foot.

## Why, and what it borrows

- The app has a notification engine (§17.2, AI-NTF) and no screen where its decisions can be reviewed. An alert you missed is currently gone.
- An inbox is also the honest place to mute: the user sees what they would stop receiving before they stop it.

# 23 · Quickactions

new · the long-press sheet




**v4** — new in this design

> **Against the lower versions:** Closest in v3's capture key, which reaches one action. This sheet reaches six, none of which moves money.

## What it does

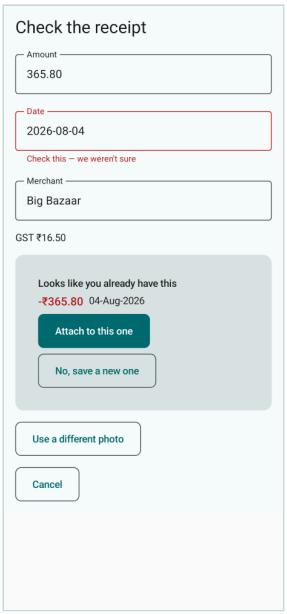
- Six actions from anywhere: add, scan, confirm drafts, affordability, add a goal, log a reading.
- A line stating that nothing in the sheet moves money.

## Why, and what it borrows

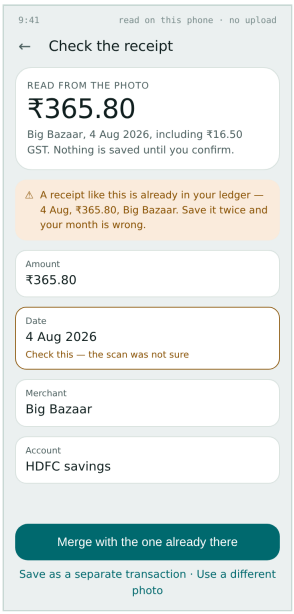
- Every one of these apps has a one-gesture path to its primary action. Here the primary action is capture, and the sheet makes the rarer ones reachable without a hunt.
- The disclaimer belongs on the sheet because this is the one surface that looks like a payments menu.

# 24 · Receipt review

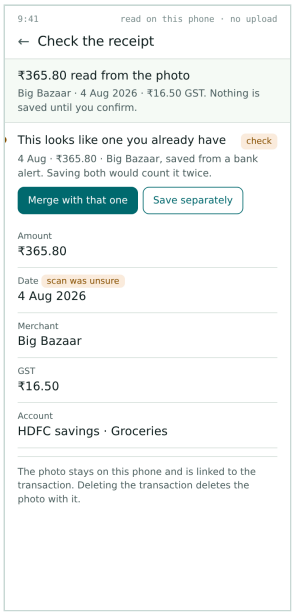
the duplicate as a banner, merge as the primary



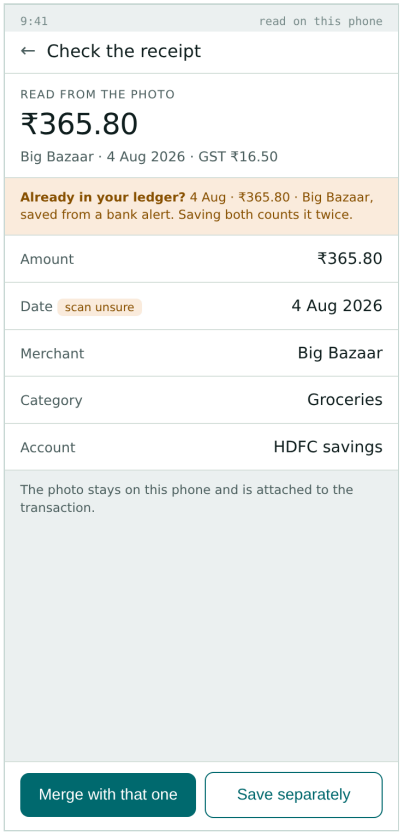

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** the read total is a KPI and the duplicate is a banner in the warning tone
> **v4 vs v2:** same decision, in the denser chrome, with the fields as rows
> **v4 vs v3:** a banner rather than an item, because v4 has no queue to add it to

## What it does

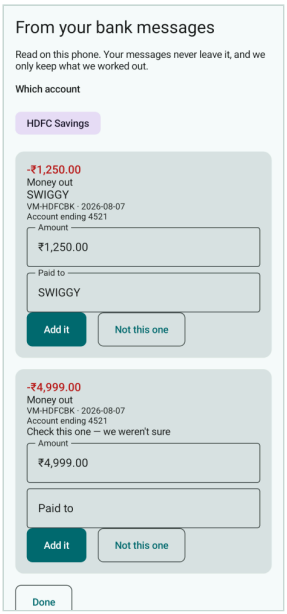
- The read total as the KPI, the duplicate warning in the warning tone above the fields, and the fields as rows.
- Merge is the sticky primary; save separately is the alternative.

## Why, and what it borrows

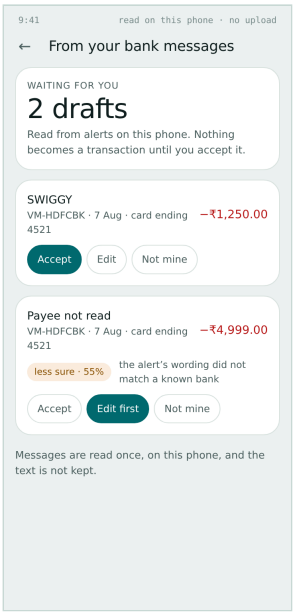
- Defaulting to the safe option is error prevention, and the expensive mistake here is counting a receipt twice.
- A field the scan was unsure of carries a badge, so the flag is a word as well as a colour.

# 25 · Bank-alert drafts

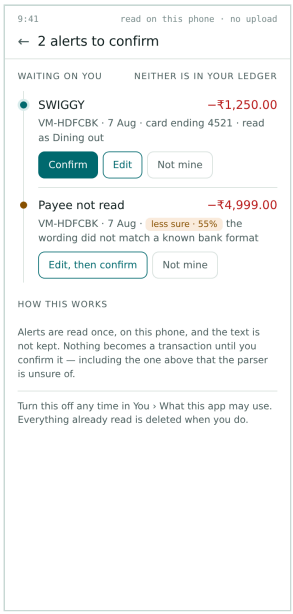
drafts as rows with per-draft actions



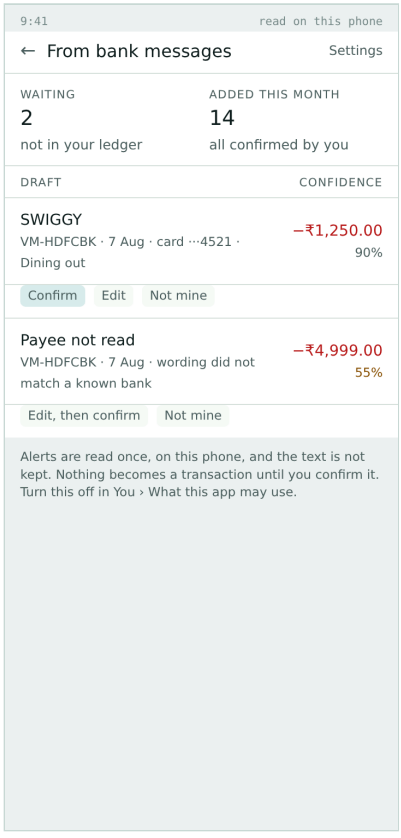

v1




v2




v3




**v4** — the instrument panel

**v4 vs v1:** confidence becomes a percentage and the month's accepted count is shown
**v4 vs v2:** adds the percentage and the running count; v2 states confidence as a word
**v4 vs v3:** stays a destination rather than an item on Now

## What it does

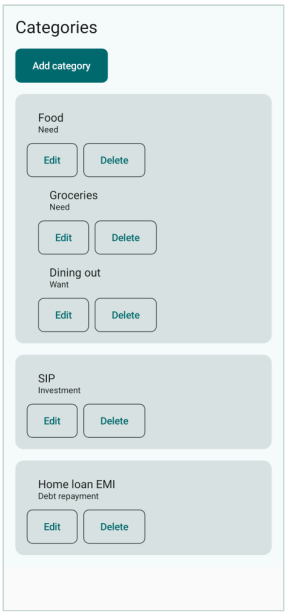
- Waiting and added-this-month as two KPIs; each draft a row with its confidence as a number.
- Actions sit under each draft — confirm, edit, not mine.

## Why, and what it borrows

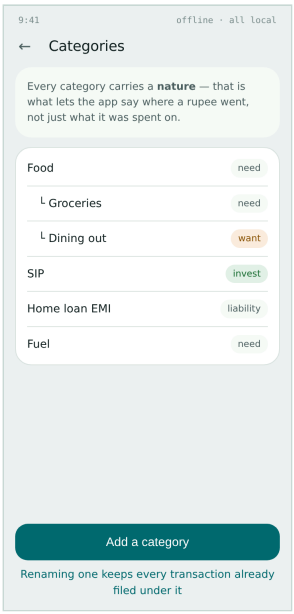
- A percentage is more honest than a word for a parser's confidence, and this audience reads percentages all day.
- Per draft, never bulk: the review is the feature, and P-07 means the app may not import on the user's behalf.

# 26 · Categories

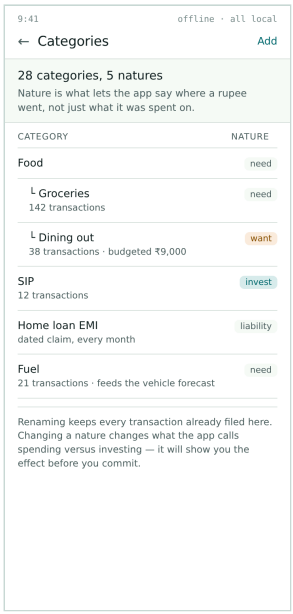
a table with this month's spend



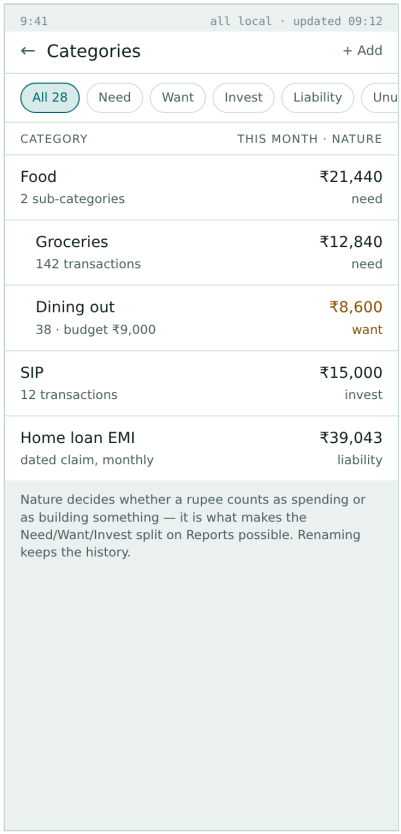

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** adds this month's spend per category and filter chips by nature
> **v4 vs v2:** adds the amounts; v2 shows structure only
> **v4 vs v3:** adds amounts to v3's counts

## What it does

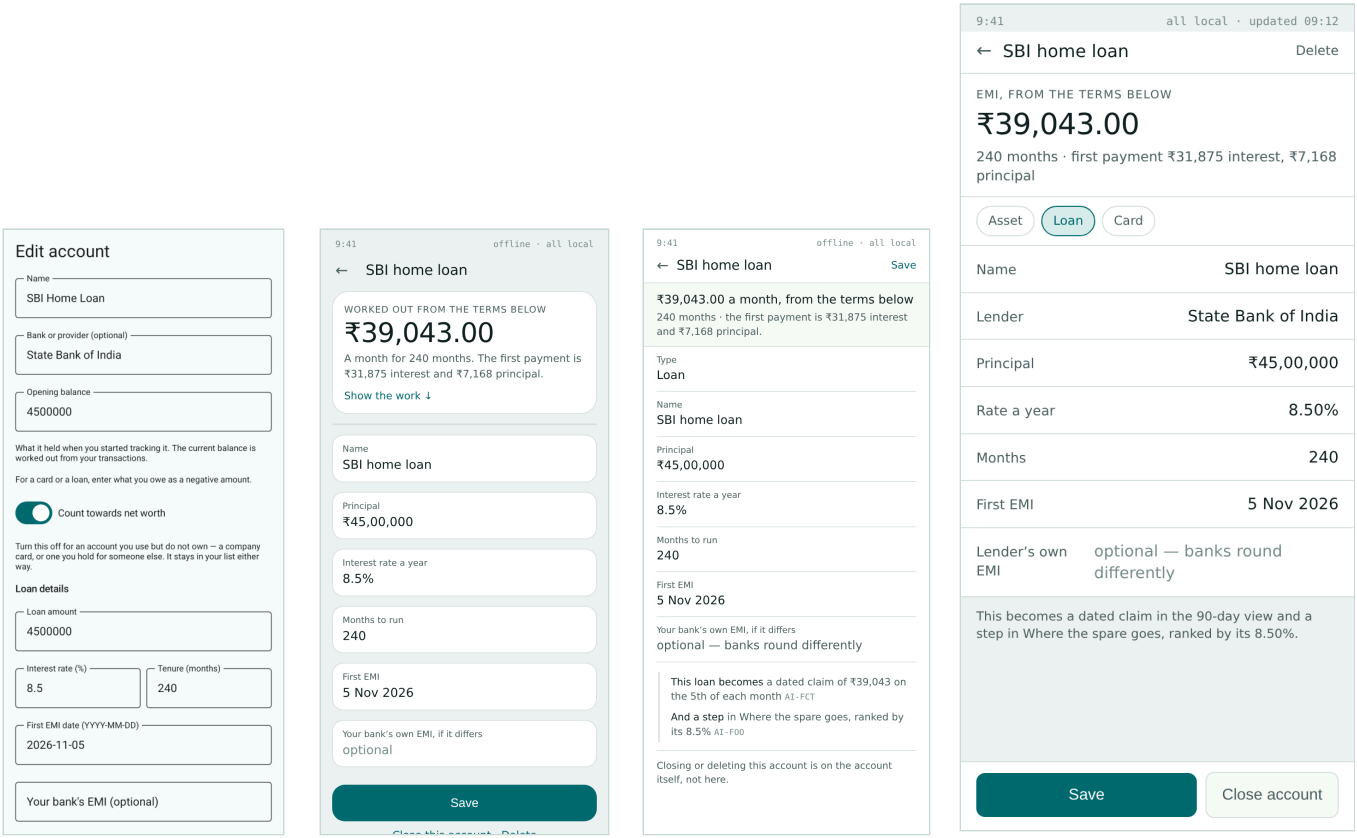
- Each category with its spend this month, its nature and its transaction count; filter chips by nature.

## Why, and what it borrows

- Counts and amounts make the taxonomy auditable — a category with two transactions is a candidate for merging, which is invisible in v1.
- The nature column is what makes the Reports split possible, so it is shown where it is set.

# 27 · Account editor

the EMI the terms produce




**v1** **v2** **v3** **v4** — the instrument panel

> **v4 vs v1:** the derived EMI is the KPI and the terms are rows
> **v4 vs v2:** same idea; the split of the first payment is shown beside it
> **v4 vs v3:** same as v3, minus the trail, plus the first-payment split

## What it does

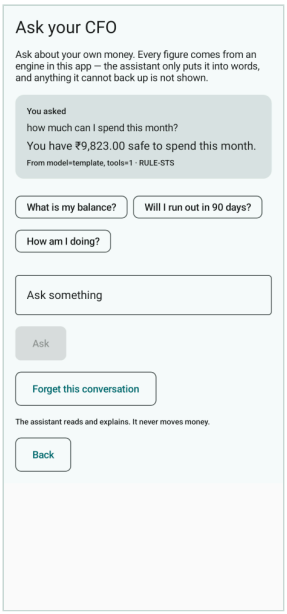
- The derived EMI as the KPI with its first-payment split; the terms as rows; a line saying where this account shows up next.

## Why, and what it borrows

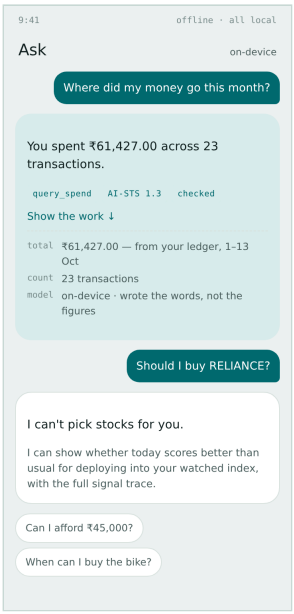
- A form whose output is a schedule should show the schedule — the user can tell a wrong rate from a right one by looking at the EMI.
- Saying it becomes a dated claim and a step in the order is how a form teaches the model of the app.

# 28 · Assistant

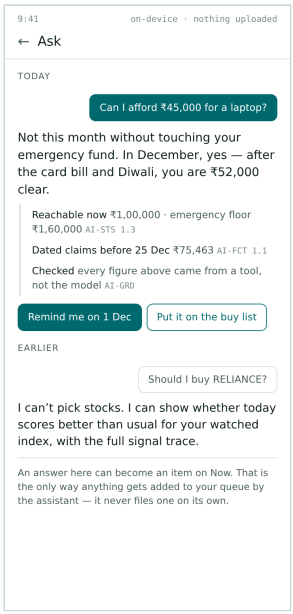
answers that carry their tools



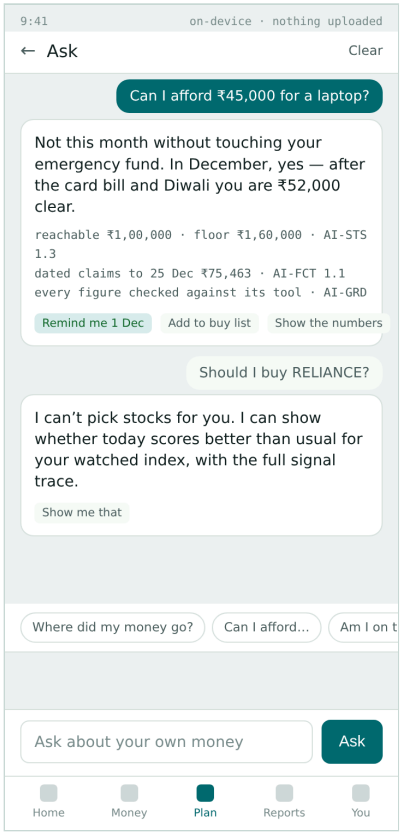

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** the trace is monospace and the answer carries actions
> **v4 vs v2:** adds actions on the answer: remind me, add to buy list
> **v4 vs v3:** same actions; v4 types the trace as machine output

## What it does

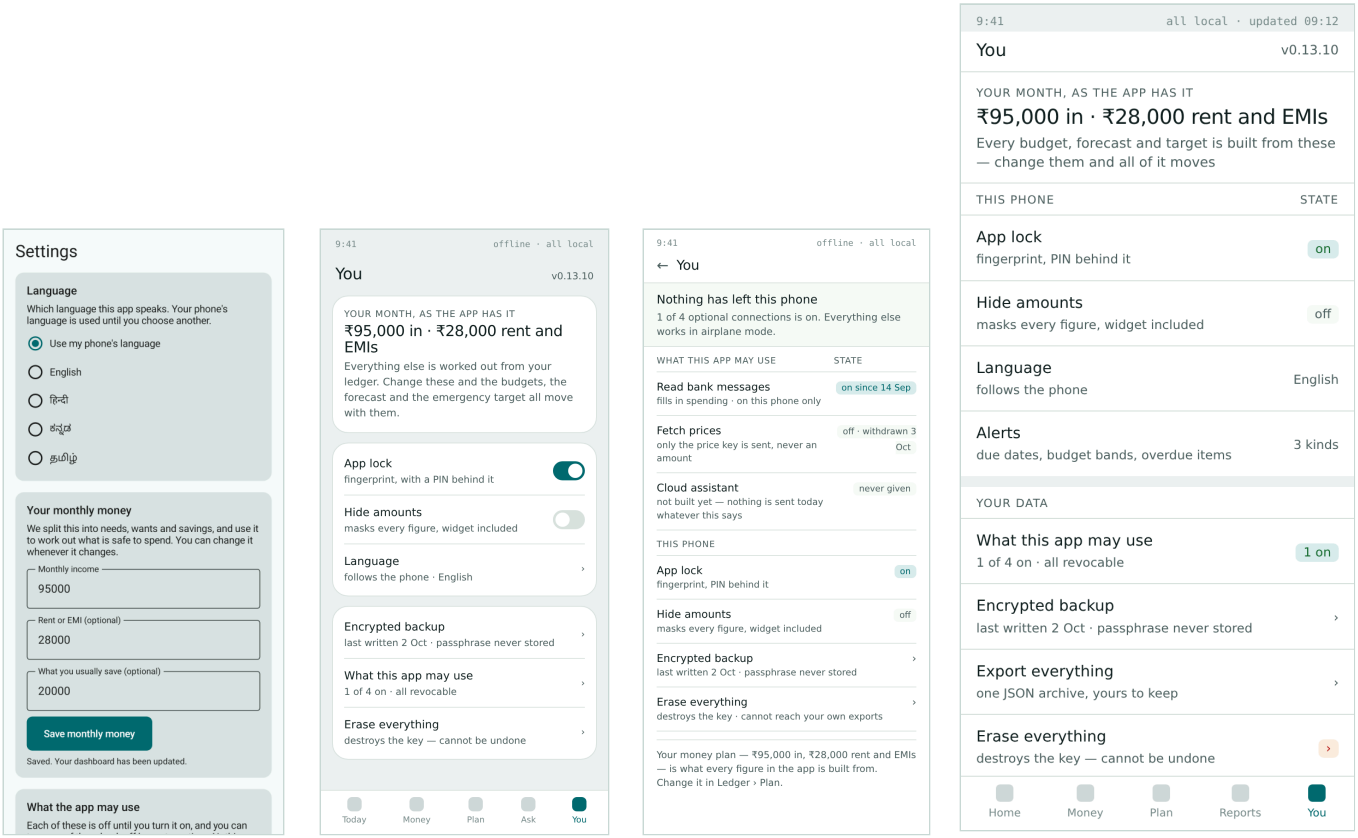
- A conversation with the composer pinned; the trace under each answer in monospace; actions that turn an answer into a reminder or a buy-list entry.
- A refusal offers the adjacent capability.

## Why, and what it borrows

- The trace is in monospace because it is machine provenance, and typography is the cheapest way to say “this is not prose”.
- AI-ARC-004 is the safety case, so what the answer was built from travels with it.

# 29 · Settings

state visible without opening a row




**v1** **v2** **v3** **v4** — the instrument panel

> **v4 vs v1:** the money plan becomes the KPI and state sits in a right-hand column
> **v4 vs v2:** adds the state column and the alerts summary
> **v4 vs v3:** the same tables; v4 adds export and alerts rows

## What it does

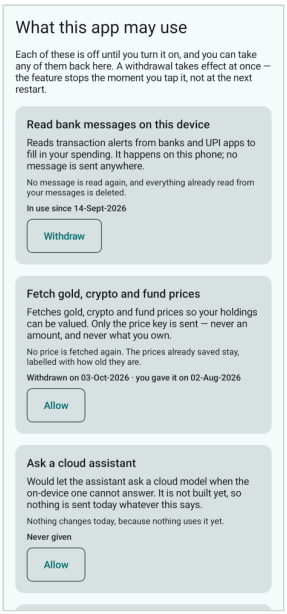
- The money plan as the KPI, because everything else is derived from it; two tables with state on the right.

## Why, and what it borrows

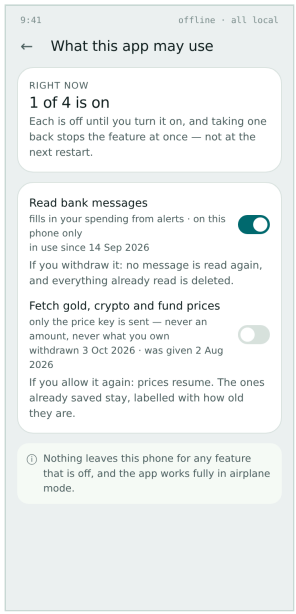
- Right-aligned state turns a settings list into something auditable at a glance, which is what the privacy posture deserves.
- Erase keeps the negative tone in the list, so the destructive row is identifiable before it is opened.

# 30 · Consents

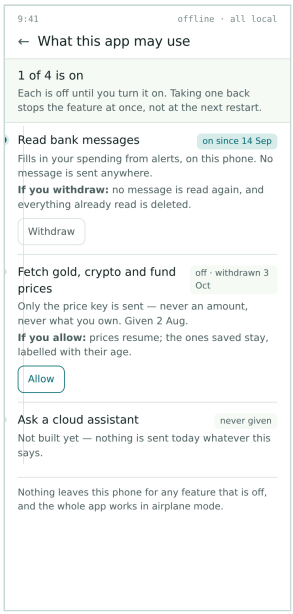
purpose, dates and consequence per row



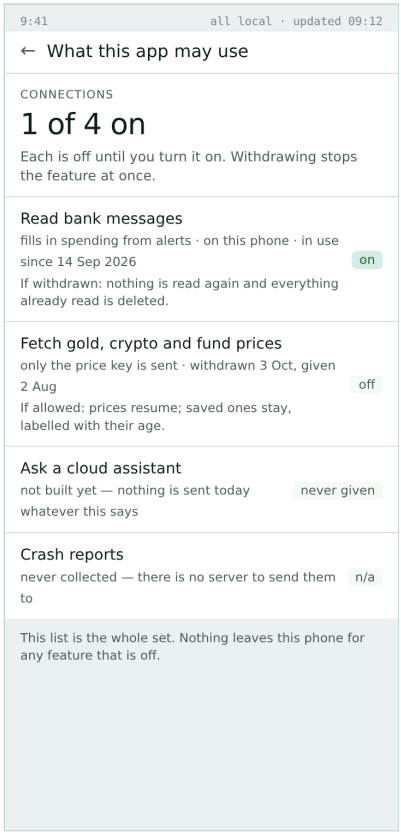

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** purpose, dates and consequence per row, plus a row for what is never collected
> **v4 vs v2:** adds the never-collected row — stating what the app does not do
> **v4 vs v3:** adds the never-collected row to v3's shape

## What it does

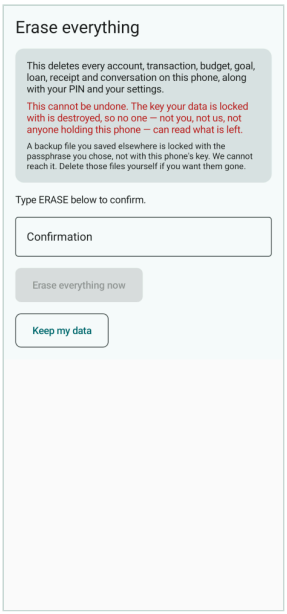
- Each consent with what it does, since when, and what stops if it is withdrawn.
- Crash reports are listed as never collected, because there is no server.

## Why, and what it borrows

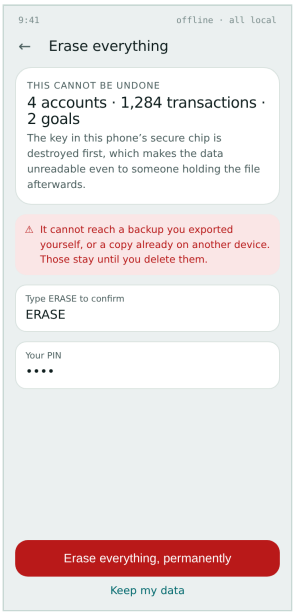
- Listing a thing the app does *not* do is the cheapest trust the design can buy, and it pre-empts the question.
- Both dates stay on a withdrawn consent — this is a ledger, not a switchboard (P-01, DPDP).

# 31 · Erase everything

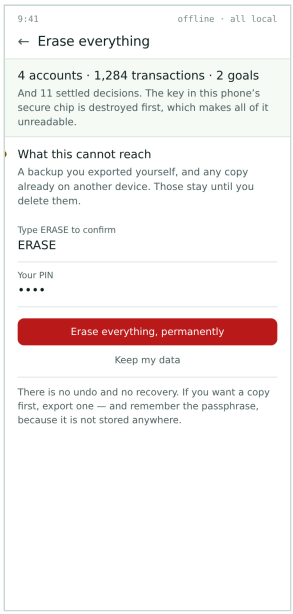
scope first, then what it cannot reach



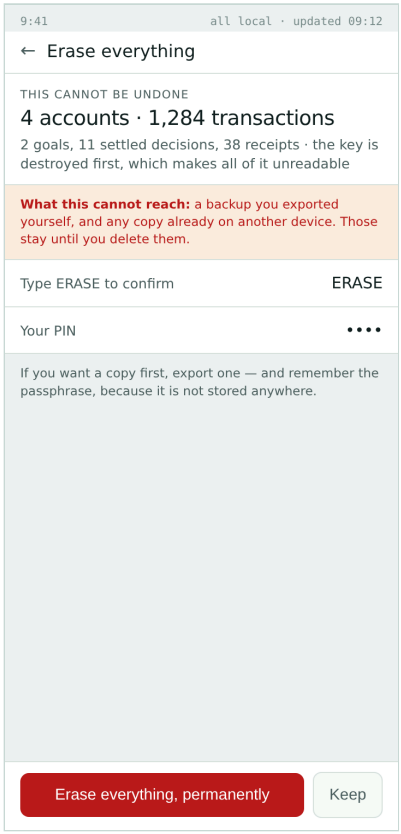

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** counts as the figure, the caveat in the warning tone, one destructive action
> **v4 vs v2:** same scope; v4 puts the counts in the KPI block
> **v4 vs v3:** same as v3, including the settled-decision count

## What it does

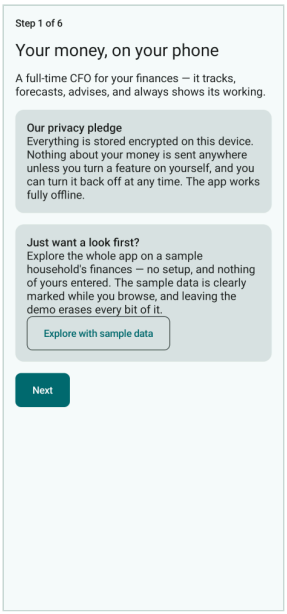
- The counts as the figure; the caveat in the warning tone; the confirmation and PIN as rows; one destructive button.

## Why, and what it borrows

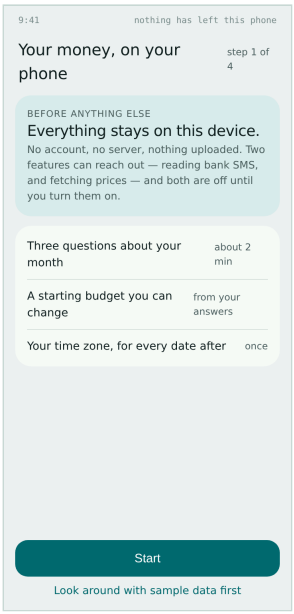
- “Everything” is not a scope. Counts are, and the settled-decision count is included because the user should know that record dies too.
- The negative role is used once in the whole app, which is what keeps it meaning *stop*.

# 32 · Onboarding

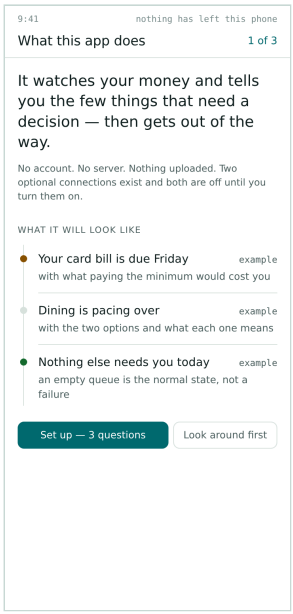
the promise, then what will be asked



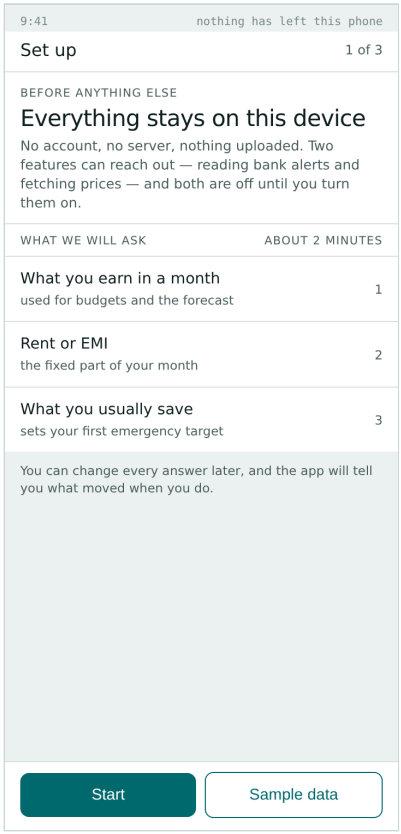

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** the promise is a KPI and the three questions are a table with what each is used for
> **v4 vs v2:** says what each answer is used for, and that changing one tells you what moved
> **v4 vs v3:** no queue to preview, so it shows the questions themselves

## What it does

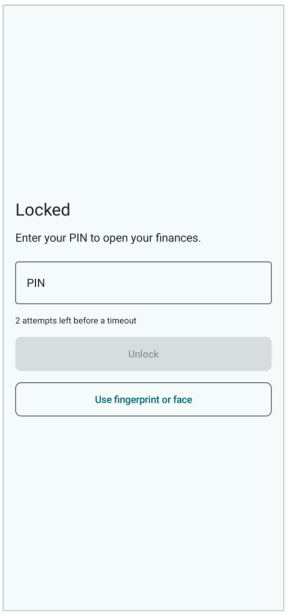
- The privacy position as the KPI, the three questions as a table with what each is used for, and two actions.

## Why, and what it borrows

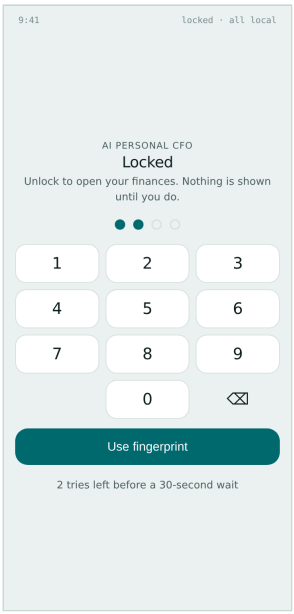
- Showing the questions before asking them is how a setup flow earns two minutes of attention.
- “You can change every answer later, and the app will tell you what moved” is the sentence that makes the first screen safe to say yes to.

# 33 · App lock

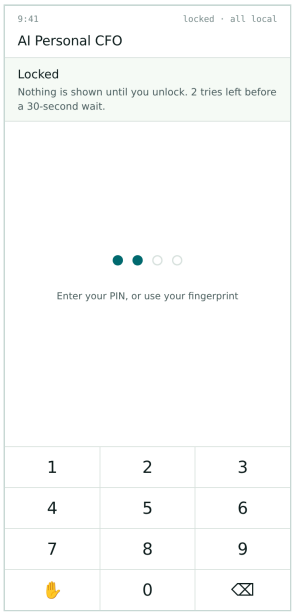
a keypad, nothing else



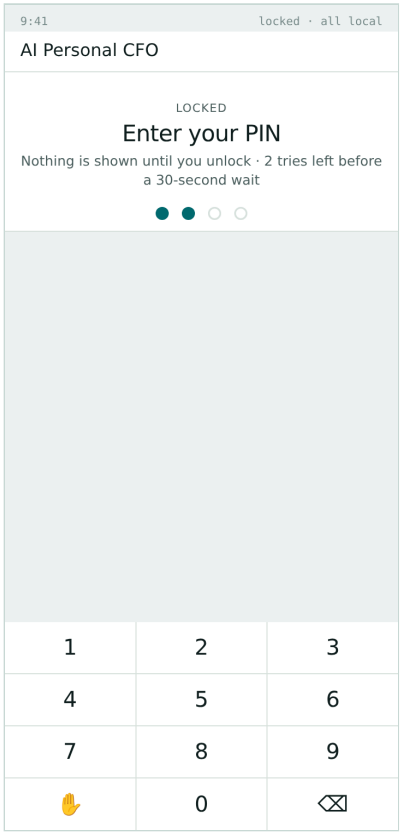

v1




v2




v3




**v4** — the instrument panel

> **v4 vs v1:** same keypad, with the lock state in the KPI block
> **v4 vs v2:** identical in substance — none of the three disagrees about this screen
> **v4 vs v3:** identical in substance; only the chrome differs

## What it does

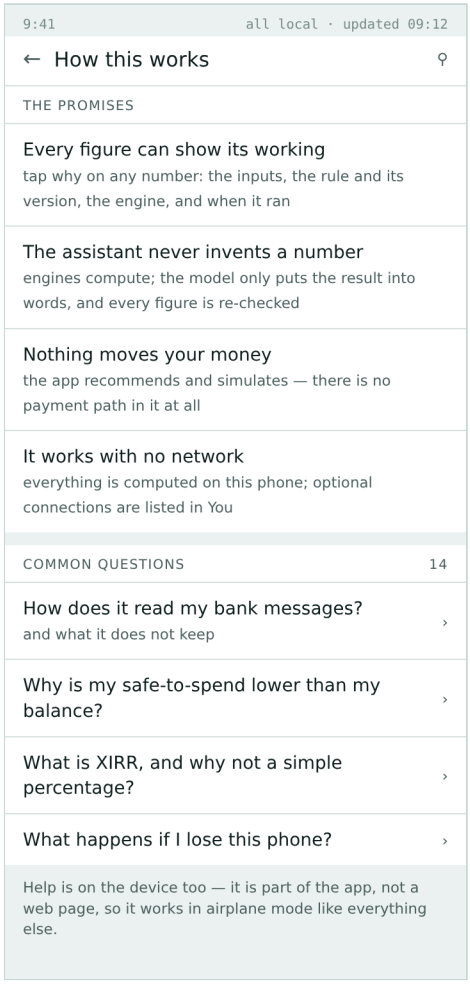
- Status, tries left, four dots and a full-bleed keypad; fingerprint is a key on the pad.

## Why, and what it borrows

- The same argument v2 and v3 make — a text field summons the keyboard that hides the action at 200% font — reached through v4's denser chrome.
- Nothing else is on the screen, because the threat §23.1 names is someone else holding the phone.

# 34 · Help

new · how this works




**v4** — new in this design

> **Against the lower versions:** No counterpart. The promises are argued in all three documents and stated to the user in none of the designs until here.

## What it does

- The four promises, each with how to check it, then the fourteen questions people actually ask.

## Why, and what it borrows

- An app that claims its figures are checkable should say where the check is. This screen is where P-02, P-03 and P-07 are stated in the user's own language.
- It ships on the device, so it works in airplane mode like everything else — a help centre that needs a network would contradict the product.

## What v4 risks

| Risk | Why it is real | What would have to be true |
|---|---|---|
| **Density versus 200% font** | This is the most text-per-screen of the three designs, and a two-column row at 200% becomes four lines. The accessibility bar is not negotiable here — it is screenshot-tested. | Every row reflows to stacked at large font, and the baselines prove it in four languages before anyone commits. |
| **Charts are not accessible on their own** | A line is invisible to a screen reader, meaningless in the privacy blur, and useless in greyscale. | Every chart's figures stated in text beside it — which v4 does, and which must stay true when a chart is added later. |
| **The trading idiom carries pressure** | Range controls and sparklines invite frequent checking; that is what they are for in a broker app. This app has no trades to earn from and a rule against pushing decisions. | No streaks, no nudges, no badge counts beyond real ones, and an alerts policy that stays at three kinds. |
| **It is the most to build** | Charts, a watchlist with persistence, search over several stores, a reports generator and an alerts inbox are all new surfaces, not re-layouts. | A staged plan: the KPI/table layer first (re-layout only), charts second, watchlist and search third. |
| **Complexity for a user who does not want it** | Someone who opens the app monthly does not need an instrument, and will find it busy. | The watchlist can hold one row, and the home screen degrades to v2's shape when it does. |

### Choosing between the three

- **v2** if the goal is comprehension for an occasional user, and the team wants the smallest change that fixes the shipped screens.
- **v3** if the goal is behaviour — fewer missed due dates, more decisions taken — and the team can afford the decision log that makes it honest.
- **v4** if the goal is engagement by competence: the user already reads Kite or Groww, and will read this. It is also the only one of the three that lets the user decide what the home screen holds.
- **The hybrid worth naming:** v2's hierarchy on each screen, v3's *Now* as the first tab, and v4's watchlist, charts and search. They are compatible: v2 is about one screen, v3 about what the app asks of you, v4 about how much it is willing to show. Only the home screen has to choose.

> Everything in all three documents is a proposal. The tokens are unchanged, the engines are unchanged, and nothing here has touched a line of application code. What any of them would need before being built is an ADR for the navigation change, and baselines re-recorded in light, dark and 200% font.