

PACEDALL ACADEMY

PRACTICAL FIELD GUIDE

AI Product Design Playbook

Design principles and working checklists for clear, trustworthy and accessible AI-powered experiences.

BUILT FOR

Product managers, designers, founders and frontend teams.

INSIDE

User journeys, AI interaction patterns, accessibility, component states and engineering handoff.

Version 1.0 | Pacedall Labs Ltd | 2026

How to use this guide

Use this as a design-review companion. Work through the chapters against a real user journey and capture unresolved decisions in the product backlog.

Contents

- 01 Design around the user journey
- 02 Make AI behaviour understandable
- 03 Design high-quality prompts in the UI
- 04 Build an accessible visual system
- 05 Design states, not static screens
- 06 Handoff with shared acceptance criteria

Pacedall principle

A useful AI platform is a complete service: product, workflow, data, safeguards, operations and commercial value. The model is only one component.

Design around the user journey

Structure screens around decisions and progress.

Give every screen a primary purpose. Make the next useful action obvious and remove competing calls to action that do not support the user's current job.

Use progressive disclosure: show essential information first, then reveal advanced controls when they become relevant.

Practical checklist

- One clear page purpose
- Visible progress and system status
- Plain-language labels
- Useful empty and error states
- Mobile-first priority order

Decision rule

If everything is prominent, nothing is prominent.

Make AI behaviour understandable

Users need a useful mental model, not technical theatre.

Explain what the feature can do, what information it uses and where its limits sit. Set expectations before the first interaction.

Show sources, assumptions and editable inputs when they materially affect trust. Give users a clear way to correct, retry or escalate.

Practical checklist

- Describe capability in outcome language
- Label generated content appropriately
- Expose relevant sources or inputs
- Support edit, retry and undo
- Avoid fake certainty

Decision rule

Trust grows when the interface makes uncertainty manageable rather than hiding it.

Design high-quality prompts in the UI

Do not make every user face a blank text box.

Use guided starters, examples, structured fields and sensible defaults. Preserve room for expert control without forcing every user to become a prompt engineer.

Collect context once and reuse it with permission. Explain what will be remembered and let users update or delete it.

Practical checklist

- Offer three to five meaningful starters
- Use examples based on real jobs
- Validate required context
- Show what will be sent
- Keep memory transparent and editable

Decision rule

The best prompt interface often looks like a well-designed workflow, not a chat box.

Build an accessible visual system

Accessibility improves clarity for everyone.

Target WCAG 2.2 AA contrast, support keyboard navigation and never rely on colour alone to communicate state. Use semantic HTML and visible focus indicators.

Define reusable tokens for colour, type, spacing, radius, elevation and motion. Test both light and dark themes at component level.

Practical checklist

- Text contrast at least 4.5:1
- Keyboard path works end to end
- Focus is always visible
- Touch targets are comfortably sized
- Motion respects reduced-motion settings

Decision rule

A theme is complete only when every component remains legible in every state.

Design states, not static screens

Products spend much of their life loading, failing or waiting.

Specify default, hover, focus, active, disabled, loading, empty, error, success and partial-result states. Include slow networks and interrupted sessions.

Use skeletons or progress indicators when waiting is unavoidable. Give errors a cause, consequence and next action.

Practical checklist

- Loading state communicates progress
- Empty state teaches the next step
- Errors explain recovery
- Destructive actions require care
- Success confirms what changed

Decision rule

A polished happy path cannot compensate for an unusable failure path.

Handoff with shared acceptance criteria

Design and engineering should share one definition of done.

Document responsive behaviour, content rules, interactions, accessibility requirements and analytics events alongside the visual design.

Review the built product in real browsers using representative content. Fix systemic token or component issues before one-off page overrides.

Practical checklist

- Tokens and components named consistently
- Breakpoints and reflow documented
- Real content tested
- Accessibility acceptance criteria included
- Design QA completed before release

Decision rule

Handoff is complete when the implementation intent is testable, not when a file link is sent.

Keep building

Use this guide as a working artefact. Record decisions, test assumptions with real users and update the document as evidence changes.

Learn more at academy.pacedall.com