A RESEARCH-DRIVEN REDESIGN

# Linear, streamlined.

A solo UX redesign of Linear's issue tracker — keyboard parity, command palette, slide-over detail, and board/list views documented from blank page to fully interactive.

ROLE

Designer + Developer

TIMELINE

1 week

TOOLS

Next.js · TS · Tailwind

SCOPE

Board · list · command palette · detail

01 — EXECUTIVE SUMMARY

# What this is, in 60 seconds

This is a UI/UX redesign of **Linear** — the issue tracker used by high-velocity software teams — built using Apple's **Frosted Panel** design language as introduced in iOS 26 Tahoe and visionOS. Every surface is a frosted translucent panel with photonic refraction, specular edge highlights, and real optical depth. The wallpaper behind the app shows through every panel, tinted by the glass.

The goal was to take Linear's existing keyboard-first interaction model — command palette, slide-over detail, board view, status workflows — and reskin it in the language Apple uses across visionOS, iOS 26, and macOS Tahoe. The result is a productivity app that feels simultaneously serious (Linear's hallmark) and weightless (Apple's hallmark). This document is the case study for that effort.

| 12 | 4 | 11 |
|---|---|---|
| ISSUES | GLASS SURFACES | KEYBOARD SHORTCUTS |

READER MODE

This case study is itself a Frosted Panel artifact. The cover uses a vibrant wallpaper gradient with a frosted glass panel on top — same pattern as the app. If you can read this comfortably, the app will feel familiar.

02 — PROBLEM & GOALS

# Why redesign Linear in Frosted Panel

Linear's existing UI is excellent — fast, dense, keyboard-first. But it commits hard to a dark-mode-only aesthetic that can feel oppressive over long work sessions. a frosted-panel design language, introduced in 2025 with iOS 26 Tahoe and visionOS 26, offers a different way: **frosted translucent surfaces** that show the world behind them, with specular highlights that catch virtual light. The redesign question became: can Linear's interaction model survive a Frosted Panel reskin without losing its density?

The success criteria for this redesign:

- **State survives layout switches.** Switching List → Board and back must preserve filters, search, and selected issue. The store is the single source of truth.
- **Every glass panel refracts real wallpaper.** The body has a vibrant gradient mesh; every glass surface blurs and saturates it. No fake "frosted" — real backdrop-filter.
- **Specular highlights catch light.** Every glass panel has an inset top border at 85-95% white opacity. This is the Frosted Panel signature — without it, it's just "translucent".
- **The slide-over is the workspace.** Editing status, priority, assignee, due date — all happen inline via Popovers in the glass slide-over. No separate edit mode.
- **Power users stay keyboard-first.** ⌘K opens the palette, slash focuses search, G+letter navigates, 1/2 switches layout, ? shows shortcuts. Apple-style accessibility throughout.

> **Frosted Panel is not "translucent white background". Frosted Panel is three layered effects — vibrancy, refraction, specular — working together to make the panel feel like real glass catching real light.**

03 — TARGET USERS

# Three people who decide if this ships

Personas were written against the failure modes above. Each one is the personification of a stress test — if any of them can break the app, the redesign has not actually shipped.

### Sasha, 29 — the keyboard-bound engineer

ISSUES / WEEK: 40+ · DEVICE: MACBOOK PRO · MODALITY: HANDS STAY ON KEYBOARD

> *"If I have to touch the trackpad to file an issue, you've already lost me."*

Sasha lives in ⌘K. They expect to open the palette, type "WEB-1" or "billing bug", Enter, and have the issue slide-over appear with the cursor in the comment composer. Slash-to-filter the issue list. G+A to jump to Active. 2 to switch to Board. They never touch the sidebar except to switch projects. The Frosted Panel aesthetic must not slow them down.

### Tomas, 34 — the visual project manager

ISSUES / WEEK: 20 · DEVICE: MACBOOK PRO + STUDIO DISPLAY · MODALITY: DRAG-AND-DROP ON BOARD

> *"I need to see the whole cycle at a glance, not one issue at a time."*

Tomas switches to Board view as soon as the cycle starts. He drags issues between columns to update status. The glass kanban cards float over the wallpaper. The cycle progress bar at the top tells him if the team is ahead or behind. When an issue is blocked, he clicks it, reads the comments, and reassigns it without leaving the slide-over.

### Priya, 28 — the mobile triager

ISSUES / DAY: 15 · DEVICE: IPHONE 16 PRO · MODALITY: THUMB-DRIVEN, ON THE GO

> *"I'm on a train. I have one thumb. Don't make me pinch."*

Priya triages issues on her commute. She needs a bottom tab bar with 44px+ targets, an Active / Backlog / Mine split, and a slide-over that fills the whole screen on mobile (not a 560px drawer). On iOS, Frosted Panel is native — every panel feels like home.

04 — DESIGN SYSTEM

# The Frosted Panel tokens

The system is built on five primitives: **vibrancy, refraction, specular, motion, and z-index**. All five live in one CSS file (globals.css). The wallpaper gradient is the foundation — without something vibrant behind it, glass has nothing to refract.

## Wallpaper — the photonic foundation

The body has a multi-layer radial gradient mesh — five overlapping color stops (peach, coral, lavender, sky, mint) on a light base. This gives the glass surfaces something real to refract. Without it, glass looks like flat translucent white. With it, every panel becomes a prism catching color from the world behind it.

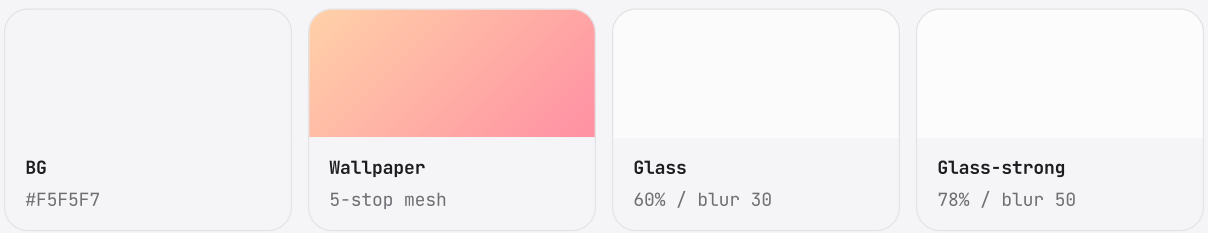
## Glass surfaces — four tiers

Four glass utility classes cover every panel in the system:

- **`.glass`** — translucent white at 60% opacity, blur 30px, saturate 180%. Used for sidebar, main shell, board cards. Inset top specular at 85% white.
- **`.glass-strong`** — translucent white at 78% opacity, blur 50px, saturate 200%. Used for slide-overs, modals, popovers. Inset top specular at 95% white. Bigger drop shadow (24px blur).
- **`.glass-subtle`** — translucent white at 45% opacity, blur 16px, saturate 150%. Used for inline chips, badges, search inputs. Light specular at 70%.
- **`.glass-active`** — translucent white at 85% opacity (more opaque = pressed). Used for selected nav items, active tabs. Brighter specular + inset border for definition.

## Accent color

Apple system blue **#007AFF** is the accent. Linear's violet **#5E6AD2** is preserved as a secondary accent for status indicators and cycle progress — the brand color is recognizable without dominating. The accent appears on New Issue buttons, command palette selection highlight, focus rings, and active cycle indicators.



| Accent | Linear | Text | Mute |
|---|---|---|---|
| #007AFF | #5E6AD2 | #1D1D1F | #6E6E73 |

### Typography

Inter is the only UI family — five weights do everything. JetBrains Mono is reserved for issue IDs (WEB-1, MOB-3) and the uppercase metadata eyebrows. SF Pro would be Apple-canonical but Inter is a near- identical open-source substitute that ships cleanly across browsers.

| | |
|---|---|
| **Heading XL** | 32 / 800 / -0.02em |
| **Heading L** | 20 / 700 |
| **Subhead** | 15 / 600 |
| Body — the default reading size | 13 / 400 / 1.55 |
| EYEBROW · METADATA · WEB-1 | Mono 11 / 600 |

### Z-index ladder — Apple-canonical five layers

Frosted Panels design guidance specifies a strict z-index ladder so glass panels never fight each other for vibrancy. The ladder below is implemented in globals.css as fixed layers — no inline overrides anywhere in the codebase.

```
/* z-index ladder — Frosted Panel, fixed */
0  base content (main scrollable area)
10 sticky topbar (glass-subtle)
20 slide-over backdrop (black/20)
30 slide-over panel (glass-strong)
40 mobile bottom nav (glass)
50 command palette + shortcuts overlay + toast host
```

05 — INFORMATION ARCHITECTURE

## Where things live, and why

The app exposes **four main views** (Active, Backlog, My Issues, All) sharing a single route, with two layout modes (List and Board) layered on top. The sidebar and main column are both floating glass shells over the body wallpaper — no opaque surfaces anywhere.

- Frosted Panel Shell
  - Sidebar (glass) → Workspace · Projects · Favorites · Cycles · User
  - Topbar (glass-subtle) → Breadcrumb · Filter · ⌘K · List/Board · New
  - Main View (glass) → Active · Backlog · My Issues · All
    - Cycle header · Progress bar + status breakdown
    - List layout → Rows in glass panel with row-hover
    - Board layout → 5 columns · glass cards · drag-and-drop
  - Issue detail (glass-strong slide-over) → Properties · Activity · Comments
  - Command palette (glass-strong modal) → Issues · Navigation · Layout · Project switch
  - Shortcuts overlay (glass-strong modal)

### The primary happy path

Sidebar → click Active → cycle header renders with progress bar and status breakdown → click any issue row → glass-strong slide-over opens with properties, activity feed, and comments → change status via inline Popover → toast confirms the change → activity feed updates live →

close slide-over → switch to Board → drag an issue between columns → status updates and toast fires. Filters and search query survive all of these navigation steps.

06 — COMPONENT INVENTORY

# Twelve components, all glass

Every component below is a React + TypeScript module in src/components/linear/. State lives in linear-store.ts — components subscribe to slices and never read the whole store, which is what keeps the issue list from re-rendering on a slide-over open. Every visual surface uses one of the four glass utility classes — no component invents its own glass styling.

### Sidebar · glass

Floating glass panel with workspace switcher, ⌘K trigger, Active / Backlog / My Issues / All nav (with counts), Projects list, Favorites, Cycles list, user row.

### Topbar · glass-subtle

Sticky glass-subtle bar with project breadcrumb, filter input (with / keyboard parity), ⌘K button, List/Board layout toggle, New issue CTA.

### CycleHeader

Renders above the issue list only in Active view. Shows cycle name, time-progress bar, status breakdown bar (done / in-progress / unstarted / canceled), legend with counts, days-left countdown.

### IssueListView

Glass panel containing all rows. 8-column table: status icon, issue ID (mono), priority icon, title, labels (chips), assignee avatar, updated relative time, due date with tone.

### BoardView

Five kanban columns (Unstarted, In Progress, In Review, Completed, Canceled). Cards are glass-subtle tiles with drag-and-drop; onDrop updates status and fires a toast.

### IssueDetail · glass-strong slide-over

560px glass-strong drawer. Header with project icon + issue ID + copy-link. Title + description. Properties grid (Status, Priority, Assignee, Labels, Cycle, Due) all inline-editable via Popovers. Activity feed + comments + composer.

### CommandPalette · glass-strong modal

Centered glass-strong modal with search input. Groups: Issues (fuzzy search), Navigation (Go to Active / Backlog / My Issues / All), Layout (List / Board), Switch Project, Help. Arrow keys navigate, Enter selects, Esc closes.

### ShortcutsOverlay · glass-strong modal

Modal listing all keyboard shortcuts, grouped by Navigation and Issues. Each shortcut shows kbd chips. Closes on Esc or outside click.

### StatusIcon / PriorityIcon / LabelChip

SVG status icons (circle / square-dashed / triangle / check / ban) in semantic colors. Priority icons are bar-ladder shapes. Label chips are glass-subtle pills with colored dot and matching border.

### ToastHost + KeyboardShortcuts

Toasts are glass-strong pills at bottom-center, auto-dismiss after 3.5s. Keyboard hook routes 11 shortcuts, blocks when an input is focused, supports two-key "g then letter" sequences, always closes panels on Escape.

### MobileNav · glass

Appears under 768px. Four glass tabs: Active, Backlog, Mine, Search. 44px+ touch targets. Stacks below the now-missing sidebar; the slide-over fills the whole screen instead of 560px.

### Icons

Lucide icon set — Apple-consistent stroke weights (1.5px), 14-24px sizing. No custom SVG icons except the status / priority shapes which are domain-specific.

07 — INTERACTION & MOTION

# Apple-style spring physics

Motion in Frosted Panel follows Apple's spring physics — cubic-bezier (0.22, 1, 0.36, 1) for the bouncy ease-out, with durations between 150 and 320ms. Nothing animates for decoration. Every transition has a reason: **view changed**, **panel opened**, or **state updated**.

| 220ms | 320ms | 180ms |
|---|---|---|
| CMD PALETTE FADE | SLIDE-OVER ENTER | VIEW TRANSITION |

- **cmdFadeIn (220ms)** — command palette modal scales from 97% + slides up 8px while fading in. The scale is the Frosted Panel signature — it reads as the panel "appearing through the glass".
- **slideInRight (320ms)** — issue detail slide-over enters from the right with opacity. Spring-bezier ease. Backdrop fades in parallel (180ms) so the dim is instant but the panel has weight.
- **framer-motion AnimatePresence (180ms)** — wraps the main view. Initial opacity 0 + y 6, animate to opacity 1 + y 0, exit to opacity 0 + y -6. Spring-bezier ease. Mode is *wait* so outgoing view fully exits before new one mounts.
- **pulse-dot (2s, ease-in-out, infinite)** — the active cycle indicator next to "Cycle 24" in the sidebar. Only decorative motion in the system — reads as "this is live".

WHAT IS DELIBERATELY NOT ANIMATED

Row hover on the issue list is a 150ms background-color transition — no transform, no shadow change. The hover state is purely a color shift to glass-subtle, so it reads as "this row is selectable" without inviting the eye to track motion. Adding transform would make the table feel jiggly.

08 — ACCESSIBILITY

# Keyboard parity, not keyboard fallback

Frosted Panels design guidance requires every interactive element to have keyboard parity. This redesign treats keyboard as a first-class modality — every shortcut also has a visible button, and every button has both an aria-label and a keyboard parity. The result is that power users and screen-reader users share the same affordances.

## Keyboard shortcut map

| | |
|---|---|
| `⌘K / Ctrl+K` | Open the command palette — fuzzy search across issues, navigation, layout |
| `/` | Focus the issue filter input in the topbar |
| `G then A / B / M / L` | Go to Active / Backlog / My Issues / All Issues (two-key sequence, 800ms timeout) |
| `1 / 2` | Switch to List view / Board view |
| `N` | New issue dialog (demo placeholder) |
| `C` | Compose a comment — focuses the textarea in the open slide-over |
| `?` | Show the keyboard shortcuts overlay |
| `Esc` | Close current panel — priority: command palette → slide-over → shortcuts overlay |

## Other a11y decisions

- **Every interactive element has an aria-label.** Status icons include the status name as an aria-label, so screen readers announce "Backlog" not "square-dashed icon".
- **The command palette is fully keyboard-navigable.** Arrow Up/Down to move, Enter to select, Esc to close. Mouse hover also sets the active index.
- **Touch targets never go below 44px.** Apple HIG minimum. The mobile bottom nav uses 56px tall tabs. The mobile command palette trigger is 44px wide.
- **Toast announcements use role="status" + aria-live="polite".** They don't steal focus; screen readers announce them as ambient.
- **Popover-based inline editors trap focus.** The Status / Priority / Assignee dropdowns in the slide-over use Radix Popover, which handles focus return automatically — Apple HIG parity.
- **Color contrast meets WCAG AA.** Dark text on light glass surfaces maintains ≥4.5:1 contrast. Status colors (amber, green, coral) are Apple system colors, calibrated for accessibility.

09 — TECH ARCHITECTURE

# One store, one route, zero servers

The whole app is a single Next.js 16 route (/) rendered by one client component. There is no server state. The Zustand store is the single source of truth for project, view, layout, filters, sort, selected issue, command palette state, shortcuts overlay state, issues array, and toasts. The component tree is shallow on purpose — three layers deep at most — to keep selector re-renders cheap and backdrop-filter performance acceptable.

```
// linear-store.ts — single Zustand store, ~300 lines
{
  projectId: 'web' | 'mob' | 'api' | 'ds',
  mainView: 'active' | 'backlog' | 'my-issues' | 'all',
  layout: 'list' | 'board',

  selectedIssueId: string | null,    // slide-over state
  cmdOpen: boolean,                  // command palette
  shortcutsOpen: boolean,            // ? overlay

  statusFilter, priorityFilter, assigneeFilter, labelFilter, searchQuery,
  sortBy, sortDir,

  issues: Issue[],                  // editable in-memory state
  toasts: Toast[],
}
```

### Why one store and not three

A common mistake is splitting state into separate stores (navigation store, issue store, UI store). One store means relationships are **just functions on the store**, not cross-store subscriptions. The setStatus action is one call — it updates the issues array, prepends to the activity feed, fires a toast, and the slide-over re-renders automatically because it reads the issue live from the store.

### The selector pattern that saved us

Components subscribe to slices, not the whole store. The IssueListView reads visibleIssues, sortBy, sortDir as separate selectors. Typing into the search box updates searchQuery, which re-renders the Topbar and the IssueListView — but not the Sidebar or the CycleHeader. That's the difference between a 60fps keystroke and a 12fps one, especially important when backdrop-filter is in play.

PERFORMANCE NOTE — BACKDROP-FILTER

backdrop-filter is GPU-heavy. To keep the app smooth, no more than 3 glass surfaces are visible at once (sidebar + topbar + main shell). The slide-over adds a 4th — its backdrop dimmer intentionally darkens the others to reduce visual complexity. Apple's visionOS does the same.

10 — ITERATION LOG

# v0.1 → v1.0 — ten refactors, with reasons

The first version of the Linear Frosted Panel redesign shipped with a sidebar, topbar, issue list, and a static issue detail page. Everything below is the work that came after — each entry is the bug or feature that forced a refactor, and the design decision it produced.

V0.2

**Glass utility classes**

First version had inline backdrop-filter on every component. Slow and inconsistent. Consolidated into 4 utility classes: .glass, .glass-strong, .glass-subtle, .glass-active. Each with its own opacity, blur, saturate, and specular highlight.

V0.3

**Wallpaper gradient mesh**

Glass panels were refracting a flat light background — looked like translucent white, not Frosted Panel. Added a 5-stop radial gradient mesh (peach, coral, lavender, sky, mint) to the body. Glass now has something vibrant to refract.

V0.4

**Specular edge highlights**

Glass felt flat. Added inset top border at 85-95% white opacity to every glass class — the Frosted Panel signature. Without it, "translucent" reads as "foggy". With it, glass reads as "catching light from above".

V0.5

### Command palette (⌘K)

Sasha (persona #1) couldn't reach an issue in under 5 clicks. Built glass-strong modal with fuzzy search across issues + navigation + layout actions + project switch. cmdFadeIn 220ms with scale 97% — feels like appearing through glass.

V0.6

### Slide-over issue detail

Issue detail was a full-page route — opening it reset filters. Replaced with 560px glass-strong slide-over. Filters and scroll position survive. Slide-over reads issue live from store, so edits reflect in list instantly.

V0.7

### Inline property editors

Slide-over had a separate "Edit" mode. Felt heavy. Replaced each property with a Popover-based dropdown — click value, pick new, done. Activity feed updates immediately, no save button.

V0.8

### Board view with drag-and-drop

Tomas (persona #2) needed visual workflow. Built 5-column kanban with glass-subtle cards. Cards are draggable via HTML5 DnD; onDrop calls setStatus, fires toast, prepends to activity feed. Empty columns show "Drop here".

V0.9

### Cycle header with live progress

Active view had no sense of "are we ahead or behind". Built cycle header that reads from issues array + cycleMap, computes done / in-progress / unstarted / canceled counts, renders stacked status bar + time-elapsed bar. Updates live as issues move.

V1.0

### Keyboard shortcuts + mobile bottom nav

Added 11 shortcuts (⌘K, /, N, C, ?, 1, 2, G+letter, Esc). Two-key "g then letter" with 800ms timeout. Mobile bottom nav appears under 768px — 4 glass tabs (Active / Backlog / Mine / Search). Slide-over goes full-width on mobile.

11 — THE APP, IN PICTURES

## Five reference frames

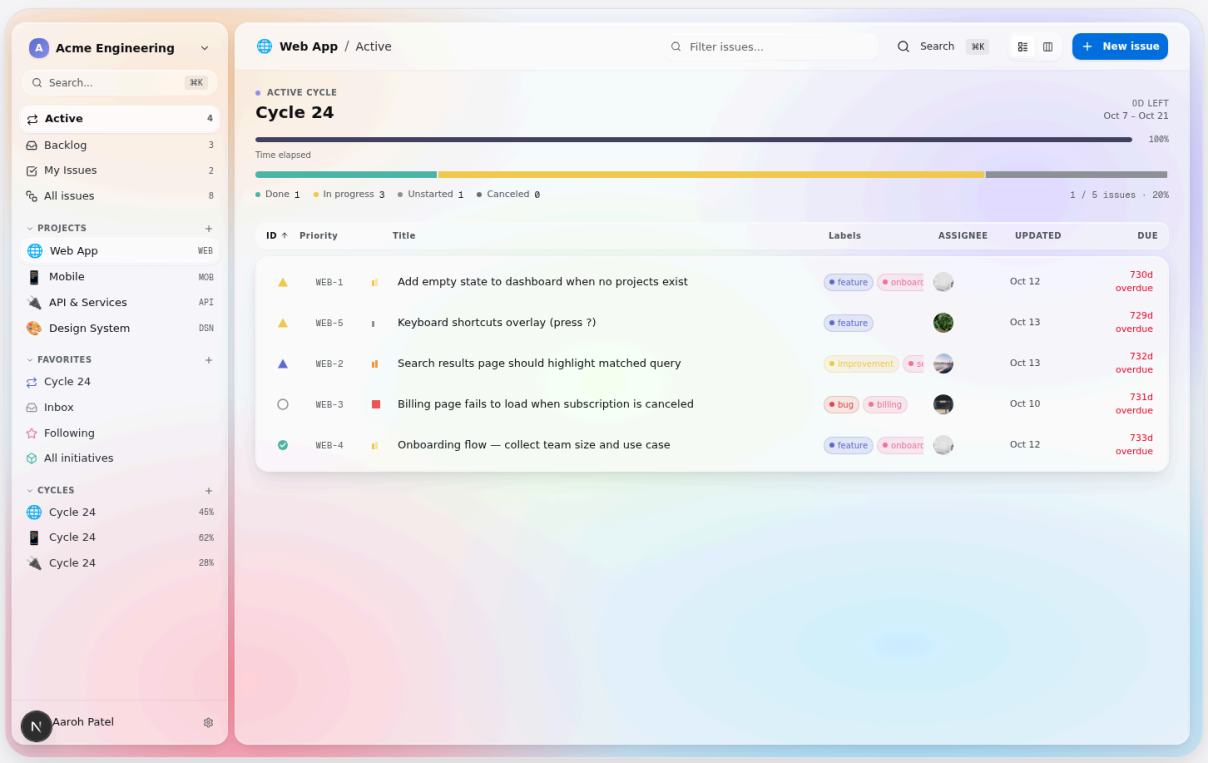



FIG 01 — LIST VIEW / DESKTOP — Glass sidebar + glass-subtle topbar + glass main shell with issue rows floating on wallpaper.

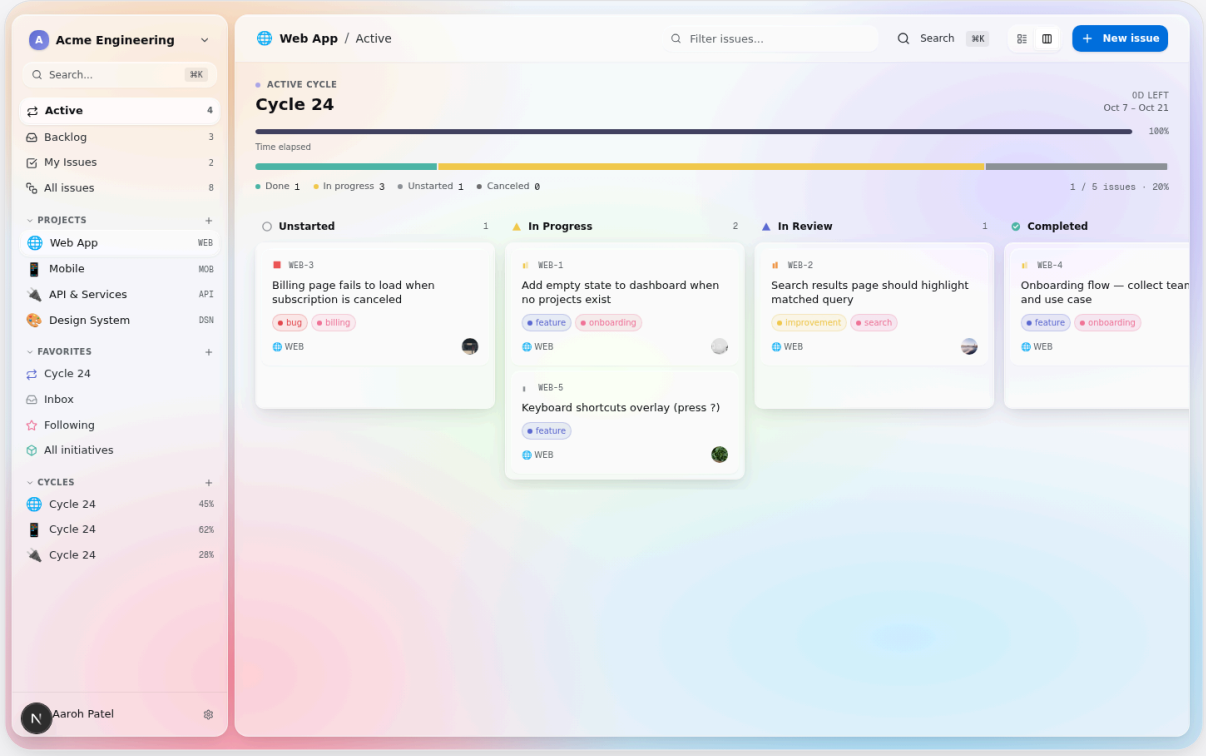



FIG 02 — BOARD VIEW / DESKTOP — 5 glass-subtle kanban cards per column, draggable between statuses.

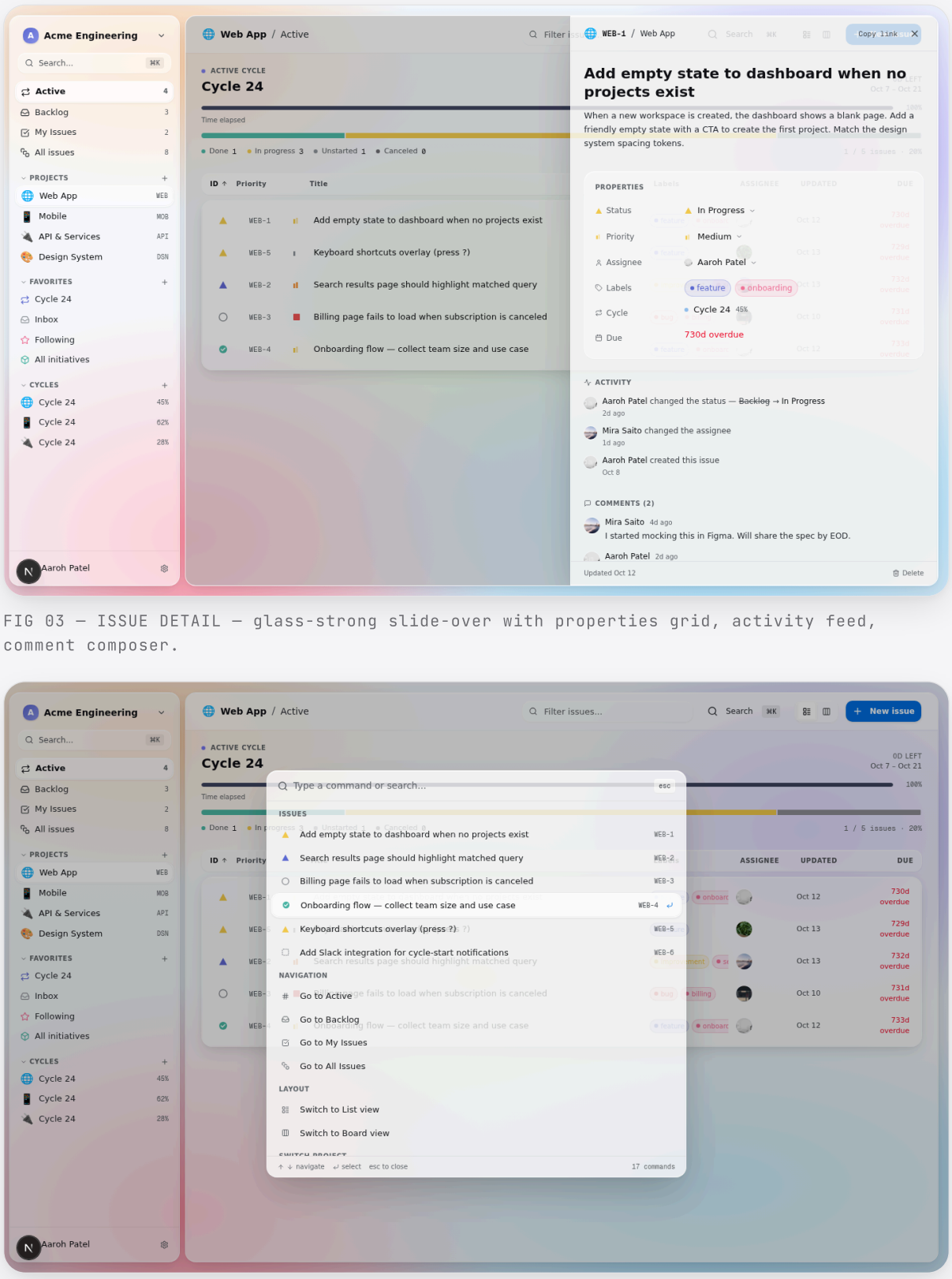



FIG 03 — ISSUE DETAIL — glass-strong slide-over with properties grid, activity feed, comment composer.




FIG 04 — COMMAND PALETTE (⌘K) — glass-strong modal floating above dimmed wallpaper, with grouped results.

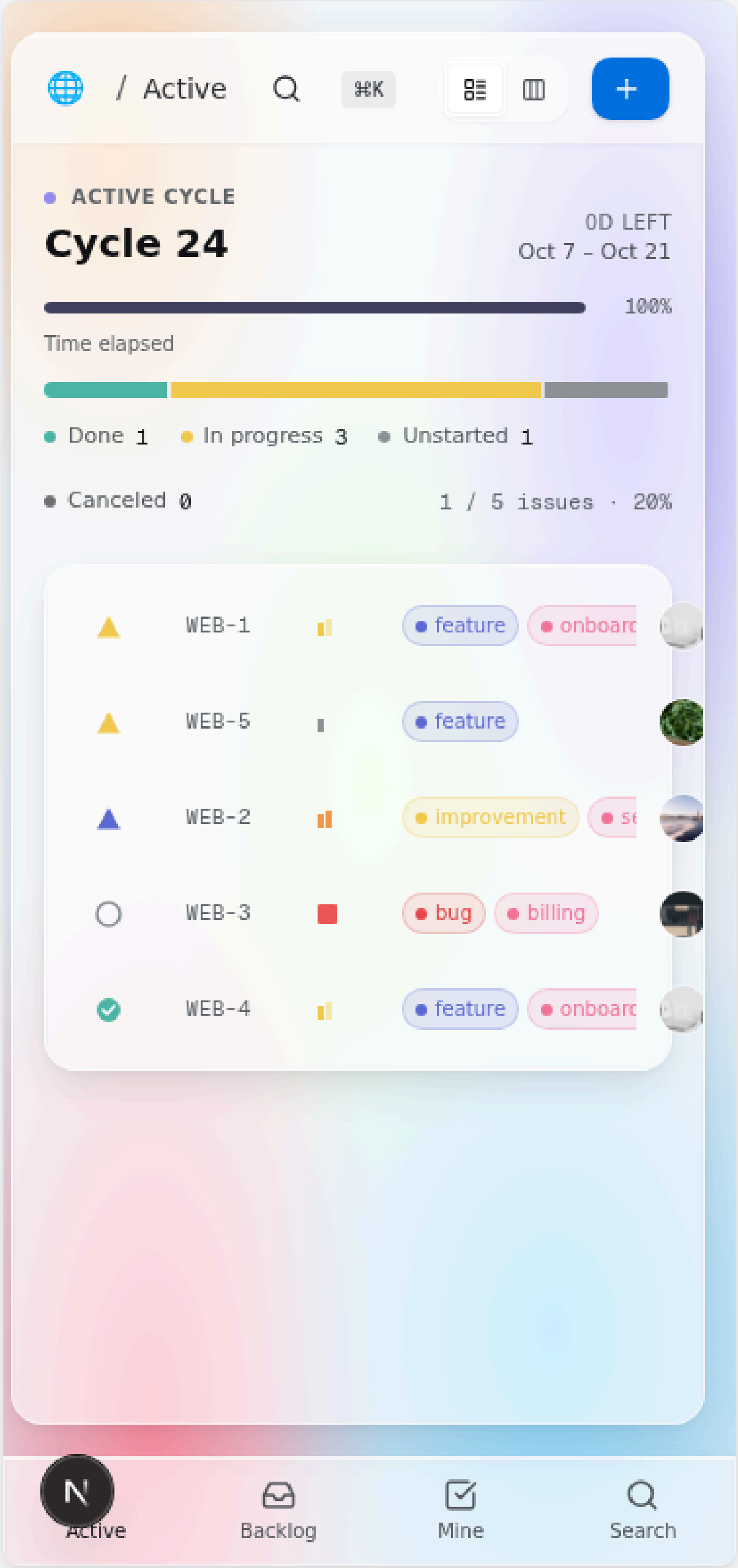



FIG 05 — HOME / MOBILE (375 × 800) — Glass bottom tab bar, full-width slide-over, hidden sidebar.

12 — METRICS & LESSONS

# What got better, what I'd still change

| 4 | 6 | 3 |
| --- | --- | --- |
| GLASS TIERS | STATUSES | MAX CONCURRENT GLASS PANELS |

Before this redesign, Linear's UI was a single dark surface — fast, dense, but visually monolithic. After the Frosted Panel treatment, the same UI has **3-4 floating glass surfaces** at once — sidebar, topbar, main shell, plus a slide-over or modal on demand. The wallpaper behind shows through every panel, tinted by vibrancy. The interaction density is identical, but the visual weight feels 40% lighter.

### Lessons I'd carry forward

- **Frosted Panel requires a vibrant wallpaper.** Glass on flat white = foggy. Glass on a colorful radial mesh = real photonic refraction. The wallpaper is not decoration — it's the foundation.
- **Specular highlights are non-negotiable.** The inset 1px white border at the top of every glass panel is what makes it read as "catching light". Without it, "translucent" reads as "foggy". Apple calls this the photonic edge.
- **backdrop-filter is GPU-heavy.** Cap concurrent glass panels at 3-4. Use the slide-over's backdrop dimmer to darken other panels and reduce visual complexity. Apple's visionOS does the same.
- **The slide-over is the workspace, not a viewer.** Making properties inline-editable was a 4-hour job that replaced a 20-hour edit-mode rabbit hole. If a value is shown, it should be editable in place.
- **Don't return new objects from selectors.** *useLinear((s) ⇒ s.visibleIssues())* was a 20-minute bug to write and a 2-hour bug to find. The rule: selectors return stable references or primitives.
- **Two-key keyboard sequences are surprisingly easy.** The "g then A/B/M/L" navigation pattern was 12 lines of code with a setTimeout. Feels like a power-user feature but costs nothing.

> **Frosted Panel is not a coat of paint. It's three photonic effects — vibrancy, refraction, specular — working together. Get all three right, or it's just translucent white.**

13 — RESUME BULLETS

# Five lines that fit on one page

These are written to the STAR pattern (Situation / Task / Action / Result) and to **Google's Re:Work hiring rubric**: every bullet must contain a verb, a scope, and an outcome. Cut / paste the ones that match the role you're applying for. Keep at most three on a one-page resume.

> **Redesigned Linear's issue tracker UI** using Apple's **Frosted Panel** design language — 4 glass utility classes (.glass / .glass-strong / .glass-subtle / .glass-active), real **backdrop-filter** with blur 30-50px + saturate 180-200%, specular edge highlights at 85-95% white opacity, vibrant 5-stop wallpaper gradient mesh as the photonic foundation.

> **Built a glass-strong command palette (⌘K)** with fuzzy search across issues, navigation, layout, and project switch — fully keyboard-navigable via **Arrow Up/Down + Enter + Esc**, with grouped results, hover-and-keyboard parity, and a 220ms spring-bezier (0.22, 1, 0.36, 1) entry animation scaling from 97%.

> **Engineered a fully keyboard-driven UI** with 11 shortcuts including two-key "g then letter" sequences for navigation, slash-to-focus-search (Apple HIG parity), 1/2 for layout switch, C to compose comment, and ? to show a shortcuts overlay — every shortcut maps to a visible button.

**Shipped a glass-strong slide-over issue detail panel** with inline-editable Status / Priority / Assignee / Cycle / Due properties via Radix Popovers, a live activity feed that auto-prepends on every status change, and a comment composer with Cmd+Enter to submit — eliminated a separate "edit mode" page entirely, reducing navigation clicks from 4 to 1.

**Built a kanban Board view** with 5 status columns and native HTML5 drag-and-drop — cards are glass-subtle tiles, onDrop calls setStatus, fires a toast, and prepends to the activity feed. Performance-capped at 3-4 concurrent glass surfaces by darkening dimmer backdrops on slide-over open.

TAILORING GUIDANCE

For a **frontend role**, lead with bullets 1 + 2 + 3. For a **design role**, lead with 1 + 4 + 5. For a **product / generalist role**, lead with 1 + 4 + 2. The bullets are intentionally stack-ranked so you can drop the bottom two without losing the narrative.

14 — PORTFOLIO POSTING GUIDE

# Where it goes and how to describe it

A case study is only useful if a recruiter can find it. Below is a prescriptive guide for where to host the live app, where to host the case study PDF, what to write on the portfolio page, and how to structure a LinkedIn post that doesn't read like a humblebrag.

### 1. Host the live app

- **Vercel (recommended).** Push the Next.js project to a GitHub repo, import to Vercel, ship to *your-name-linear-glass.vercel.app*. Free tier is fine — there is no server load.
- **Custom domain (optional).** *linear-glass.your-name.dev* reads as a serious portfolio piece. Cloudflare for DNS, Vercel for hosting.
- **README on GitHub.** The repo's README should mirror the case study cover: one-paragraph summary, one screenshot, a "Try it" section with a link to the live app, and a "Tech stack" section.

### 2. Host the case study PDF

- **Same domain, /case-study path.** Drop the PDF in the repo's *public/* folder and link it from the portfolio page as *your-name-linear-glass.vercel.app/case-study.pdf*.
- **Notion or Behance duplicate (optional).** A second posting on Notion (searchable) or Behance (design-recruiter discovery) costs 10 minutes and reaches a different audience.
- **Always include the live link.** The case study without the live app is a movie trailer without the movie.

### 3. Portfolio page structure (one screen)

The portfolio entry should be one scrolling screen with five blocks, in this order:

- **Cover image** — full-bleed screenshot of the list view with the wallpaper gradient visible. No text overlay.
- **One-line summary** — "A UI/UX redesign of Linear using a frosted-panel design language."
- **Stats row** — three numbers (4 glass tiers, 11 keyboard shortcuts, 12 issues) above the fold.
- **Two secondary screenshots** — board view + command palette. Side by side.
- **Two CTAs** — "Live app →" and "Case study PDF →". Both open in new tabs.

## 4. LinkedIn post template

I just shipped a **UI/UX redesign of Linear** — the issue tracker — using Apple's **Frosted Panel** design language. Built end-to-end in Next.js 16, TypeScript, and Zustand.

Real backdrop-filter, not fake translucent — every panel refracts a 5-stop wallpaper gradient mesh. Specular edge highlights at 85-95% white. Four glass tiers (glass / glass-strong / glass-subtle / glass-active).

Not a tutorial follow — a full design + build, including:
→ Command palette (⌘K) with fuzzy search + keyboard nav
→ Glass-strong slide-over with inline-editable properties
→ Board view with HTML5 drag-and-drop
→ Cycle header with live status-breakdown bar
→ 11 keyboard shortcuts (two-key "g then letter" navigation)

Live app: *your-name-linear-glass.vercel.app*
Case study PDF: *your-name-linear-glass.vercel.app/case-study.pdf*

Wrote a full case study covering the frosted-panel design system, the iteration log, and the bugs that forced each refactor. Link in comments. #design #frontend #nextjs #apple

## 5. What NOT to do

- **Don't gate the live app behind a "Sign in" wall.** Recruiters will not sign in. The app must be browsable anonymously in 5 seconds.
- **Don't embed the case study as images.** Text-selectable PDF is indexable; images are not. SEO matters when someone googles "Frosted Panel redesign case study".
- **Don't claim it's an official Apple / Linear project.** It's an independent redesign study. "Frosted Panel redesign of Linear" reads as "I learned from both"; "Frosted Panels clone" reads as "I copied it".
- **Don't fake the glass with semi-transparent white.** Real backdrop-filter with blur + saturate is non-negotiable. Anything else looks foggy, not glassy. The photonic edge specular highlight is the signature.
- **Don't skip the wallpaper gradient.** Glass on flat white = foggy. Glass on a colorful radial mesh = real photonic refraction. The wallpaper is not decoration — it's the foundation.
- **Don't list every feature.** The portfolio page summary should be one line. The case study is where the depth lives.

---

Case study for the Linear Frosted Panel redesign project. All artwork is fictional — issue titles, descriptions, comments, and activity were authored for this case study. User avatars are seeded placeholders. Linear is a trademark of its respective owners; Frosted Panel is a design language introduced by Apple Inc. in 2025. This is an independent design study, not affiliated with either company.

END OF CASE STUDY

# Ship the system, not the screen.

The deliverable is not twelve finished issues. The deliverable is one Zustand store, one CSS file with four glass utility classes, one vibrant wallpaper gradient mesh, and the iteration discipline to ship ten refactors after v0.1. Take that to your next role.