

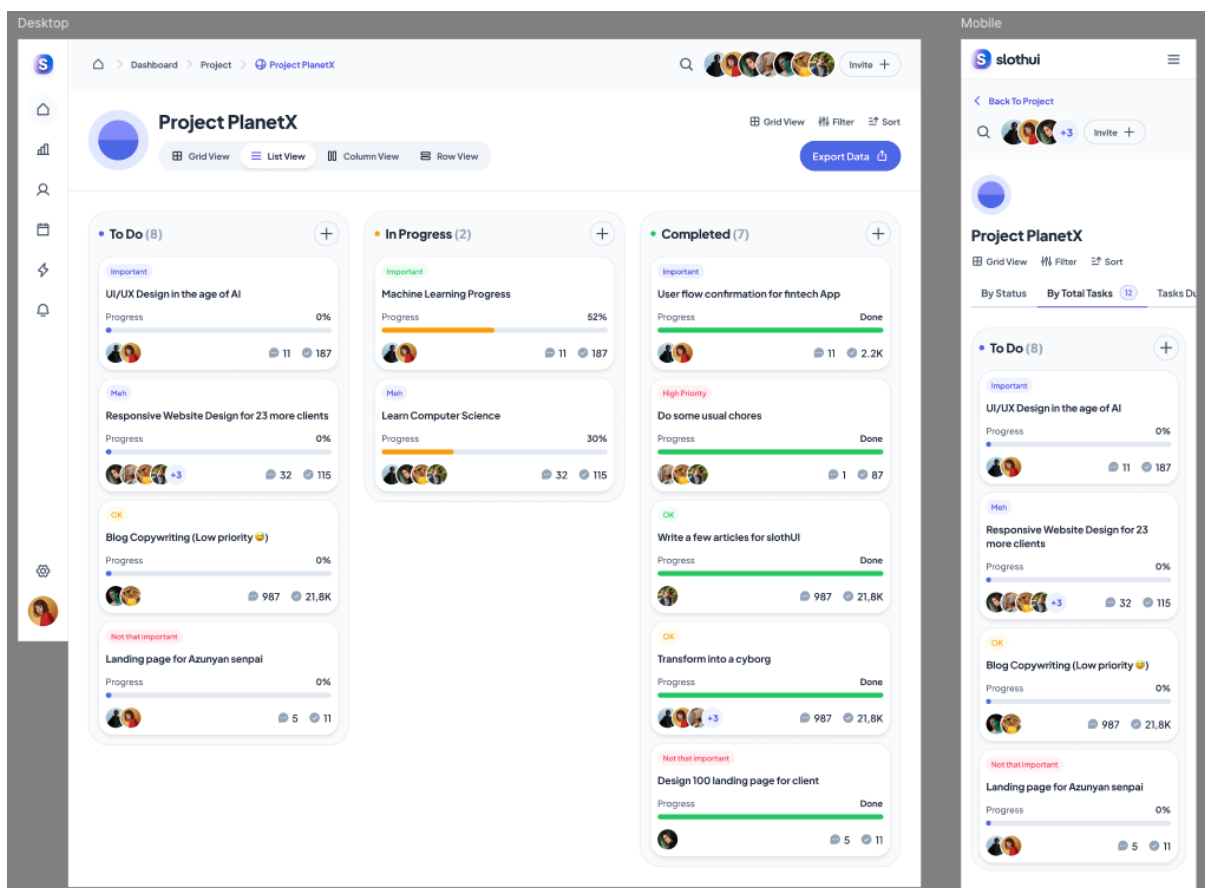
Zadatak: “Mini Kanban Board”

Trebam malu web-aplikaciju za praćenje zadataka – nešto poput Trella, ali light-weight. Dizajn je u Figma, aplikacija mora biti pixel-perfect, raditi offline i lako se skalirati na timsku kolaboraciju.

Figma dizajn

<https://www.figma.com/design/HAtUF2ipmWKm9TpuHYevqK/3-Free-Kanban-Board-App-UI-Kit--Community-?node-id=5773-39069>

(u Figma-i klikni **Duplicate**, koristi **Dev Mode** za gap-ove i mjere)



1. MVP razina

Funkcija	Implementacijski detalji
Tri kolone	<i>To Do, In Progress, Done</i> – layout, boje i tipografija identični Figmi.
Inicijalni podaci	GET https://jsonplaceholder.typicode.com/todos?_limit=12 → mapiraj u Task {id, title, status}. Nasumično ih rasporedi (id % 3) u kolone.
Drag & drop	Pomicanje kartica između kolona (koristiti drag&drop biblioteku po izboru)
Klijentski state	useReducer. Cijeli niz zadataka se persistira u localStorage kroz useEffect.
Responsivnost	CSS Grid ili Flex: 1 kolona (< 600 px), 2 (600-960), 3 (≥ 960 px).

2. razina (advanced)

Funkcija	Implementacijski detalji
Modal „New Task”	Dizajn prema Figmi. React-Hook-Form + yup validacija. Optimistic POST /todos (JSONPlaceholder vraća id).
Offline-first	PWA sa Workbox ili vite-plugin-pwa (cache statičkih asseta + Network First za API). Prikaži banner “Offline – promjene će se sinkati”.
Guardirana ruta /settings	React Router v6. Dummy JWT (localStorage.token). RequireAuth wrapper + Axios interceptor koji dodaje Authorization.
Unit testovi	≥ 5 testova React Testing Library + Jest (reduktor, drag&drop, forma, guard ...).

3. Razina

Funkcija	Implementacijski detalji
Real-time kolaboracija	Yjs + y-websocket (public echo server ili vlastiti). Promjene kartica broadcast-aj i primaj bez refresh-a.
UI za konflikte	Kad kartica stigne “remote” update, doda se badge “Updated remotely” + kratka animacija (Framer Motion).
Monorepo	Nx 17 ili Turborepo : apps/kanban, packages/ui, packages/data-access.
CI pipeline	GitHub Actions: lint → jest (coverage ≥ 80 %) → build → Cypress e2e (provjera DnD) → Lighthouse-CI (perf ≥ 90, JS ≤ 150 KB).
Perf budžet	CLS < 0.1, LCP < 2 s (3G), bundle ≤ 150 KB (first load).

Korištenje JSONPlaceholder

Operacija	Endpoint	Napomena
Dohvat startnih zadataka	GET /todos?_limit=12	Vraća id, title, completed – mapiraj completed =true → <i>Done</i> .
Dodavanje zadatka	POST /todos	Fake-API vraća {id:<generated>} – UI ostaje “optimistic”.
Ažuriranje zadatka	PUT /todos/:id	Vraća payload koji si poslao.
Brisanje zadatka	DELETE /todos/:id	Vraća {}.

(Mutacije su lažne – na error fallback i toast.)

Tehnička očekivanja

- **React 18+, TypeScript**
- Funkcionalne komponente, Hooks. Redux samo gdje potrebno.
- ESLint & Prettier clean, semantic commit poruke.
- Slobodno (čak i poželjno) koristiti UI library po želji

Predaja & evaluacija

- **Repo** na GitHub/GitLab – javni ili privatni (share invite).
- README: setup i upute za pokretanje aplikacije