

Tidum App Summary (Repo-based)

What it is

Tidum is a web app for logging work time, managing workforce workflows, and handling case-report processes.

The codebase combines a React frontend with Express APIs and PostgreSQL/Drizzle storage.

Who it is for

Primary persona (inferred from role model and pages): Norwegian operations teams where staff log hours/case work and team leads/admins review operations.

Formal persona document: Not found in repo.

What it does

- Track time with live timer, manual entry, bulk entry, and CRUD endpoints.
- View dashboard stats, activity feed, and chart analytics.
- Handle leave, overtime, recurring entries, and invoices (including invoice PDF endpoint).
- Manage case reports through draft, submit, review, approve/reject, and comment states.
- Use role-based auth and admin controls, including vendor API key management and `/api/v1/vendor/*` access.
- Edit public content in CMS: builder pages, blog, forms, media, SEO, versions, and scheduling.

How it works (architecture)

- Client: React + Wouter (client/src/App.tsx) with TanStack Query (client/src/lib/queryClient.ts).
- Server: Express app in server/index.ts wires registerRoutes, registerSmartTimingRoutes, and vendorApi.
- Data: PostgreSQL (pg pool) + Drizzle in server/db.ts with shared tables in shared/schema.ts.
- Flow: Browser -> `/api/*` routes + middleware -> Drizzle queries -> JSON -> React Query cache -> UI.

How to run (minimal)

- Prereqs: Node.js 20+ and PostgreSQL 14+ (CONTRIBUTING.md).
- npm install
- cp .env.example .env, then set at least DATABASE_URL and SESSION_SECRET.
- npm run db:push
- npm run dev, then open `http://localhost:5000`