Tldraw Project Structure
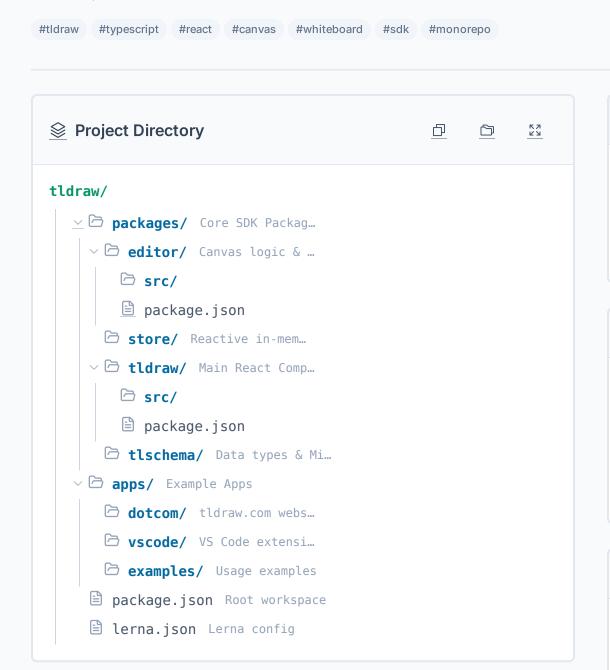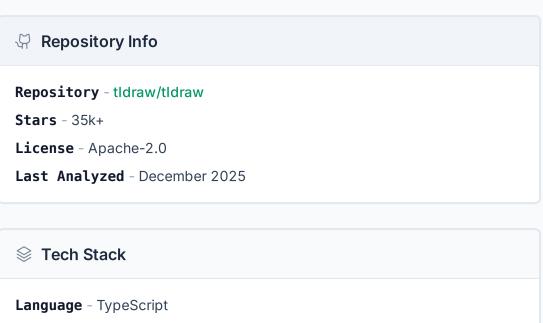
The tiny little drawing app and SDK. A sophisticated React/TypeScript monorepo designed to be embedded in other applications.

Updated 2025-12-30

#tldraw #typescript #react #canvas #whiteboard #sdk #monorepo

PNG PDF Copy </> Prompt

Project Directory

```
tldraw/
  packages/        Core SDK Packag…
    editor/        Canvas logic & …
      src/
      package.json
    store/         Reactive in-mem…
    tldraw/        Main React Comp…
      src/
      package.json
    tlschema/      Data types & Mi…
  apps/            Example Apps
    dotcom/        tldraw.com webs…
    vscode/        VS Code extensi…
    examples/      Usage examples
  package.json     Root workspace
  lerna.json       Lerna config
```

Repository Info

Repository - tldraw/tldraw

Stars - 35k+

License - Apache-2.0

Last Analyzed - December 2025

Tech Stack

Language - TypeScript

Framework - React

Build Tool - Vite / Yarn

State Mgmt - Custom (Signia/Store)

Styling - Tailwind / CSS Modules

Architecture Notes

Tldraw is architected as an SDK first. The `packages/editor` contains the framework-agnostic logic for the canvas (coordinate systems, tools, history). The `packages/tldraw` provides the React bindings and default UI. The state is managed by a custom reactive store (`packages/store`) optimized for high-frequency updates (60fps interactions).

Key Directories

packages/editor/ - The brain of the whiteboard. Handles pointer events, camera movements, and the state machine for tools (e.g., 'draw', 'select').

packages/store/ - A custom in-memory database that powers the undo/redo stack and collaborative features. It uses a signal-based reactivity system.

apps/dotcom/ - The code for the official `tldraw.com` website. It consumes the packages from the monorepo.

Why This Structure?

Tldraw is a masterclass in building complex, interactive UIs on the web. It shows how to separate the 'Editor' logic from the 'UI' components, making the canvas embeddable anywhere.