



Real-Time Activity Tracker Frontend Development Plan (Inspired by ActivityWatch)

Foundation Completed (Phase 0)

- React + Vite + TypeScript scaffolded
 - TailwindCSS + ShadCN installed and working
 - Theme system with CSS variables
 - API integration for /events endpoint
 - Category Chart (Bar Graph) working
 - Recent Sessions Table built
 - Data ingestion pipeline fully integrated with backend
-

Phase 1 - Daily Summary Page

- Call /summaries/daily endpoint
 - Components:
 - SummaryHeader.tsx: Total active time, idle %, longest streak
 - CategoryChart.tsx (already partly built)
 - SessionTable.tsx (already built)
 - DatePicker.tsx (ShadCN component)
 - Implement simple user_id selector
 - Hook: useDailySummary(userId, date)
-

Phase 2 - Timeline Page (Hourly Usage)

- Backend: implement /summaries/timeline
 - Components:
 - TimelineChart.tsx (stacked bar chart with active/idle blocks)
 - Legend.tsx (active vs idle colors)
 - Hook: useTimelineSummary(userId, date)
-

Phase 3 - Top Domains Page

- Backend: implement /summaries/domains
- Components:
 - DomainsTable.tsx: list of top domains by time spent
 - DomainsPage.tsx: page wrapper
- Hook: useTopDomains(userId, date)

Phase 4 - Category Tree View

- Use nested category breakdown from summary response
- Components:
 - CategoryTree.tsx (ShadCN accordion or tree)
 - CategoryTreePage.tsx
- Hook: useCategoryTree(userId, date)

Phase 5 - Full Dashboard Layout

- Build layout similar to ActivityWatch:
- Summary header
- Timeline chart
- Top categories
- Top domains
- Category tree
- Recent sessions table
- Use responsive grid layout (TailwindCSS)

Phase 6 - Global Filtering System

- Global state management (React Context)
- Global FiltersBar.tsx (date picker, user selector, category filter)
- Pass filters to all API hooks

Phase 7 - Navigation & App Shell

- Full AppLayout.tsx using ShadCN UI components
- Sidebar navigation:
 - Dashboard
 - Timeline
 - Domains
 - Categories
 - Settings (future)
- Topbar with theme switcher
- Routing (React Router)

Phase 8 - User Auth (Deferred)

- OAuth2 or JWT integration
- Token-based protected routes

- Login & logout functionality
 - Multi-user support activated
-

File Structure Preview

/src /api /components /hooks /pages /context /types /utils

Notes:

- Fully inspired by ActivityWatch's clean modular UI
- Backend is ~85% ready; minor API additions will unlock full functionality
- This plan is highly scalable and clean for production-level code