

FitTrack – Athlete Health & Step Logger

Build a complete **React.js app** called **FitTrack**, a **modern athlete-style health and fitness tracker**. The app should allow users to **manually log daily steps, calories, and water intake**, and visualize progress with charts.

Use **localStorage** for data persistence, and **recharts.js** for visualization.

The theme should feel **sporty, dynamic, and inspired by Adidas design** — clean, minimal, with bold typography and a professional athletic look.

Core Features

1. Manual Entry Form

- Users can input daily:
 - Steps taken
 - Calories consumed
 - Water intake (in liters)
- Each entry is stored with the current date.
- Show a “Save Entry” button that stores data in localStorage.

2. Progress Dashboard

- Display daily progress in **charts** using recharts.js.
- Show 3 separate line or bar charts for:
 - Steps over the last 7 days
 - Calories over the last 7 days
 - Water intake over the last 7 days
- Each chart should include:
 - X-axis: Day
 - Y-axis: Value
 - Gradient line color that matches the sporty theme.

3. Weekly Reset

- Include a “Reset Week” button to clear all data from localStorage.
- Confirmation popup before deletion.

4. Summary Stats

- Show total steps, average calories, and total water for the current week.
- Display these in **metric cards** with bold typography and subtle shadow.

UI / Styling Details

- Use **Tailwind CSS** for styling.
- Use **Framer Motion** for simple fade/slide animations.
- Overall design theme: **Sporty + Adidas-inspired**
 - Black, white, and neon blue as the main color palette.
 - Clean geometric layout with grid-based design.
 - Use sharp button edges and hover effects (no rounded corners).
 - Use background gradients like from-black to-gray-900.
 - Add subtle motion effects when charts or cards load.
- Use **Adidas-style font**:

- Font: "Adineue PRO", "Helvetica Neue", "Montserrat", sans-serif (or use **Montserrat** from Google Fonts as close substitute).
- Include Adidas-like bold uppercase text for titles and metrics.

Components to Create

1. Header.jsx

- Title: FITTRACK in bold uppercase.
- Small tagline below: "Track. Train. Triumph."
- Include a logo icon (e.g., dumbbell or shoe) using react-icons.

2. InputForm.jsx

- 3 input fields: Steps, Calories, Water Intake.
- "Save Entry" button with animation on click.
- Validate inputs (no negatives, must be numbers).

3. Dashboard.jsx

- Display progress charts using recharts.
- Include 3 chart components inside a responsive grid.

4. StatsCards.jsx

- Cards showing:
 - Weekly Total Steps
 - Avg Calories
 - Total Water Intake
- Animate cards when values update.

5. ResetButton.jsx

- Button labeled "RESET WEEK".
- When clicked, show confirmation modal using Framer Motion for fade-in.

6. App.jsx

- Combines all components.
- Use a responsive layout (mobile → stacked, desktop → grid).
- Keep consistent padding/margins for a premium athletic feel.

Libraries to Install

npm install recharts framer-motion react-icons tailwindcss

Data Structure Example

```
{  
  
  steps: [{ date: '2025-10-28', value: 7500 }],  
  
  calories: [{ date: '2025-10-28', value: 2200 }],  
  
  water: [{ date: '2025-10-28', value: 2.5 }]  
  
}
```

Store this structure in localStorage as JSON and update daily.

Animation & Interaction Notes

- Animate cards and charts when they load or update.
- Fade in header when app starts.
- Add hover scale effect to buttons.
- On form submit, show a brief success check animation.

File Structure Suggestion

src/

├── components/

| ├── Header.jsx

| ├── InputForm.jsx

| ├── Dashboard.jsx

| ├── StatsCards.jsx

| └── ResetButton.jsx

├── App.jsx

├── index.js

├── styles.css

└── utils/

└── storage.js // helper functions for localStorage

Expected Output

- A fully functional, responsive React.js app.
- Users can log, view, and reset weekly health metrics.
- Data persists between reloads via localStorage.
- Charts update dynamically with animations.
- Theme feels bold, athletic, and clean — similar to Adidas UI.