

StreakBoard: Simple Habit Tracker

A one-page web app where you add daily habits, tick them off each day, and see your current streak. Data stays in the browser, so there are no accounts or servers.

How to use this kit: read the spec, set up the stack, then paste the prompts into your AI coding tool one at a time, in order. Test after every prompt before moving on.

1. One-page spec

Problem

Habit apps are bloated and push subscriptions. The user wants a fast, private checklist with streaks.

Target user

Someone who wants to track 3-10 daily habits on phone and laptop.

Features

- Add, rename, delete habits
- Tick today's habit done or undone
- Current and best streak per habit
- Last-7-days dot history
- Data saved in the browser (localStorage)
- Export and import JSON backup

Screens

Screen	Purpose
Today	List of habits with a check button and streak count.
Add/Edit habit	Small modal to name or delete a habit.
Settings	Export/import backup and reset data.

MVP: build this first

- ✓ Today screen with add/check/delete
- ✓ Streak calculation
- ✓ localStorage persistence
- ✓ Mobile-friendly layout

Later, not in v1

- Accounts and cloud sync
- Reminders and notifications
- Charts and stats
- Installable PWA

2. Beginner-friendly tech stack

Layer	Choice	Why
Front end	HTML + CSS + vanilla JavaScript	No build tools to break; your AI tool can see every line.
Storage	Browser localStorage	Free, instant, and private, with no database to set up.
Editor	Cursor	Agent mode can create and edit the files for you.
Hosting	Netlify or Cloudflare Pages (free)	Drag and drop a folder and it's live.

Setup before prompt 1

1. Install Cursor and sign in.
2. Create an empty folder called streakboard and open it in Cursor (File > Open Folder).
3. Install the 'Live Server' extension, or use any local preview.

3. Build prompts (12 steps, in order)

Copy each block into your AI tool's chat/agent panel. Accept the changes, run the test, then continue.

Prompt 1: Project skeleton

Create a plain HTML/CSS/JS project: index.html, styles.css, app.js. index.html links both files. Add a header 'StreakBoard' and an empty <main id="habits">. No frameworks, no build step.

Test it: Open index.html with Live Server and see the header.

Prompt 2: Mobile-first styling

In styles.css, make a clean mobile-first dark theme: max-width 480px centered column, large tap targets (min 44px), system font. Don't add any JavaScript yet.

Test it: Resize the browser narrow and wide; it should look good on both.

Prompt 3: Habit data model

In app.js create a habits array saved to localStorage under key 'streakboard.v1'. Each habit: { id, name, createdAt, doneDates: []} where dates are 'YYYY-MM-DD' in local time. Write load() and save() helpers. Don't render anything yet.

Test it: In the browser console, call save() then reload and check localStorage.

Prompt 4: Render the list

Write render() that shows each habit as a row: name, a big check button, and a streak placeholder. Show an empty state 'Add your first habit' when the list is empty.

Test it: Add a fake habit in console, call render().

Prompt 5: Add habit

Add a '+ Add habit' button that opens a small dialog with a text input (max 40 chars, trimmed, no empty names). Saving adds the habit, saves, and re-renders.

Test it: Add 3 habits, reload the page, they're still there.

Prompt 6: Toggle today

Clicking the check button toggles today's local date in doneDates. Done rows look visibly checked. Use local time, not UTC, for the date string.

Test it: Tick a habit, reload, still ticked.

Prompt 7: Streaks

Write currentStreak(habit) and bestStreak(habit) from doneDates. Current streak counts back from today, or from yesterday if today isn't done yet. Show 'Streak ■ N · Best N' on each row. Add a few unit-style console.assert checks.

Test it: Fake dates in console and confirm the numbers.

Prompt 8: 7-day history

Under each habit, show 7 small dots for the last 7 days (filled = done), oldest left. Tapping a past dot toggles that day.

Test it: Toggle yesterday and watch the streak change.

Prompt 9: Edit and delete

Long-press or click the habit name to open the dialog with Rename and Delete (with a confirm step).

Test it: Rename, then delete a habit; reload to confirm.

Prompt 10: Backup

Add a Settings section with Export (downloads streakboard-backup.json), Import (file input, validates shape before replacing data), and Reset all (confirm twice).

Test it: Export, reset, import, and all habits return.

Prompt 11: Polish and accessibility

Add aria-labels to icon buttons, visible focus styles, and a prefers-reduced-motion check. Make sure everything works with keyboard only.

Test it: Tab through the whole app without a mouse.

Prompt 12: Deploy prep

Add a <title>, meta description, favicon, and a short README.md. Check there are no console errors. List the files I need to upload to Netlify.

Test it: Drag the folder to app.netlify.com/drop and open the live URL on your phone.

4. Common pitfalls

UTC vs local dates. toISOString() uses UTC, so streaks break at night. Build date strings from getFullYear/getMonth/getDate.

Asking for everything in one prompt. Go one prompt at a time and test between each one.

Losing data. localStorage is per browser and per device. Tell users to export backups.

The AI adds React or a build step. Say 'no frameworks' again and reject changes that add package.json.

Silent JSON errors on import. Wrap JSON.parse in try/catch and validate before overwriting.

5. Launch checklist

- ☐ Works on phone and desktop
- ☐ No console errors
- ☐ Data survives reload
- ☐ Streak logic checked across midnight
- ☐ Export/import tested
- ☐ Title, description, favicon set
- ☐ Deployed to Netlify or Cloudflare Pages
- ☐ Shared the link with 3 people for feedback
- ☐ Wrote down the next feature from 'later'

Questions or problems with this kit? Email thetenderventure@gmail.com. Kit generated by AI agents from the Charging4ward template; double-check anything involving money, security, or personal data.