

3-Month Plan: AI + Frontend + Backend + Product Design

3-Month Plan: AI + Frontend + Backend + Product Design (Free, Beginner, 5-8 hrs/week)

Goal: Go from zero to shipping a small, real, AI-powered product. **Reality check:** At 5-8 hrs/week you have ~65-100 total hours. That's enough to ship something *small and real* — not enough to become an expert in four fields. This plan is deliberately narrow so you actually finish.

The scope we're aiming for: a simple web app with a form/input, a database, one AI-powered feature (via an API, not building your own model), and a basic UI — deployed live with a real URL you can share.

How the 3 months are split

Month	Focus	Outcome
1	Foundations + pick your idea	You can write basic HTML/CSS/JS, understand product design basics, and have validated a small idea
2	Build the core	Working frontend + backend + one AI feature, running locally
3	Polish, deploy, launch	Live product, tested with real users, publicly shareable

Work in 2-3 sessions/week of 2-3 hrs each. Don't binge one day and skip the rest — consistency beats intensity here.

MONTH 1 — Foundations & Idea Validation

Week 1: Product thinking + pick your idea

Don't start coding yet. Spend this week deciding *what* you're building. - Learn the basics of product discovery: talk to 3-5 people about a problem you want to solve, don't just assume it's needed. - Video: [Y Combinator's "How to Talk to Users"](#) - Video playlist: [Free UX Design Course \(full playlist\)](#) — watch the first few videos on the design process, skip the rest for now (you'll return to it in Week 10) - Deliverable: One paragraph describing your idea + who has this problem + why AI helps solve it.

Keep the idea tiny. Examples that fit this timeline: an AI-powered recipe generator from leftover ingredients, a tool that summarizes long articles/PDFs, an AI journaling app that gives

reflections, a simple AI-assisted resume reviewer.

Week 2: HTML & CSS basics

- Video playlist: [freeCodeCamp — Responsive Web Design \(full video course\)](#)
- Watch the HTML and CSS/Flexbox sections this week, ~4-5 hrs. Pause and type along — don't just watch.
- Deliverable: A static single-page site with your product's name, a headline, and a fake "sign up" form (no functionality yet).

Week 3: JavaScript basics

- Video: [freeCodeCamp — Full JavaScript Course for Beginners](#) (7-hour video, watch in chunks — variables, functions, arrays, objects, loops are the priority)
- Keep [JavaScript.info](#) open in a tab as a written reference when a video moves too fast on something.
- Deliverable: Make your static form actually respond — e.g., typing something and clicking a button shows a message on the page using JS.

Week 4: Git/GitHub + intro to APIs + wireframing

- Video: [Git & GitHub Crash Course for Beginners](#) (~80 min, covers everything you need)
- Tool: [Figma](#) (free tier) — sketch 3-4 screens of your product. Don't overdesign; boxes and labels are fine. Figma has its own YouTube channel with short getting-started videos if the UI feels unfamiliar.
- Deliverable: A GitHub account with your Week 2-3 code pushed, plus a rough wireframe of your 3-4 core screens.

MONTH 2 — Build the Core

Week 5-6: Frontend framework (React) — just enough

You don't need to master React, just enough to build your UI. - Video: [Learn React JS — Full Beginner's Tutorial](#) (freeCodeCamp) — watch components, props, and state; you can skip advanced hooks for now. - Keep [React official "Learn" docs](#) open as a written reference for anything the video glosses over. - Deliverable: Your wireframed screens rebuilt as React components — no backend yet, just UI with fake/hardcoded data.

Week 7: Backend basics (Node.js + Express)

- Video: [Node.js Crash Course](#) (Traversy Media, ~2 hrs)
- Video: [Express Crash Course](#) (Traversy Media, ~1.5 hrs)
- Deliverable: A simple backend server running locally with one working API route (e.g., `/api/generate` that returns dummy text).

Week 8: Database + connecting frontend to backend

- Video playlist: [Supabase Crash Course](#) — easiest free Postgres database with a generous free plan, great for beginners.
- Deliverable: Your app can save and retrieve a piece of user data (e.g., a saved entry) from a real database, and your React frontend talks to your Express backend successfully.

MONTH 3 — Add AI, Polish, Ship

Week 9: Add the AI feature

This is where “AI” enters — via an API call, not building a model from scratch. - Video: [Claude API Tutorial for Beginners \(Node.js\)](#) — walks through calling the Anthropic API from a Node backend, which fits directly onto your Week 7 Express server. - Written reference if you get stuck: [Anthropic API docs — Get Started](#) (Claude API has a free trial credit) - Alternative free option: [Google AI Studio / Gemini API](#) has a free tier. - Deliverable: Your backend route now calls an AI API and returns a real AI-generated response, shown in your frontend.

Week 10: Product design polish + basic UX

- Video playlist: go back to the [Free UX Design Course playlist](#) from Week 1 and watch the visual design / usability sections you skipped.
- Quick written reference: [Laws of UX](#) — short, skimmable, good to have open while you’re actually polishing.
- Deliverable: Clean up spacing, colors, loading states, and error states (“Generating...” spinner, friendly error messages).

Week 11: Deploy it live

- Video: [How to Deploy a React App to Vercel](#) (free tier, deploys React apps in minutes)
- Backend: [Render](#) free tier, or combine backend+frontend on Vercel with serverless functions
- Deliverable: A live URL you can send to anyone.

Week 12: Real user testing + launch

- Share with 5-10 real people (not just friends who’ll be nice). Watch them use it if possible, or ask for screen recordings.
- Fix the top 2-3 pain points they hit.
- Resource: [How to launch on Product Hunt](#) (free) or just share in relevant subreddits/communities for your niche.
- Deliverable: Launched, shared publicly, with a short list of what you’d build next.

Ongoing / parallel learning (light touch, not full courses)

Sprinkle these in when you have spare time — don’t treat them as mandatory weekly blocks: - **AI concepts (non-coding):** [Google’s AI Essentials](#) — free to audit, video lectures throughout, gives you vocabulary (prompting, tokens, context, hallucination) so you understand what you’re working with. - **General product sense:** search YouTube for “Lenny’s Podcast” clips — same author as the newsletter, interview format, easy to half-watch while doing chores.

Tools you’ll use (all free tiers)

- Code editor: [VS Code](#)
- Design: [Figma](#)
- Database: [Supabase](#)
- Hosting: [Vercel](#) / [Render](#)
- AI API: [Anthropic Claude API](#) or [Google Gemini API](#)
- Version control: [GitHub](#)

If you fall behind

Cut scope, not corners on understanding. It’s better to ship a plain HTML/JS app with one AI

feature than to abandon a half-built React app. The wireframe-to-launch pipeline matters more than which framework you used.