

# 5 Claude Code Skills — Real Project Setup Guide

The GitHub Skills That Ship Real Projects

01

## Skill 1: mcp-builder

[github.com/anthropics/skills/tree/main/skills/mcp-builder](https://github.com/anthropics/skills/tree/main/skills/mcp-builder) Scaffolds a working MCP (Model Context Protocol) server — the bridge that lets Claude connect to your own APIs, databases, or external tools. Install it by dropping the skill file into `.claude/skills/`. Then describe the tool or service you want Claude to access and it generates the full server code with protocol boilerplate included.

### QUICK TIPS

- Use this first — before building anything else, give Claude access to your data sources
- Works with Airtable, REST APIs, databases, and any custom tool you already use

02

## Skill 2: webapp-testing

[github.com/anthropics/skills/tree/main/skills/webapp-testing](https://github.com/anthropics/skills/tree/main/skills/webapp-testing) Writes and runs automated tests for your web application — unit, integration, and end-to-end — based on the features you describe. After building any new feature, describe what should work and Claude generates the test suite, runs it, and flags what is broken before you push.

### QUICK TIPS

- Run it after every new feature, not just at the end of the project
- Point it at a specific route or component for targeted coverage — not just the whole app

**03**

### Skill 3: frontend-design

[github.com/anthropics/skills/tree/main/skills/frontend-design](https://github.com/anthropics/skills/tree/main/skills/frontend-design) Designs and implements frontend UI components using your existing stack (Tailwind, React, plain HTML). Describe the screen or component you need — optionally share a reference screenshot — and it builds the working component to match. No Figma, no designer, no handoff.

#### QUICK TIPS

- Include a screenshot reference or written wireframe for best results
- Works with your existing design system if you have one — or makes sensible choices if you do not

**04**

### Skill 4: requesting-code-review

[github.com/obra/superpowers/tree/main/skills/requesting-code-review](https://github.com/obra/superpowers/tree/main/skills/requesting-code-review) Structures a proper code review request — packages your diff, adds context about intent, and asks Claude to review for correctness, security, and style. Run this before any significant merge to get specific, actionable feedback — not generic suggestions. Claude reviews intent alignment, not just syntax.

#### QUICK TIPS

- Run before every significant push, not just big features
- The skill formats the context automatically — Claude will flag security issues and logic bugs you missed

**05**

### Skill 5: skill-creator

[github.com/anthropics/skills/tree/main/skills/skill-creator](https://github.com/anthropics/skills/tree/main/skills/skill-creator) Builds new Claude Code skills from scratch — writes the SKILL.md file, defines trigger conditions, and structures the workflow steps. Once you have repeated any workflow three times in a project, use this to package it as a skill. Your skill library compounds with every project you finish.

#### QUICK TIPS

- Use it at the end of every project to capture what you built
- Tell it what you want automated and it writes the full skill file — you never describe those steps again

## Putting It All Together

Run these 5 skills in sequence on any real project: 1. mcp-builder — connect Claude to your data sources first 2. frontend-design — build UI components feature by feature 3. webapp-testing — test each feature before moving on 4. requesting-code-review — get a structured review before every push 5. skill-creator — at the end, package any repeated workflow into a new skill Each skill removes a manual step. Together they cover the full build loop: connect → build → test → review → systematize.

### QUICK TIPS

- All 5 are open source — install by dropping the .md file into .claude/skills/
- Start with mcp-builder if your project needs external data, skill-creator if you want to build your skill library first

## Ready to go deeper?

This guide is just the starting point. In Vibe Coding Mastery you get the full step-by-step system for building real agentic projects with Claude Code — from first prompt to deployed product.

[Join the Vibe Coding Build →](https://builds.digicuratoragency.com/)

<https://builds.digicuratoragency.com/>