

• PROJECT OVERVIEW

# SKILLSMITH

Build, test, and share custom Claude skills with an AI-assisted editor and split-pane testing playground.

---

PLATFORM	FRAMEWORK	AI ENGINE	DATABASE	DATE
Web Application	Next.js 14 + React 18	Claude (Anthropic SDK)	Supabase (PostgreSQL)	February 2026

# AI skills are **hard to build** and **harder to validate**

Prompt engineering is one of the most impactful skills of the AI era, yet the tooling around it remains primitive. SkillSMITH addresses five fundamental friction points.

01

## Creation Friction

Translating a vague idea into a structured, production-grade prompt requires expertise most users don't have. There's no guided workflow to go from concept to polished skill.

02

## Quality Blindness

There's no easy way to know if a prompt actually works well. Users test manually, one message at a time, with no systematic evaluation or scoring methodology.

03

## Discovery Gap

Proven skills live in scattered docs, tweets, and repos. There's no central hub to browse, evaluate, and fork high-quality skill templates.

04

## Refinement Tedium

Improving a skill requires manual iteration: try, fail, edit, repeat. No AI-powered suggestions, no targeted improvements for specific sections.

## The Core Insight

The SKILL.md format — a structured markdown file with instructions, edge cases, and examples — is the emerging standard for reusable AI behaviors. But writing great

SKILL.md files requires the same AI assistance they're designed to deliver. SkillSMITH closes this loop.

# An end-to-end skill forge

SkillSMITH is a full-stack platform for creating, testing, and sharing Claude skills. Every step — from blank idea to community template — is AI-assisted.



## 01 Multi-Path Skill Creation

Three ways to start: describe your idea and let AI generate a complete SKILL.md, fork an existing community template, or start from a blank canvas. An animated card-stack interface guides you through naming, categorization, and generation.

## 02 Dual-Mode Editor

**Guided Mode** breaks skill authoring into five structured steps: name & trigger, instructions, edge cases, examples, and review. **Markdown Mode** opens a full Monaco editor for raw SKILL.md control. Switch between modes seamlessly — changes sync automatically.

## 03 Split-Pane Testing Playground

Send the same prompt to Claude with and without your skill applied. See responses side-by-side to visually compare the impact. Instantly validate that your skill produces the behavior you want.

## 04 Automated Test Suites

Build a suite of test cases with prompts and expected behaviors. Run them individually or all at once. AI scores each response across five quality dimensions, giving you a quantitative view of skill performance.

Browse a curated library of proven skill templates. Filter by category (Writing, Code, Business, Education, Productivity), sort by popularity or recency. Fork any template to make it your own. A public explore page lets unauthenticated users discover skills before signing up.

---

# Five layers of Claude integration

SkillSMITH isn't a thin wrapper around an API. Every AI interaction is purpose-built with specialized system prompts, structured output parsing, and quality guardrails.

**GEN**

## Skill Generation

Takes a natural-language description and produces a complete SKILL.md with 5–10 behavioral rules, 5–8 edge case handlers, and 3 diverse examples. The system prompt enforces specificity over generality and demands behavioral boundaries.

**IMP**

## Section Improvement

Targeted AI suggestions for individual sections. Context-aware: considers recent test failures to recommend fixes. Returns a rationale explaining *why* each change matters, teaching users skill design principles.

**EVL**

## Response Evaluation

Scores Claude's output against expected behavior using a five-dimension rubric. Returns a 0–100 score, detailed reasoning, and a pass/fail determination. Powers the automated test suite.

**A/B**

## Side-by-Side Testing

Runs parallel requests — one with the skill injected as a system prompt, one without. Surfaces the behavioral delta so users can see the exact impact of their skill in real-time.

## Evaluation Rubric

Every test case response is scored across five weighted dimensions:

DIMENSION	WEIGHT	WHAT IT MEASURES
Instruction Adherence	<b>30%</b>	Does the response follow the skill's behavioral rules?

DIMENSION	WEIGHT	WHAT IT MEASURES
Format Compliance	<b>20%</b>	Does the output match the expected structure and formatting?
Tone Consistency	<b>15%</b>	Is the voice and tone aligned with the skill's persona?
Edge Case Handling	<b>20%</b>	Does it gracefully handle unusual inputs and boundaries?
Completeness	<b>15%</b>	Is the response thorough without being excessive?

### Rate Limiting & Safety

All AI endpoints are rate-limited per user: 50 requests/hour and 100,000 tokens/day. Usage is tracked in the database to prevent abuse while allowing generous experimentation. Every request validates authentication before processing.

# Built on a modern, production-grade foundation

Every layer is chosen for developer experience, performance, and scalability.

No unnecessary abstractions.

## FRAMEWORK & RUNTIME

Next.js 14

React 18

TypeScript 5

App Router

Server Components

## STYLING & UI

Tailwind CSS

Framer Motion

Monaco Editor

Tabler Icons

Custom Component Library

## AI & INTELLIGENCE

Anthropic SDK

Claude API

OpenAI SDK (fallback)

Custom System Prompts

Structured Output Parsing

## DATA & AUTH

Supabase

PostgreSQL

Row-Level Security

SSR Auth

Middleware Guards

## CONTENT PROCESSING

Gray Matter (YAML)

Remark GFM

React Markdown

Custom SKILL.md Parser

## The SKILL.md Format

Each skill follows a structured markdown standard with YAML frontmatter:

```
---  
name: Explain Like I'm 5  
description: Breaks down concepts to match a specified comprehension level  
---
```

```
# Explain Like I'm 5
```

## ## Instructions

- Default to age-5 level unless specified otherwise
- Use concrete, physical-world analogies
- Never use jargon without immediately explaining it

## ## Edge Cases

- If asked about inherently adult topics, be honest and age-appropriate
- If concept can't be simplified without inaccuracy, explain at lowest honest level

## ## Examples

```
### Example 1: Happy path
```

User: What is gravity?

Claude: Imagine you're holding a ball. When you let go, it falls down...

# Secure, scalable, real-time

A clean separation between public and authenticated routes, with row-level security ensuring users only access their own data.

<b>5</b> AI ENDPOINTS	<b>12</b> API ROUTES	<b>4</b> DB TABLES	<b>20+</b> COMPONENTS	<b>5</b> CATEGORIES
-----------------------------	-------------------------	-----------------------	--------------------------	------------------------

## Database Schema

### profiles

Extends Supabase auth.users with display name and avatar. Auto-created on signup via database trigger.

### skills

Core entity: title, description, markdown content, visibility, category, tags, fork lineage, template flags, and usage count.

### test\_cases

Per-skill test cases with prompt, expected behavior, last results (with/without skill), AI score, and reasoning.

### api\_usage

Rate-limiting tracker: input/output tokens, endpoint, request count per user. Enforces 50 req/hr and 100k tokens/day.

## Route Architecture

### Public Routes

- / Landing page
- /explore Browse templates
- /login Sign in
- /signup Register

### Authenticated Routes

- /dashboard Skill list
- /skills/new Create skill
- /skills/[id]/edit Edit skill
- /skills/[id]/test Test playground
- /templates Template library

## **Security Model**

- Supabase Row-Level Security (RLS) policies on all tables
- Users can only read their own skills + public skills
- Middleware intercepts unauthenticated requests to protected routes
- API routes validate sessions before processing
- Fork operations verify source skill is public or owned by user

# The complete platform for AI skill engineering

SkillSMITH democratizes prompt engineering by making skill creation, validation, and sharing as intuitive as writing a document. Every step is AI-assisted. Every skill is testable. Every improvement is measurable.

## Create

AI-generated skills from natural language descriptions, with guided and raw editing modes

## Test

Side-by-side comparison, automated test suites, and five-dimension AI scoring

## Share

Community template library with forking, discovery, and usage tracking

---

Built with Next.js 14, React 18, TypeScript, Tailwind CSS, Supabase, and the Anthropic Claude SDK.  
Designed and developed by **Karthik** — February 2026.