

AGENTIC WEB · SKILL AS A SERVICE

WebSKILL

— A SaaS Service for Personal AI on the Web

A front-end SaaS (Skill-as-a-Service) running entirely in the browser · A personalized agent ecosystem for everyone

Chunhui Mo

2026.9



WEBSKILL

Contents

1 What is WebSkill ?

CONCEPT & VALUE

2 What can WebSkill do ?

CORE CAPABILITIES

3 WebSkill feature summary

FEATURE OVERVIEW



Agent Skill is a modular, reusable, standardized capability unit that packages professional domain knowledge, instructions, metadata, and optional resources (such as scripts and templates) to extend the capabilities of AI agents.

Use Cases

Widely applied where specific professional standards, business norms, or deep domain knowledge are required, e.g.:



Development & Code: Systematic code reviews, test-driven development (TDD), or complex Git branch management.



Business & Docs: Write client proposals with enterprise templates, financial audit variance analysis, multi-format reports (Word, PDF, PPT), or brand-guideline layout design.



Security & Compliance: Automated compliance checks, penetration testing, and vulnerability scanning.



Problems Solved



Context Bloat: Traditional AI crams massive instructions into a single prompt for business context, overloading the LLM context, driving up costs, and making it prone to "forgetting" key info.



Lack of Domain Expertise: General LLMs lack company-specific business logic (e.g., pricing strategies, approval workflows). Skills codify human expertise, turning agents from "chat-savvy bots" into "professional specialists for the enterprise".



No Repetitive Prompting: Users no longer re-teach the Agent in every session. A Skill is created once and reused automatically, standardizing end-to-end workflows with consistency.

Agent Skill Ecosystem Is Growing Rapidly

85,000+

Publicly available Agent Skills

As of Feb 2026

200,000+

Skills milestone in the Claude Code ecosystem

As of May 2026

54%

Anthropic's share of the AI coding tools market

27

Major platforms supporting the Skill standard

Including Microsoft, OpenAI, GitHub



Standards Linux Foundation to adopt it as an AI & Data Foundation candidate standard



Skill Essence: Package domain expertise into reusable, standardized capability units via "methodology encapsulation + toolchain integration + execution isolation", giving Agents cross-domain professional capabilities



Web-Based Agent Skill (WebSkill) is also a modular, reusable, standardized capability unit — a declarative contract that runs entirely in a closed loop within the browser. It is essentially a front-end native skill running inside Agentic Web applications.

3 Key Features of WebSkill



Closed-Loop Execution in the Browser

Unlike backend Skills on Node.js or cloud servers, WebSkill runs in a closed loop entirely in the browser. A lightweight directory of instructions, scripts, templates, and resources gives agents domain expertise for complex form filling, multi-step workflows, or data analysis.



Privacy via Absolute Isolation

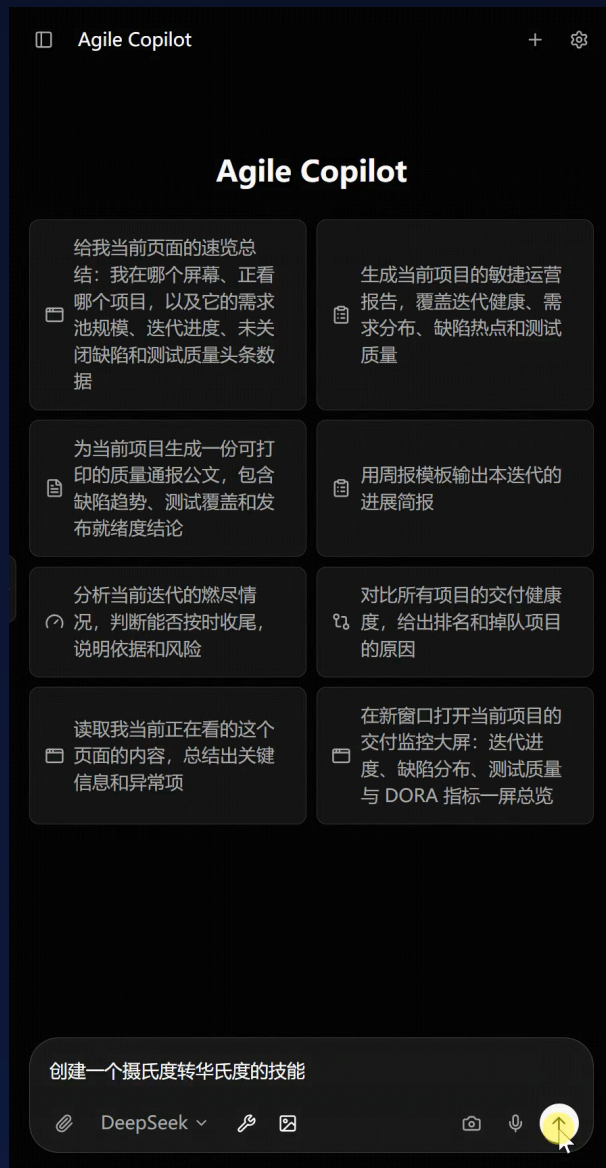
WebSkill is stored in the browser's OPFS (Origin Private File System) with strict same-origin isolation. Scripts run in a Web Worker sandbox, fully isolated from the main thread. The entire chain — intent parsing, routing, execution, rendering — completes locally, so sensitive data and credentials never leave the device.



Dynamic Evolution & Personal Assets

Each user's skill directory is stored locally and evolves with personal usage. When existing skills fall short, the on-device LLM can analyze needs and record operation traces, then — with user consent — generate new candidate SKILL files into OPFS, achieving true experience accumulation.

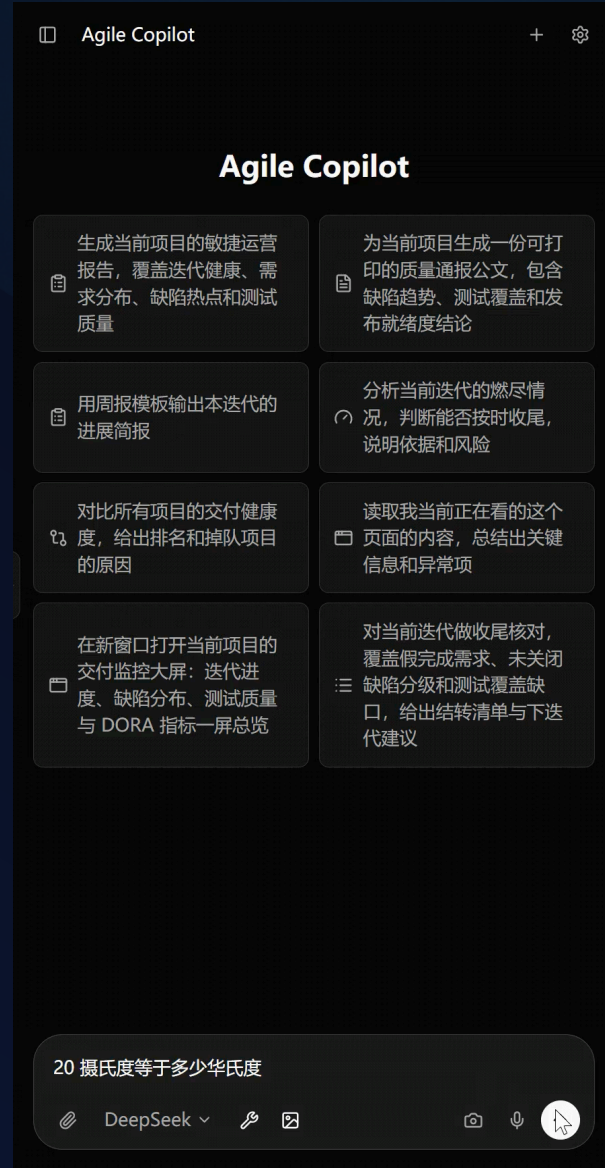
	Traditional Backend Skill	WebSkill
Runtime Environment	 Node.js / Cloud Server	 Browser Web Worker (OPFS storage)
Execution Carrier	 Backend Sandbox / Container	 Front-end Worker Secure Sandbox
Data Flow	 Browser => Backend => LLM	 Closed loop in browser, zero data egress
State Management	 Front/back-end split, needs sync	 Direct DOM & session-state access
Deployment	 Server-side deploy & ops	 Ships with the web app, ready to use
Evolution	 Unified server-side version	 Personalized on-device evolution



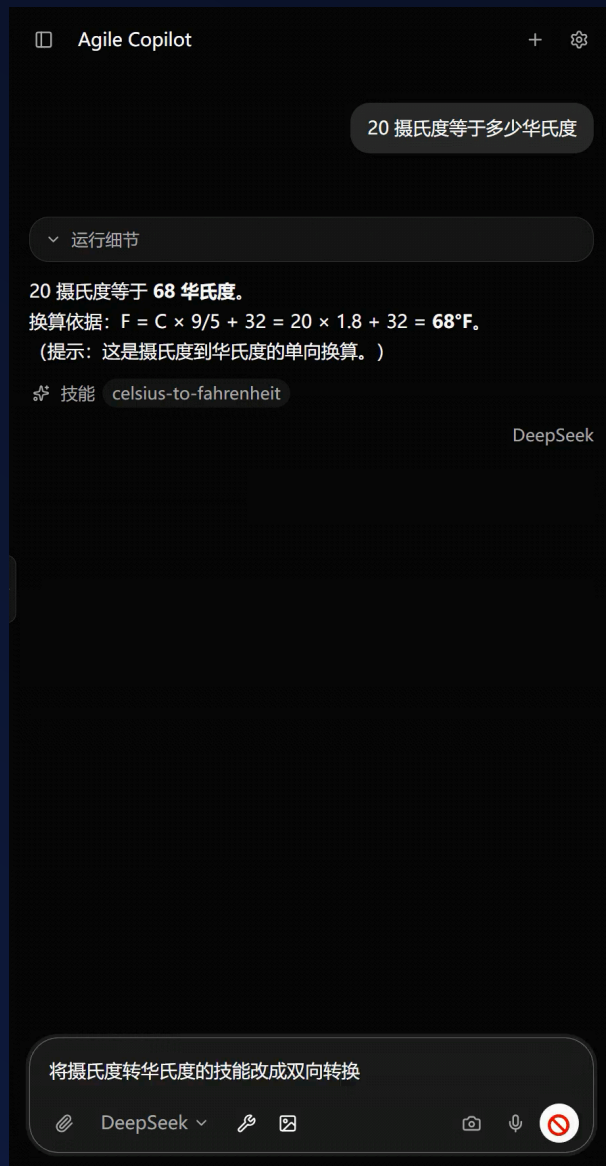
Create a Celsius-to-Fahrenheit skill



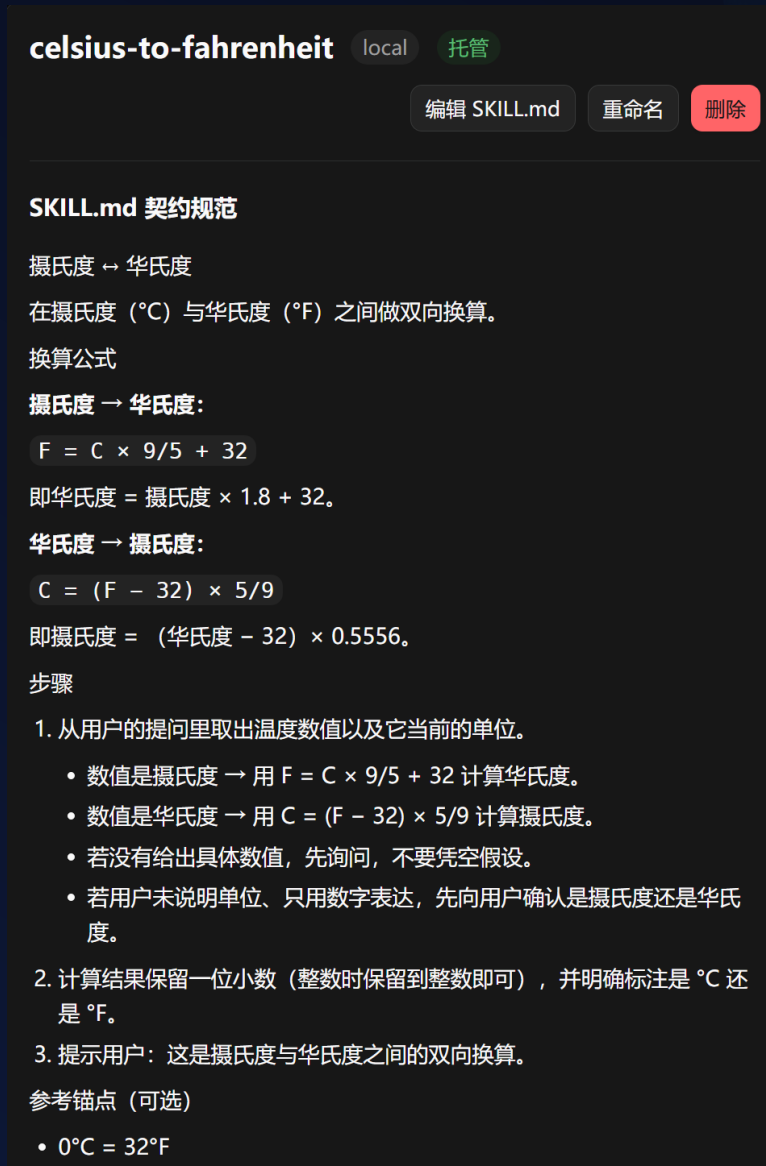
Auto-generate skill document



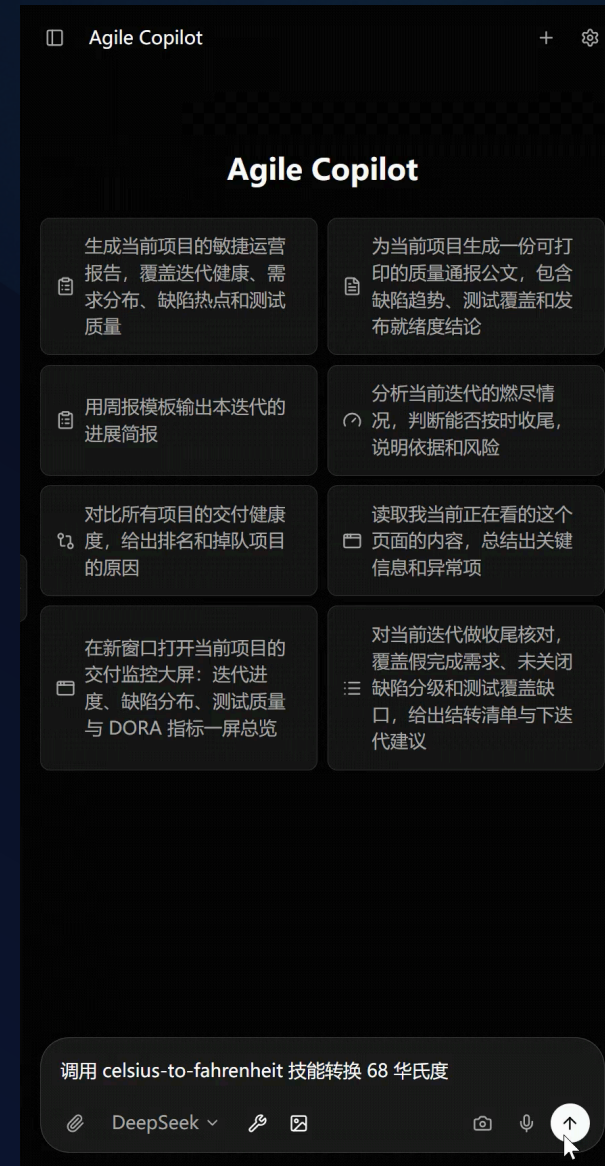
Ready to invoke immediately



Change the skill to two-way conversion



Skill content auto-updates



Two-way conversion goes live



View the temperature-conversion skill document in the library

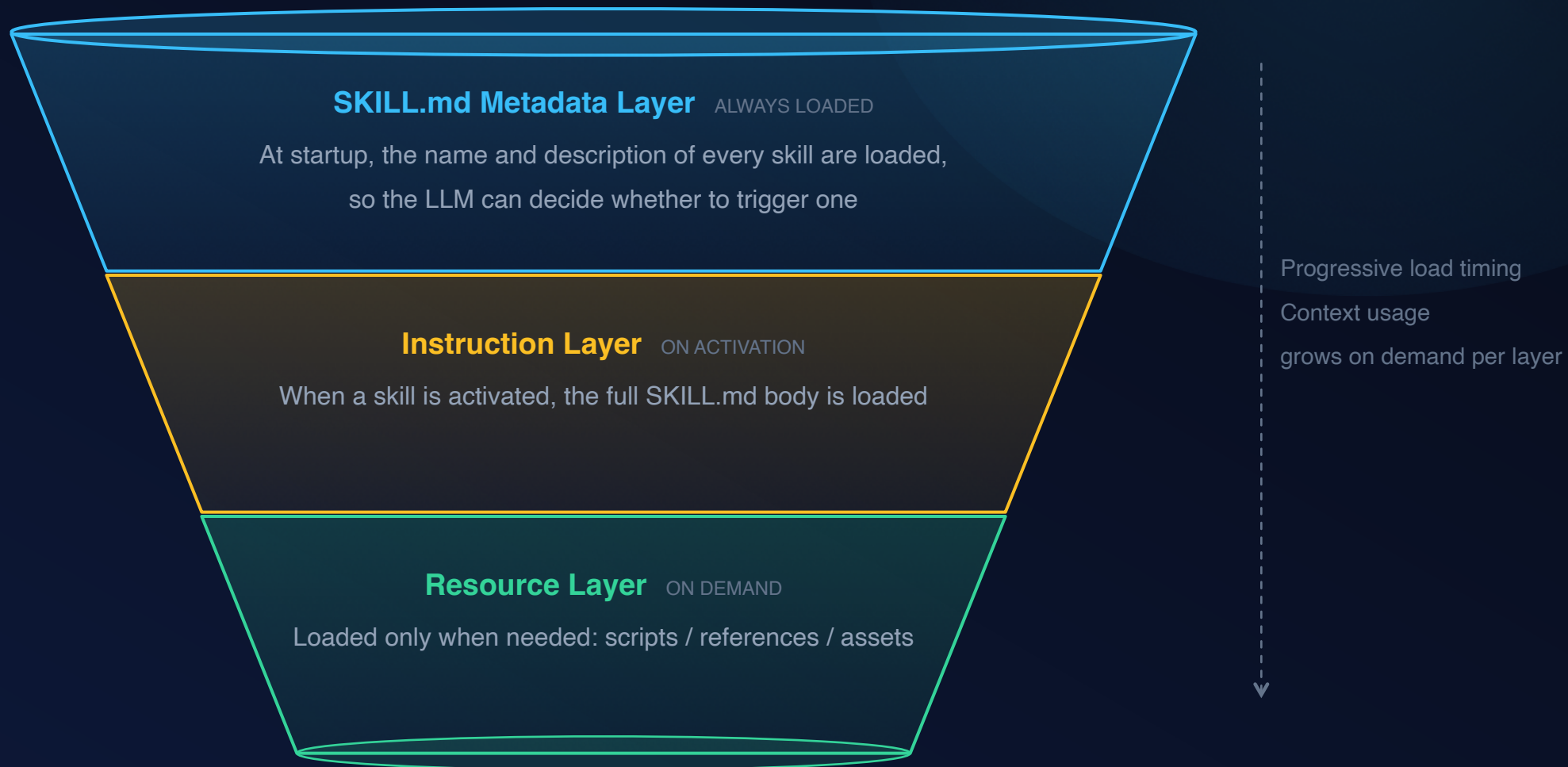


Inspect skills with OPFS Explorer in browser DevTools

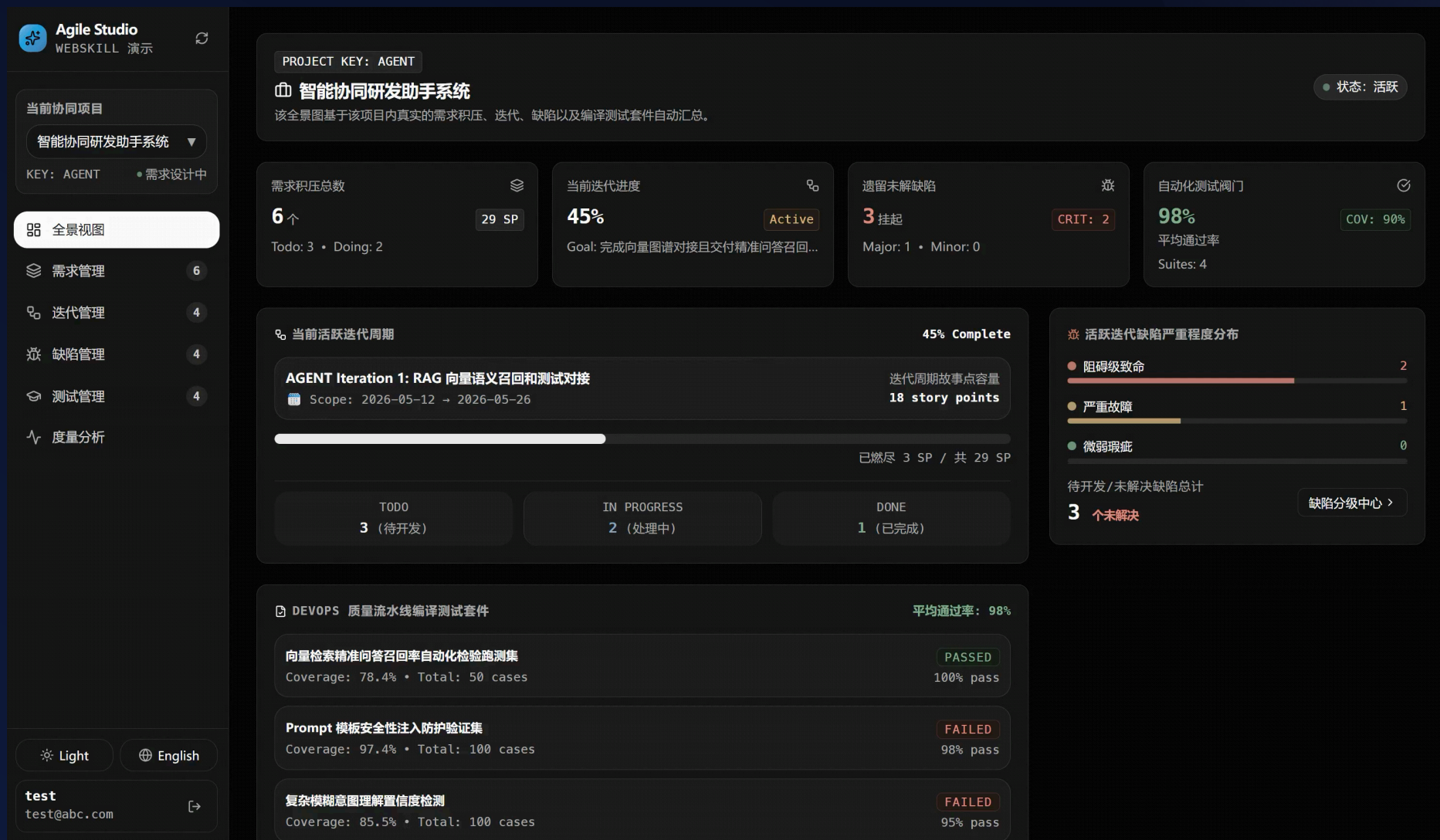
WebSkill Follows the Standard Agent Skills Protocol



The WebSkill directory specification fully complies with the Agent Skill standard protocol



Architecture for progressive skill loading



Owns three projects at once

Reports progress biweekly

Briefings follow a fixed template

Data must be collected per template

Then analysis and conclusions are written



编辑器

编辑技能文件, 保存即校验 SKILL.md 契约。

技能资源管理器

新建技能

过滤技能...

📁 📄 🗑️

agile-ops-dashboard

scripts

run.js

SKILL.md

- agile-ops-screen
- agile-slide-deck
- authored-bulletin
- authored-screen
- authored-slides
- bug-screenshot-triage
- celsius-to-fahrenheit
- cross-project-health
- quality-bulletin
- requirement-doc-digest
- sprint-closeout
- sprint-progress-report
- sprint-weekly-brief

agile-ops-dashboard local 本地

编辑 SKILL.md

SKILL.md 契约规范

何时使用

当用户要求敏捷运营报告、运营小结或项目交付健康的会话内一页总览时使用。与 agile-ops-screen (独立窗口的深蓝监控大屏) 区分: 本技能的报告直接渲染在对话框里。

输入

- projectId (可选): 缺省取当前选中项目。

步骤

- 调用 agile-ops-dashboard__run (可带 projectId)。脚本经 fetchData 读取迭代、需求、缺陷、测试套件与 DORA 指标, 一次渲染出报告 surface。
- 只解说头条结论 (一两句)。报告本身是交付物——不要在聊天里复述上面的数字。

结构

目录完整度——技能声明了哪些约定子目录 (scripts/references/resources/assets)。

🕒 scripts 已声明 🕒 references 未声明 🕒 resources 未声明 🕒 assets 未声明

脚本工具

技能激活时给模型脚本清单——与模型看到的同源、同工具名。这里未暴露的脚本, 模型也调不到。

文件	工具名	输入 schema
scripts/run.js	agile-ops-dashboard__run	not loaded yet

元数据

>

Briefing content is generated from a template preset in the skill's Run function

Generate the progress briefing in one click

Agile Studio - WebSkill Demo

localhost:4173/demo

Agile Studio
WEBSKILL 演示

当前协同项目
低代码自动化交付流...

KEY: DEVOPS • 架构优化中

全景视图

需求管理 7

迭代管理 4

缺陷管理 5

测试管理 4

度量分析

Light English

test
test@abc.com

PROJECT KEY: DEVOPS

低代码自动化交付流水线

该全景图基于该项目内真实的需求积压、迭代、缺陷以及编译测试套件自动汇总。

状态: 活跃

需求积压总数 7个 52 SP
Todo: 3 • Doing: 2

当前迭代进度 60% Active
Goal: 打通可视化拓...

遗留未解缺陷 4 挂起 CRIT: 3
Major: 1 • Minor: 0

自动化测试闸门 85% COV: 88%
平均通过率
Suites: 4

当前活跃迭代周期 60% Complete
CD Iteration 1: 云原生交付拓扑流打通 迭代周期故事点容量
Scope: 2026-05-10 → 2026-05-24 40 story points
已燃尽 13 SP / 共 52 SP

TODO 3 (待开发) IN PROGRESS 2 (处理中) DONE 2 (已完成)

活跃迭代缺陷严重程度分布
阻碍级致命 3
严重故障 1
微弱瑕疵 0
待开发/未解决缺陷总计 4 个未解决
缺陷分级中心

DEVOPS 质量流水线编译测试套件 平均通过率: 85%
拓扑节点连线互操作在各种边界拓扑排版场景压力测试 PASSED
Coverage: 91.2% • Total: 35 cases 100% pass
K8s部署Deployment故障恢复集成测试套 FAILED
Coverage: 82.5% • Total: 10 cases 50% pass
大规模多阶段并发构建压测验证 FAILED
Coverage: 80% • Total: 40 cases 90% pass

Agile Copilot

Agile Copilot

给我当前页面的速览总结: 我在哪个屏幕、正看哪个项目, 以及它的需求池规模、迭代进度、未关闭缺陷和测试质量头条数据

请运行 celsius-to-fahrenheit 技能

生成当前项目的敏捷运营报告, 覆盖迭代健康、需求分布、缺陷热点和测试质量

为当前项目生成一份可打印的质量通报公文, 包含缺陷趋势、测试覆盖和发布就绪度结论

用周报模板输出本迭代的进展简报

分析当前迭代的燃尽情况, 判断能否按时收尾, 说明依据和风险

对比所有项目的交付健康度, 给出排名和掉队项目的原因

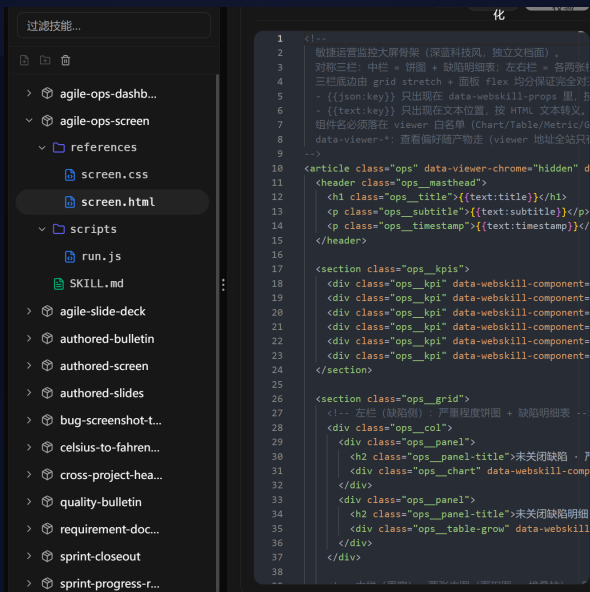
读取我当前正在看的这个页面的内容, 总结出关键信息和异常项

在新窗口打开当前项目的交付监控大屏: 迭代进度、缺陷分布、测试质量与 DORA 指标一屏总览

对当前迭代做收尾核对, 覆盖假完成需求、未关闭缺陷分级和测试覆盖缺口, 给出结转清单与下迭代建议

发送一条消息...

DeepSeek



Rendered from a preset HTML + CSS template

The screenshot displays the Agile Studio web application interface. The browser address bar shows 'localhost:4173/demo'. The application has a dark theme and a sidebar on the left with navigation options: '需求管理' (7), '迭代管理' (4), '缺陷管理' (5), '测试管理' (4), and '度量分析'. The main content area is divided into several sections:

- PROJECT KEY: DEVOPS** and **低代码自动化交付流水线** (Low-code automated delivery pipeline) with a status of '活跃' (Active).
- 需求积压总数** (Total demand backlog): 7 items, 52 SP. Sub-items: Todo: 3, Doing: 2.
- 当前迭代进度** (Current iteration progress): 60% (Active). Goal: 打通可视化拓... (Connect visualization expansion...).
- 遗留未解缺陷** (Legacy unresolved defects): 4 items. Sub-items: Major: 1, Minor: 0.
- 自动化测试闸门** (Automated test gate): 85% (Average pass rate), COV: 88% (Coverage). Suites: 4.
- 当前活跃迭代周期** (Current active iteration cycle): 60% Complete. **CD Iteration 1: 云原生交付拓扑流打通** (Cloud-native delivery topology flow connection). Scope: 2026-05-10 → 2026-05-24. 40 story points. Progress bar shows '已燃尽 13 SP / 共 52 SP' (Burned 13 SP / Total 52 SP).
- 活跃迭代缺陷严重程度分布** (Active iteration defect severity distribution):
 - 阻碍级致命 (Blocking critical): 3
 - 严重故障 (Severe fault): 1
 - 微弱瑕疵 (Weak flaw): 0
- 待开发/未解决缺陷总计** (Total defects to be developed/unsolved): 4 items. **缺陷分级中心** (Defect classification center).
- DEVOPS 质量流水线编译测试套件** (DEVOPS quality pipeline compilation test suite) with an average pass rate of 85%.
 - 拓扑节点连线互操作在各种边界拓扑排版场景压力测试** (Topology node connection interoperability in various boundary topology layout scenarios pressure test): PASSED. Coverage: 91.2% • Total: 35 cases. 100% pass.
 - K8s部署Deployment故障恢复集成测试套** (K8s deployment Deployment fault recovery integration test suite): FAILED. Coverage: 82.5% • Total: 10 cases. 50% pass.
 - 大规模多阶段并发构建压测验证** (Large-scale multi-stage concurrent build stress test verification): FAILED. Coverage: 80% • Total: 40 cases. 90% pass.

The right sidebar features the **Agile Copilot** section with a list of prompts:

- 给我当前页面的速览总结: 我在哪个屏幕、正看哪个项目, 以及它的需求池规模、迭代进度、未关闭缺陷和测试质量头条数据
- 请运行 celsius-to-fahrenheit 技能
- 生成当前项目的敏捷运营报告, 覆盖迭代健康、需求分布、缺陷热点和测试质量
- 为当前项目生成一份可打印的质量通报公文, 包含缺陷趋势、测试覆盖和发布就绪度结论
- 用周报模板输出本迭代的进展简报
- 分析当前迭代的燃尽情况, 判断能否按时收尾, 说明依据和风险
- 对比所有项目的交付健康度, 给出排名和掉队项目的原因
- 读取我当前正在看的这个页面的内容, 总结出关键信息和异常项
- 在新窗口打开当前项目的交付监控大屏: 迭代进度、缺陷分布、测试质量与 DORA 指标一屏总览
- 对当前迭代做收尾核对, 覆盖假完成需求、未关闭缺陷分级和测试覆盖缺口, 给出结转清单与下迭代建议

At the bottom right, there is a '发送一条消息...' (Send a message...) input field with a 'DeepSeek' logo and a '发送' (Send) button.

The screenshot displays the Agile Studio web application interface. The browser window shows the URL 'localhost:4173/demo'. The application has a dark theme and a sidebar on the left with navigation options: '需求管理' (7), '迭代管理' (4), '缺陷管理' (5), '测试管理' (4), and '度量分析'. The main content area is titled 'PROJECT KEY: DEVOPS' and '低代码自动化交付流水线' (Low-code automated delivery pipeline). It features several key metrics: '需求积压总数' (7), '当前迭代进度' (60%), '遗留未解缺陷' (4), and '自动化测试通过率' (85%). Below these, there's a section for '当前活跃迭代周期' (60% Complete) showing 'CD Iteration 1: 云原生交付拓扑流打通' with a scope from 2026-05-10 to 2026-05-24. A progress bar indicates '已燃尽 13 SP / 共 52 SP'. To the right, a '活跃迭代缺陷严重程度分布' (Active iteration defect severity distribution) chart shows 3 critical, 1 severe, and 0 minor defects. At the bottom, a 'DEVOPS 质量流水线编译测试套件' (DEVOPS quality pipeline compilation test suite) section shows three test cases: '拓扑节点连线互操作在各种边界拓扑排版场景压力测试' (PASSED), 'K8s部署Deployment故障恢复集成测试套' (FAILED), and '大规模多阶段并发构建压测验证' (FAILED). On the right side of the interface, there's an 'Agile Copilot' section with a chat interface. The chat history includes prompts like '给我当前页面的速览总结' and '请运行 celsius-to-fahrenheit 技能'. The bottom of the chat interface has a '发送一条消息...' (Send a message...) input field and a 'DeepSeek' logo.

Generate the project analysis slide in one click

The screenshot displays the Agile Studio web application interface. The browser tabs show 'Agile Studio - WebSkill Demo' and several 'Agile Studio document view' tabs. The address bar indicates 'localhost:4173/demo'.

Agile Studio WEBSKILL 演示

当前协同项目

低代码自动化交付流... ▼

KEY: DEVOPS • 架构优化中

全景视图

- 需求管理 7
- 迭代管理 4
- 缺陷管理 5
- 测试管理 4
- 度量分析

PROJECT KEY: DEVOPS

低代码自动化交付流水线 状态: 活跃

该全景图基于该项目内真实的需求积压、迭代、缺陷以及编译测试套件自动汇总。

需求积压总数 7个 52 SP
Todo: 3 • Doing: 2

当前迭代进度 60% Active
Goal: 打通可视化拓...

遗留未解缺陷 4 挂起 CRIT: 3
Major: 1 • Minor: 0

自动化测试阀门 85% COV: 88%
平均通过率
Suites: 4

当前活跃迭代周期 60% Complete

CD Iteration 1: 云原生交付拓扑流打通 迭代周期故事点容量 40 story points
Scope: 2026-05-10 → 2026-05-24

已燃尽 13 SP / 共 52 SP

TODO 3 (待开发) **IN PROGRESS** 2 (处理中) **DONE** 2 (已完成)

活跃迭代缺陷严重程度分布

- 阻碍级致命 3
- 严重故障 1
- 微弱瑕疵 0

待开发/未解决缺陷总计 4 个未解决

DEVOPS 质量流水线编译测试套件 平均通过率: 85%

拓节点连线互操作在各种边界拓扑排版场景压力测试 PASSED
Coverage: 91.2% • Total: 35 cases 100% pass

K8s部署Deployment故障恢复集成测试套 FAILED
Coverage: 82.5% • Total: 10 cases 50% pass

大规模多阶段并发构建压测验证 FAILED
Coverage: 80% • Total: 40 cases 90% pass

Agile Copilot

给我当前页面的速览总结: 我在哪个屏幕、正看哪个项目, 以及它的需求池规模、迭代进度、未关闭缺陷和测试质量头条数据

请运行 celsius-to-fahrenheit 技能

生成当前项目的敏捷运营报告, 覆盖迭代健康、需求分布、缺陷热点和测试质量

为当前项目生成一份可打印的质量通报公文, 包含缺陷趋势、测试覆盖和发布就绪度结论

用周报模板输出本迭代的进展简报

分析当前迭代的燃尽情况, 判断能否按时收尾, 说明依据和风险

读取我当前正在看的这个页面的内容, 总结出关键信息和异常项

在新窗口打开当前项目的交付监控大屏: 迭代进度、缺陷分布、测试质量与 DORA 指标一屏总览

对当前迭代做收尾核对, 覆盖假完成需求、未关闭缺陷分级和测试覆盖缺口, 给出结转清单与下迭代建议

请对所有项目做一次横向的综合分析, 生成用于汇报使用的PPT

DeepSeek

First generate two Slide pages: requirements & iterations, then "Add one defects page"

**Only .ts and .js file types**

Natively supported by the web environment

**Must export a function named run**

The unified execution entry point

**Must provide inputSchema**

Explicit declaration automatic type inference

3 Ways to Infer inputSchema Types SCHEMA INFERENCE**Explicit Schema** Best for complex parameters

```
export const inputSchema = {  
  type: 'object',  
  properties: {  
    projectId: { type: 'string' }  
  },  
  required: ['projectId']  
};
```

TypeScript Inference Best for standard parameters

```
export interface Input {  
  projectId: string;  
  period?: 'week' | 'month';  
}
```

JSDoc Annotations Best for JS script compatibility

```
/**  
 * @param {string} projectId  
 * @param {'week'|'month'} [period]  
 */  
export function run(args) { ... }
```

**Follows the MCP Standard Return Format**

```
{ content: [{ type: 'text' | 'resource', ... }], isError?: boolean }
```



Artifacts can be stored in assets



Invoke as endpoint:toolName or mcp#toolName



Traditional Approach

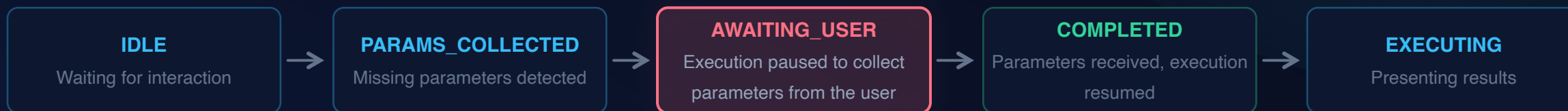
The LLM can only ask in Markdown text — limited interactivity



WebSkill + Generative UI

The LLM outputs structured JSON Schema, rendered in real time

Human-Interaction State Flow



UIBridge — a bridge for swapping generative UI frameworks



UIBridge backplane: a stable protocol decoupling execution from rendering



json-render-react



Vercel AI SDK



OpenUI



A2UI

Execution decoupled from rendering — frameworks are pluggable

设置

概览

技能

运行

治理

连接

大模型

MCP 端点

WebMCP 工具

页面技能

设置

MCP 端点

远程 MCP 服务器的连接状态与暴露的工具。

添加端点

端点	URL	状态	工具	操作
agile-page	in-process://agile-page (page tools & temporary skills) streamable-http 出网严格	已连接	10 个工具	

端点注册表

已连接端点暴露的全部工具，聚合为一个注册表视图。技能在运行中调用这些工具。

agile-page 10 个工具

list_requirements

List requirements of a project, optionally filtered by status.

已启用 按需启用 已禁用

list_sprints

List sprints (iterations) of a project.

已启用 按需启用 已禁用

list_bugs

List defects of a project, optionally filtered by severity.

已启用 按需启用 已禁用

list_test_suites

List test suites of a project with coverage and pass rate.

已启用 按需启用 已禁用

get_dora_metrics

Get the workspace-level DORA metrics (lead time, cycle time, deployment frequency, failure rate).

已启用 按需启用 已禁用

环境: 浏览器 OPFS

Define data-fetching tools on the page using the standard MCP protocol

设置

概览

技能

运行

治理

连接

大模型

MCP 端点

WebMCP 工具

页面技能

设置

WebMCP 工具

当前页面通过浏览器 WebMCP 通道暴露给模型的工具。

WebMCP 实验性

WebMCP 让浏览器通过标准通道暴露页面工具。实验性: API 可能变化。

启用 WebMCP

agile_get_current_view

http://localhost:4173

不可信内容

获取 Agile Studio 当前视图上下文: 用户所在屏幕、当前选中项目, 以及头条数据 (需求池规模、活跃迭代进度、未关闭缺陷、测试质量与 DORA 指标)。

已启用 按需启用 已禁用

agile_get_projects

http://localhost:4173

不可信内容

工作区内的全部项目: 编号、名称、状态与负责人。

已启用 按需启用 已禁用

agile_get_requirements

http://localhost:4173

不可信内容

当前项目 (或 projectId 指定项目) 的需求池条目, 含优先级、状态与故事点。

已启用 按需启用 已禁用

agile_get_sprints

http://localhost:4173

不可信内容

当前项目的迭代列表, 含起止日期、状态与完成进度。

已启用 按需启用 已禁用

agile_get_bugs

http://localhost:4173

不可信内容

当前项目的缺陷清单, 含严重级别、状态、经办人与创建时间。

已启用 按需启用 已禁用

agile_get_test_suites

http://localhost:4173

不可信内容

当前项目的测试套件, 含用例数、覆盖率与通过率。

已启用 按需启用 已禁用

agile_get_metrics

http://localhost:4173

不可信内容

工作区级 DORA 四项指标时间序列 (不按项目切分)。

已启用 按需启用 已禁用

环境: 浏览器 OPFS

Define data-fetching tools on the page using experimental WebMCP

mcp-server.ts — Registration via the MCP standard protocol

```
import { McpServer } from '@modelcontextprotocol/server';
import { Client } from '@modelcontextprotocol/sdk/client/index.js';
import { MessageChannelServerTransport,
  MessageChannelClientTransport } from './channel.ts';

const server = new McpServer({ name: 'greeting-server', version: '1.0.0' });

server.registerTool('greet', {
  description: 'Greet someone by name',
  inputSchema: z.object({ name: z.string() }),
},
async ({ name }) => ({
  content: [{ type: 'text', text: `Hello, ${name}!` }],
}));

const serverTransport = new MessageChannelServerTransport(endpoint);
const clientTransport = new MessageChannelClientTransport(endpoint);
const client = new Client({ name: 'greeting-client', version: '1.0.0' });

await serverTransport.listen();
await server.connect(serverTransport);
await client.connect(clientTransport);

const result = await client.listTools();
const response = await client.callTool({
  name: 'greet', arguments: { name: 'Jack' } });
```

Naming convention endpoint:toolName

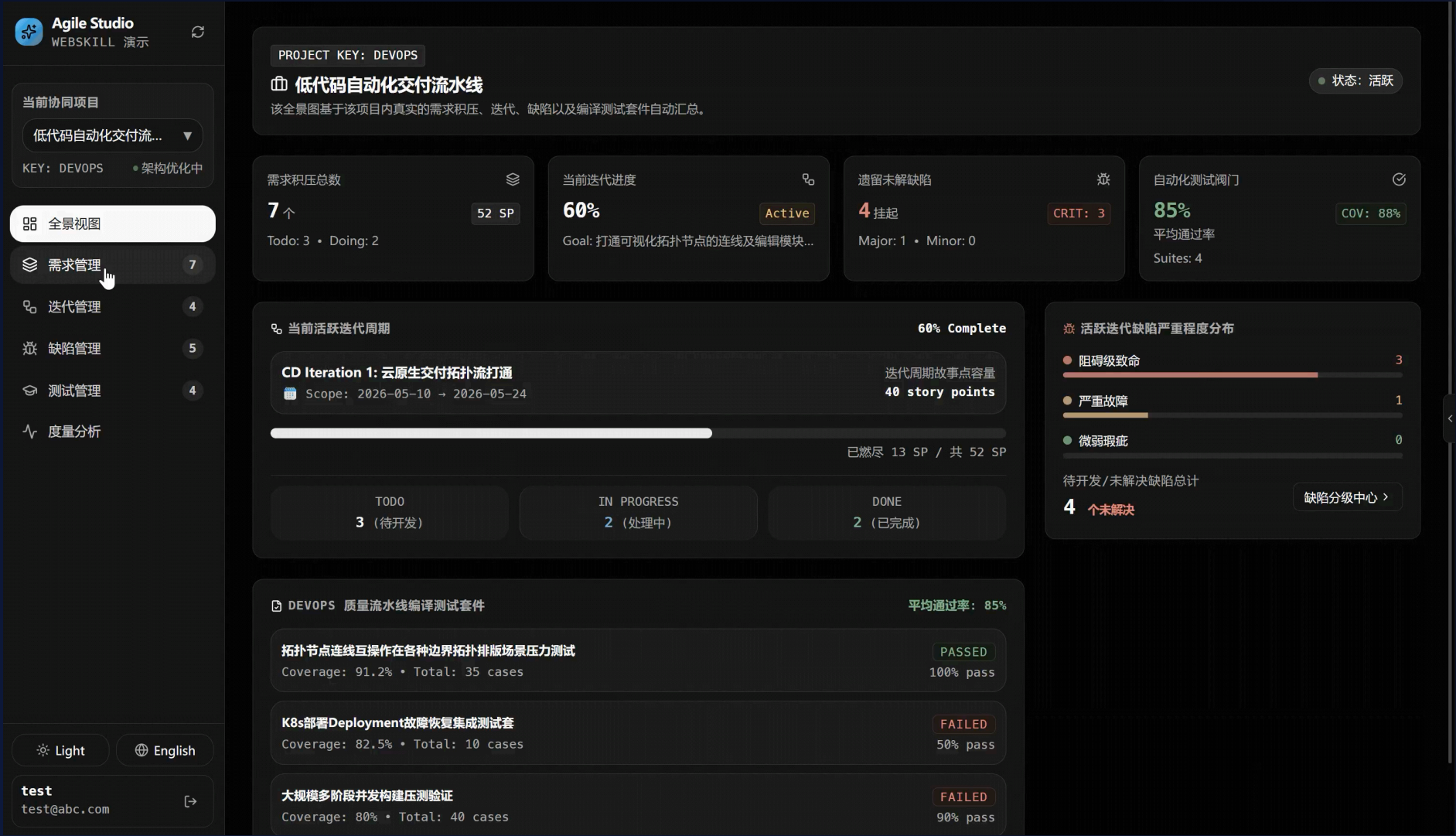
chrome-api.js — Chrome experimental API

```
const controller = new AbortController();

document.modelContext.registerTool({
  name: "fetch_page_summary",
  description: "Get the current page title and body text for page analysis",
  inputSchema: {
    type: "object",
    properties: {
      maxLength: { type: "integer",
        description: "Maximum length of the returned summary", default: 100 }
    }
  },
  execute: async ({ maxLength }) => {
    const title = document.title;
    const bodyText = document.body.innerText.slice(0, maxLength);
    return {
      content: [{ type: "text",
        text: `Title: ${title}\nSummary: ${bodyText}` } ]
    };
  }, { signal: controller.signal });

// Query and invoke registered page-level tools
const tools = await document.modelContext.listTools();
const response = await document.modelContext.executeTool(
  "fetch_page_summary", JSON.stringify({ maxLength: 150 }));
```

Naming convention mcp#toolName



Create a new requirement

Enter it into requirement management

Then split it into iterations

Finally add test cases

Every requirement is filled in by hand

Agile Studio
WEBSKILL 演示

当前协同项目
智云协同核心业务平台
KEY: CORE 迭代活跃中

全景视图

需求管理 9

迭代管理 5

缺陷管理 7

测试管理 6

度量分析

Light English

test
test@abc.com

需求管理看板
项目「智云协同核心业务平台」需求管理看板

搜索需求 ID、经办人、史诗模块、关键字... 过滤全部优先级

需求 ID	概况信息与描述	史诗模块	优先级	状态
CORE-101	企业单点登录 (SSO) 安全保护对接 接入企业级安全 LDAP 校验与 OAuth 2.0 服务, 强化账户防碰撞验证, 支持多维度 RBAC 数据级越权保护。 企业单点登录 (SSO) 安全保护对接.docx	用户中心安全	High	
CORE-102	团队敏捷工作台流式大模型对话卡片渲染 开发流式生成式 UI 卡片渲染机制, 深度绑定后台结构化 DSL 规则, 支持在线组件级拖拽交互。 团队敏捷工作台流式大模型对话卡片渲染.docx	智能卡片模块	High	
CORE-103	项目多重视图与排期时间轴切换 开发看板排课树、需求列表树、以及跨模块甘特图的多重视图配置, 优化大数据量下拖拽帧数。 项目多重视图与排期时间轴切换.docx	交互看板模块	Medium	
CORE-104	双机沙箱数据库多租户数据落盘加密 在全局 ORM 执行器层插入租户 ID 空间过滤器, 使用 AES-256 加密落盘数据, 满足金保级审计规范。 双机沙箱数据库多租户数据落盘加密.docx	数据存储层安全	High	
CORE-105	敏捷持续跑测试套件自动化重试机制 优化 CI 流水线测试套件并发环境, 对低概率跑丢的测试加入最大 3 次指数退避重试保护。 敏捷持续跑测试套件自动化重试机制.docx	效能质量跑测	Low	
CORE-106	跨租户数据同步接口 开发支持多租户间数据准实时同步的 API 层, 支持增量消费, 确保一致性。	开放平台接口	Medium	

Agile Copilot

读取我当前正在看的需求列表里带附件的那几条需求文档, 抽出验收标准, 并列成文档里有、需求单里缺的条目

检查当前项目的需求状态和迭代阶段是否自治, 把不一致的列出来

把当前页面上处于 Todo 且优先级为 High 的需求, 逐条挪进进行中状态

根据我的想法规划新需求、迭代任务和测试用例, 带我到各个页面逐个创建落地

请运行 celsius-to-fahrenheit 技能

生成当前项目的敏捷运营报告, 覆盖迭代健康、需求分布、缺陷热点和测试质量

为当前项目生成一份可打印的质量通报公文, 包含缺陷趋势、测试覆盖和发布就绪度结论

用周报模板输出本迭代的进展简报

分析当前迭代的燃尽情况, 判断能否按时收尾, 说明依据和风险

读取我当前正在看的这个页面的内容, 总结出关键信息和异常项

发送一条消息...
DeepSeek 附件 语音 发送

The LLM splits the idea into iterations and test cases auto-entered

Agile Studio

WEBSKILL 演示

当前协同项目

智云协同核心业务平台

KEY: CORE

迭代活跃中

全景视图

需求管理13

迭代管理5

缺陷管理7

测试管理7

度量分析

Light

English

test

test@abc.com

需求管理看板

项目「智云协同核心业务平台」需求管理看板

创建需求/故事

搜索需求 ID、经办人、史诗模块、关键字...

过滤全部优先级

需求 ID	概况信息与描述	史诗模块	优先级	状态	经办人	点数	操作
CORE-101	企业单点登录 (SSO) 安全保护对接 接入企业级安全 LDAP 校验与 OAuth 2.0 服务, 强化账户防碰撞验证, 支持多维度 RBAC 数据级越权保护。 企业单点登录 (SSO) 安全保护对接.docx	用户中心安全	High	IN PROGRESS	林晨	5	
CORE-102	团队敏捷工作台流式大模型对话卡片渲染 开发流式生成式 UI 卡片渲染机制, 深度绑定后台结构化 DSL 规则, 支持在线组件级拖拽交互。 团队敏捷工作台流式大模型对话卡片渲染.docx	智能卡片模块	High	IN PROGRESS	孙雪	8	
CORE-103	项目多重视图与排期时间轴切换 开发看板排课树、需求列表树、以及跨模块甘特图的多重视图配置, 优化大数据量下拖拽帧数。 项目多重视图与排期时间轴切换.docx	交互看板模块	Medium	TO DO	周杰	3	
CORE-104	双机沙箱数据库多租户数据落盘加密 在全局 ORM 执行器层插入租户 ID 空间过滤器, 使用 AES-256 加密落盘数据, 满足金保级审计规范。 双机沙箱数据库多租户数据落盘加密.docx	数据存储层安全	High	DONE	陈俊宇	5	
CORE-105	敏捷持续跑测试套件自动化重试机制 优化 CI 流水线测试套件并发环境, 对低概率跑丢的测试加入最大 3 次指数退避重试保护。 敏捷持续跑测试套件自动化重试机制.docx	效能质量跑测	Low	DONE	郑伟	2	
CORE-106	跨租户数据同步接口 开发支持多租户间数据准实时同步的 API 层, 支持增量消费, 确保一致性。	开放平台接口	Medium	TO DO	赵海峰	5	

Every attachment must be downloaded; every screenshot opened one by one

Agile Copilot

Agile Copilot

读取我当前正在看的需求列表里带附件的那几条需求文档, 抽出验收标准, 并列出文档里有、需求单里缺的条目

检查当前项目的需求状态和迭代阶段是否自治, 把不一致的列出来

把当前页面上处于 Todo 且优先级为 High 的需求, 逐条挪进进行中状态

根据我的想法规划新需求、迭代任务和测试用例, 带我到各个页面逐个创建落地

请运行 celsius-to-fahrenheit 技能

生成当前项目的敏捷运营报告, 覆盖迭代健康、需求分布、缺陷热点和测试质量

为当前项目生成一份可打印的质量通报公文, 包含缺陷趋势、测试覆盖和发布就绪度结论

用周报模板输出本迭代的进展简报

分析当前迭代的燃尽情况, 判断能否按时收尾, 说明依据和风险

读取我当前正在看的这个页面的内容, 总结出关键信息和异常项

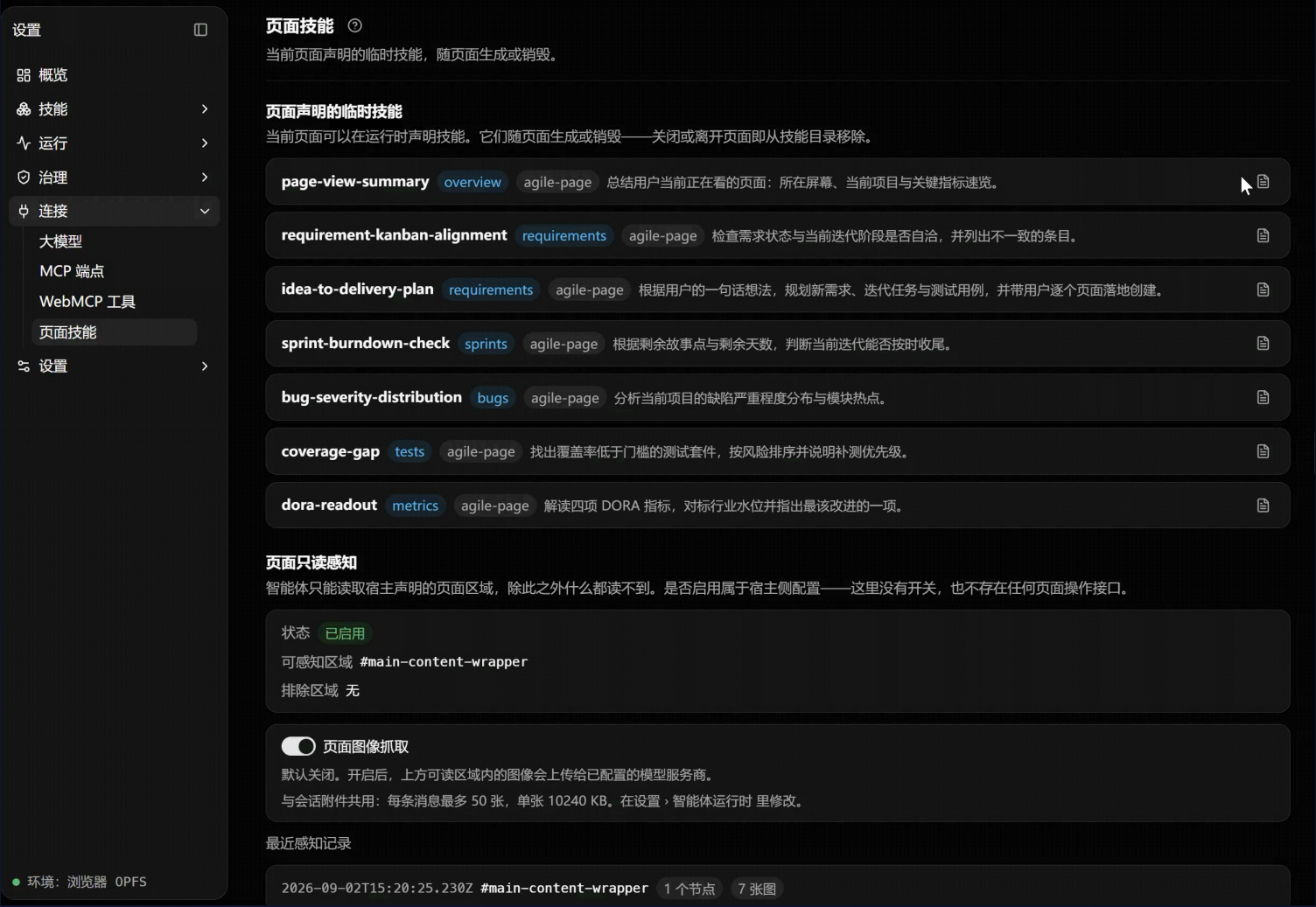
总结当前项目需求管理的所有附件文档和缺陷管理的所有截图

DeepSeek

Automatically review all attachments and screenshots



Each page has its own skill quick commands



Define page-specific skills for each page

Web Page Main Thread (MCP Server)

registerPrompt

maps to SKILL.md

registerTool

maps to scripts

registerResource

maps to references / assets

Use an MCP Server to dynamically declare ad-hoc skills on the web page



MessageChannel Transport (MCP standard transport protocol)

Worker Subthread (MCP Client)

client.listPrompts**client.getPrompt****client.listTools****client.callTool****client.listResources****client.readResource**

Use an MCP Client in the Worker subthread to discover and invoke ad-hoc skills



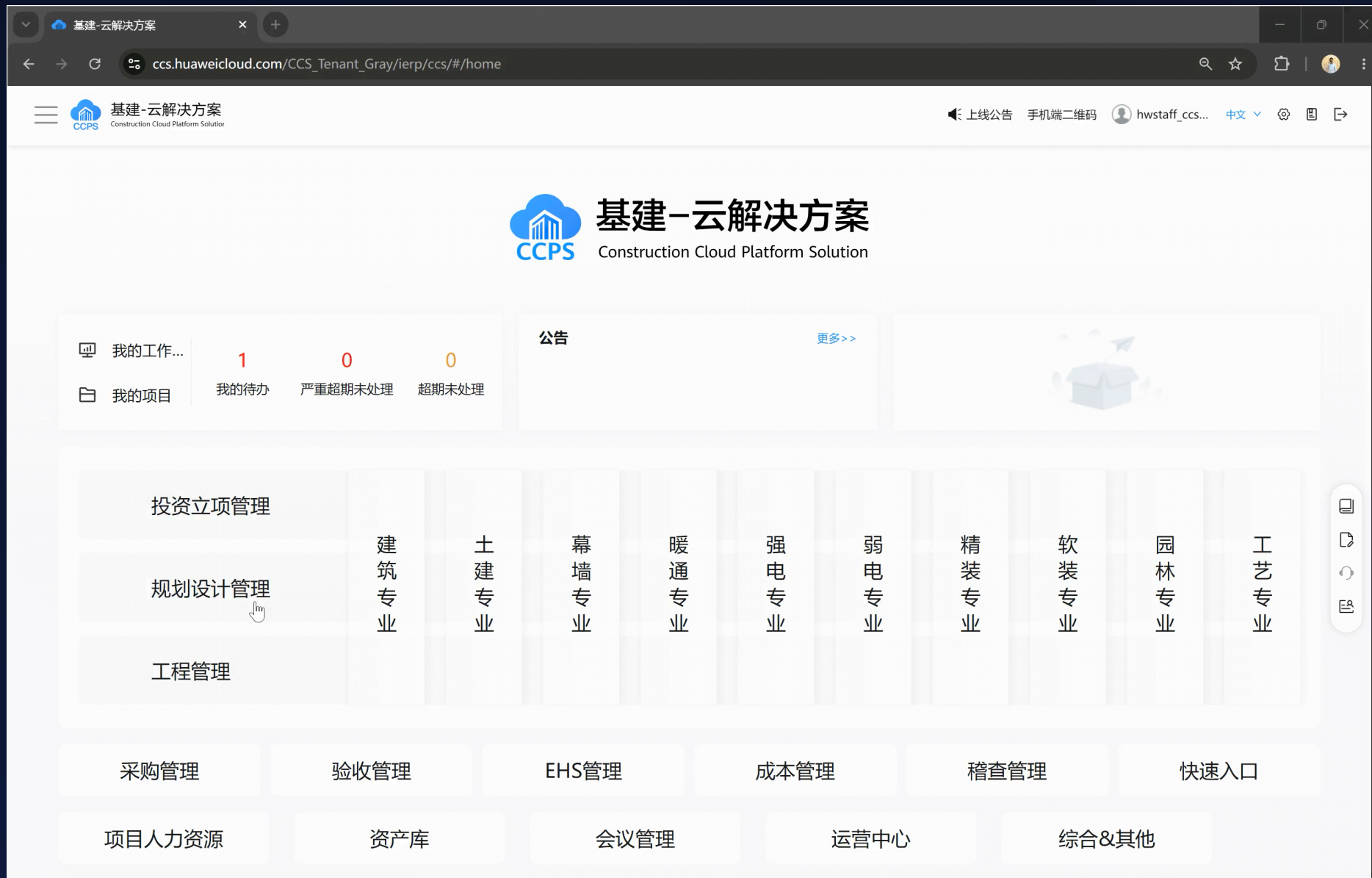
Persistent SKILL (OPFS)

Shipped or installed with the web app, stored in OPFS, persists across pages



Dynamic SKILL (Web Page)

Ad-hoc skills declared at runtime on the current page; they expire when the page closes



The screenshot displays the CCPS Construction Cloud Platform Solution web interface. The page features a header with the CCPS logo and navigation links. The main content area includes a dashboard with various management modules. A hand cursor is pointing at the '规划设计管理' (Planning Management) module.

基建-云解决方案
Construction Cloud Platform Solution

公告 更多>>

我的工作... 1 0 0
我的项目 我的待办 严重超期未处理 超期未处理

投资立项管理
规划设计管理
工程管理

建筑专业
土建专业
幕墙专业
暖通专业
强电专业
弱电专业
精装专业
软装专业
园林专业
工艺专业

采购管理 验收管理 EHS管理 成本管理 稽查管理 快速入口

项目人力资源 资产库 会议管理 运营中心 综合&其他

Every link pops a new window

Pages are hard to manage centrally

Data needed by skills is hard to obtain

The quality-issue database is huge

Analysis means exporting to Excel by hand

基建-云解决方案

全部专业

建筑 土建 幕墙 暖通 强电 弱电 精装 软装 园林 工艺

上线公告 手机端二维码 hwstaff_ccs... 中文

首页 > 全部专业 > 现场实施管理 > 质量管理 > 示例项目

下发整改 转交 整改进度 问题修改 查看流程图 查看审批记录 刷新 导出Excel 退出 刷新当前处理人 更新及时关闭 批量转交 新增巡检问题

按工程项目查询 示例项目 所有的

共21条 选择全部

共2页 第 1 页 20条/页

#	问题类型	问题状态	当前处理人	实际验收人	问题位置	问题标题	检查项 (合同/图纸/规范要求)	问题描述	问题等级	计划关闭日期	整改开始日期	问题关闭时
1	现场巡检	整改中	机电单位人员	基小建	3栋地下室空...	桥架盖板缺少卡扣		桥架盖板缺少卡扣	C	2026-05-15	2026-04-29	
2	现场巡检	整改中	机电单位人员	基小建	3栋地下室空...	压差开关取压管未固定		压差开关取压管未固定	C	2026-05-15	2026-04-29	
3	现场巡检	整改中	机电单位人员	基小建	3栋地下室空...	桥架缺失标识		桥架缺失标识	C	2026-05-15	2026-04-29	
4	现场巡检	整改中	机电单位人员	基小建	3栋屋面	线管端口未做防腐		线管端口未做防腐	C	2026-05-15	2026-04-29	
5	现场巡检	整改中	机电单位人员	基小建	4#	收口不规范, 要求全面...		收口不规范, 要求全面排查整改	B	2026-05-15	2026-04-29	
6	现场巡检	整改中	机电单位人员	基小建	7号楼屋面机房	风管影响防火门正常开...		风管影响防火门正常开启、关闭	B	2026-05-15	2026-04-29	
7	现场巡检	整改中	机电单位人员	基小建	5栋变电所	高低压柜进出线口未做...		高低压柜进出线口未做防火封堵, 存在重大安全隐患, 要求全面排...	B	2026-05-15	2026-04-29	
8	现场巡检	整改中	机电单位人员	基小建	5栋变电所	电房管控问题		电房内无人施工, 多处电柜未上锁, 存在重大安全隐患, 要求全...	B	2026-05-15	2026-04-29	
9	现场巡检	关闭	—	基小建	7栋屋面排烟...	防火门未安装、地面环...		防火门未安装、地面环氧未完工	A	2026-05-15	2026-04-29	2026-04-29
10	现场巡检	关闭	—	基小建	G地块地下室	半地下室送风机房隔壁...		半地下室送风机房隔壁露天井道墙面渗漏	B	2026-05-15	2026-04-29	2026-04-29
11	现场巡检	关闭	—	基小建	负二层管廊坡...	负二层管廊坡道口施工...		负二层管廊坡道口施工错误, 需整改。	B	2026-05-15	2026-04-29	2026-04-29
12	现场巡检	关闭	—	基小建	20栋1层	墙面真石漆施工, 洞口...		墙面真石漆施工, 洞口封堵未抹灰、腻子未施工	B	2026-05-15	2026-04-29	2026-04-29
13	现场巡检	关闭	—	基小建	地下室	防火卷帘门两侧砌筑封...		防火卷帘门两侧砌筑封堵未剔凿腻子, 存在开裂风险	B	2026-05-15	2026-04-29	2026-04-29
14	现场巡检	关闭	—	基小建	13栋	13 栋烟道根部未修补		13 栋烟道根部未修补	C	2026-05-15	2026-04-29	2026-04-29
15	现场巡检	关闭	—	基小建	13栋	柱子模板未拆除		柱子模板未拆除	C	2026-05-15	2026-04-29	2026-04-29
16	现场巡检	关闭	—	基小建	7号坡道	环氧漏做		环氧漏做	C	2026-05-15	2026-04-29	2026-04-29

QA Supervisor : One Sentence to Verify Photos Against Handover Standards

基本信息

项目名称*	单据名称*	单据编码*
示例项目	示例项目移交计划	YJJH-260428-003
二级计划名称*	计划开始时间	计划结束时间
土建工作面移交暖通	2025-03-20	2025-06-30
三级计划名称*	计划开始时间	计划结束时间
土建工作面移交暖通-地下主...	2025-03-20	2025-04-30

四级计划

导出全部

#	*四级计划名称	*计划移交开始时间	*计划移交完成	供应商信息
1	地下室空调机房移交	2025-03-20	2025-04-10	查看
2	地下室排风机房移交	2025-03-31	2025-04-30	查看

移交标准

#	移交标准	参考图片
1	1、结构顶板：已封闭（预留吊装通道除外），室内结构断水完成，无积水、...	2
2	2、墙体工程：砌筑及抹灰施工完成（设备运输通道除外）。	0
3	3、天花与墙面：抹灰、腻子及乳胶漆施工完成（含打磨），地面2.2米以下...	0
4	4、设备基础及排水设施：设备基础、排水沟、围堰等结构施工完成。	0
5	5、吸音房间吸音天面及墙面施工流程：a、天面：机电先完成支架安装，管...	0
6	6、地面环氧工程：设备投影线位置及周边150mm宽范围环氧面层完成，	0
7	7、地面湿作业及排水：地面湿作业完成，排水沟、集水坑结构施工完成，水...	0
8	8、混凝土水箱防水（如有）：防水施工完成，水箱预留套管完成。	0
9	9、楼梯（如有）：结构施工完成。	0

拍照

拍好的照片会作为附件放进输入框，你点发送才会发出去。

取消 拍照

Take photos on site — auto-checked against the handover standards on the page



The screenshot shows a web browser window with multiple tabs: 基建-云解决方案, 工程管理, 质量检查, 日常检查, and 现场巡检问题. The active tab is 现场巡检问题, displaying a form for reporting a quality issue. The form is titled "巡检验收问题" and contains sections for "基本信息" (Basic Information) and "问题登记" (Issue Registration). The "问题登记" section is filled out with the following details:

问题编号	问题提出方	质量问题
WT_XJ-20260903-000001	工程质量管理部	质量

Additional form fields include:

- 合同名称: 中建科工试用项目
- 分部名称: [empty]
- 专业经理: [empty]
- 当前环节: [empty]
- 当前处理人: [empty]
- 所属专业: 软装专业
- 问题等级: C
- 计划关闭日期: 2026-09-10
- 问题标题: 2B12会议室桌子腿坏了
- 问题描述: 2B12会议室的桌子腿坏了, 10号要修好。

At the bottom, there is a section for "现场图片(最多录入4张)" with an "上传文件" button and a note: "支持ctrl+v粘贴截图 支持扩展名: .JPG, .JPEG, .PNG, .GIF".

On the right side, a "WebSkill Extension" panel is open, displaying a table of extracted information:

问题等级	C
问题位置	2B12会议室
问题属性	功能问题
计划关闭日期	2026-09-10 (10号修好)
问题标题	2B12会议室桌子腿坏了
问题描述	2B12会议室的桌子腿坏了, 10号要修好。

Below the table, the extension provides a "说明:" (Note) section:

- 合同名称 按照您的要求没有填写。
- 问题编号 (WT_XJ-20260903-000001) 由系统自动生成。
- 由于是"桌子腿坏了"这类家具损坏问题, 我选择了 **软装专业** 和 **功能问题** 属性; 问题等级选择了 **C** (一般问题)。

A summary text follows: "表单已经填写完成, 但目前尚未点击'保存'。需要我帮您点击保存吗? 另外, 如果您想调整某些字段的选择 (比如所属专业、问题等级、问题属性), 也可以告诉我。"

At the bottom of the extension panel, there is a button labeled "将上述操作固化成技能" (Solidify the above operations into a skill) and a "DeepSeek" logo.

设置

概览

技能

运行

治理

连接

设置

智能体运行时

沙箱与安全

生成式 UI

快捷指令

信任与签名

隐私与用户画像

外观

关于与诊断

隐私与用户画像

智能体是否从你的回答里学习，以及它在本机保留了什么。

从你的回答里学习

记录智能体问过你什么、你答了什么，再总结成一份画像，在后续对话里发给模型。默认关闭：你在技能表单里填的每一项都会成为模型输入。

加密存储

用浏览器不可导出的密钥对行为记录与画像做静态加密。关闭后，任何能访问这个浏览器配置的人都能直接读到。

注入模型的画像上限 (KB)

超出后先丢较旧、置信度较低的条目。

4

行为记录条数上限

超出后丢弃最早的记录。

500

User built a temperature conversion skill that supports both Celsius-to-Fahrenheit and Fahrenheit-to-Celsius.

高置信

User communicates in Chinese.

高置信

User prefers the temperature conversion skill to support both Celsius-to-Fahrenheit and Fahrenheit-to-Celsius.

高置信

User uses a weekly report template for progress reports.

低置信

User asks for agile operations reports covering iteration health, requirement distribution, defect hotspots, and test quality.

高置信

User requests a delivery monitoring dashboard showing iteration progress, defect distribution, test quality, and DORA metrics.

低置信

User asks for a printable quality report covering defect trends, test coverage, and release readiness.

高置信

User asks for a cross-project analysis for reporting in a PPT.

低置信

User asks for PPT slides summarizing current project requirements and iterations with charts and defect analysis.

高置信

User uses an idea-to-delivery-planning workflow to plan new requirements, iteration tasks, and test cases.

低置信

User prefers a one-way Celsius-to-Fahrenheit conversion.

低置信

导出

把画像导出为文件。导出前可逐条审阅，取消勾选的不进文件。

导出

导入

载入此前导出的画像。写入之前会先展示差异。

导入

环境：浏览器 OPFS

- Analyze user behavior from daily conversations
- Distill and summarize the user profile and preferences
- Inject the user profile into the LLM's context
- Stored only in the user's local browser
- Supports deletion, import & export



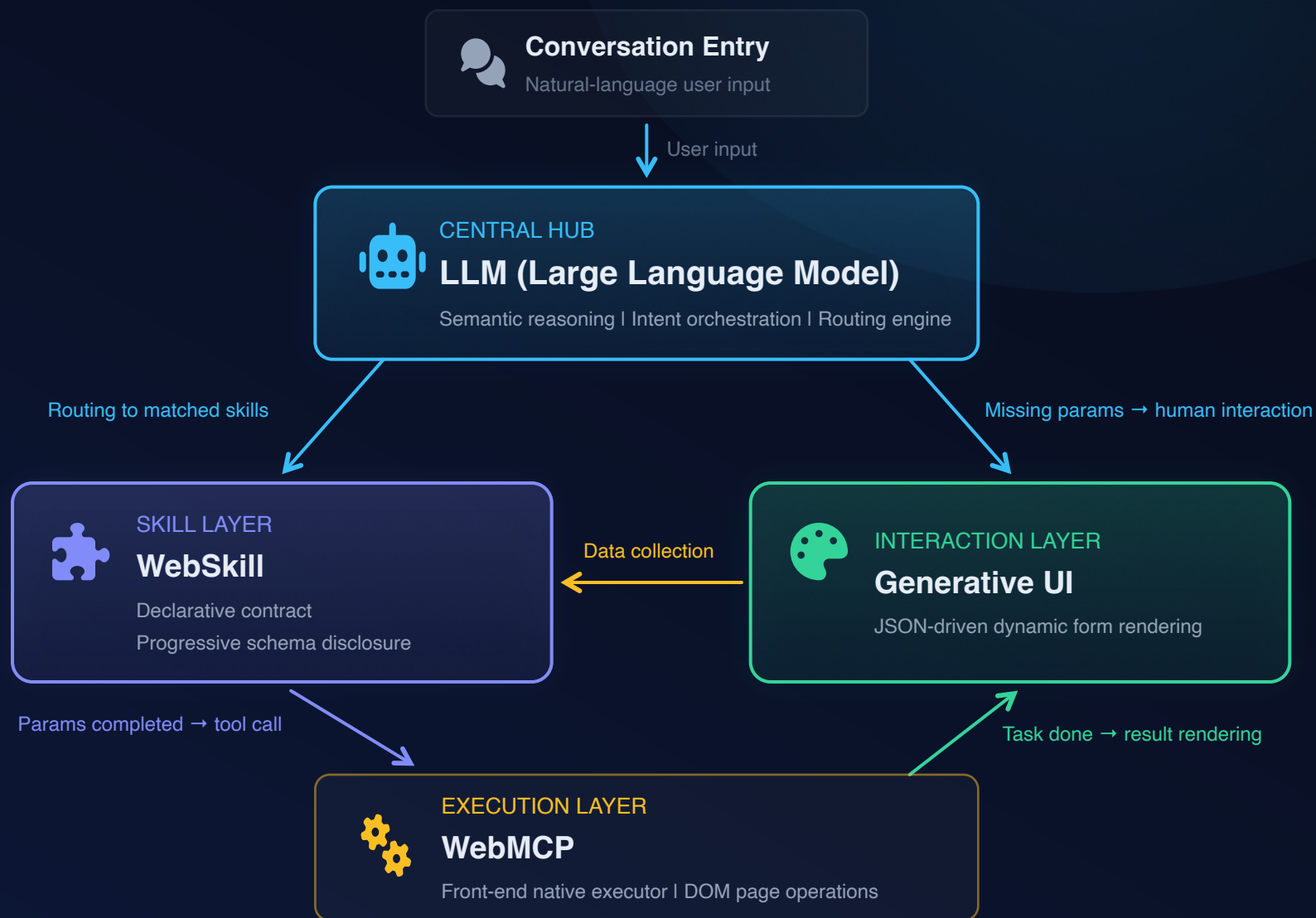
WebSkill solves "when to call tools" for the LLM, closing the front-end Agent task loop

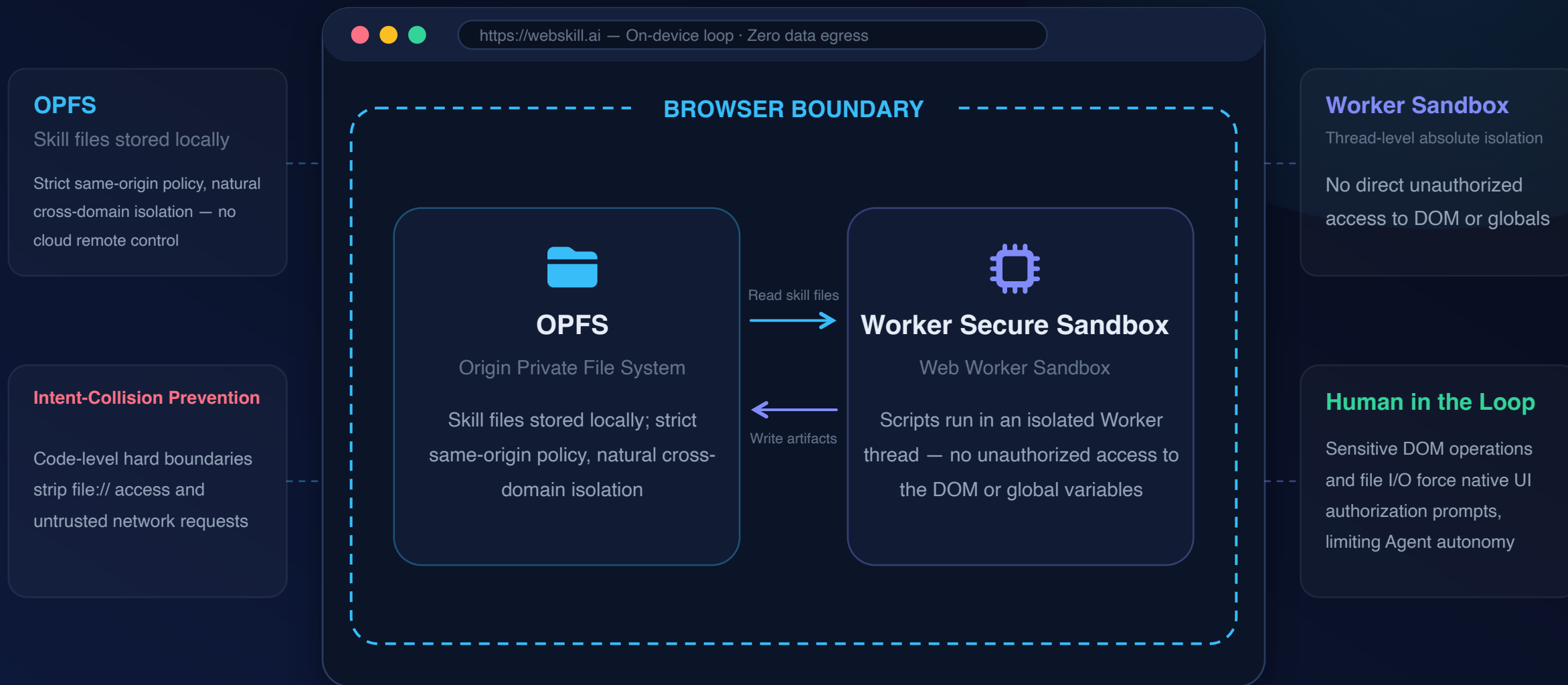


WebSkill solves context explosion when many MCP tools are loaded

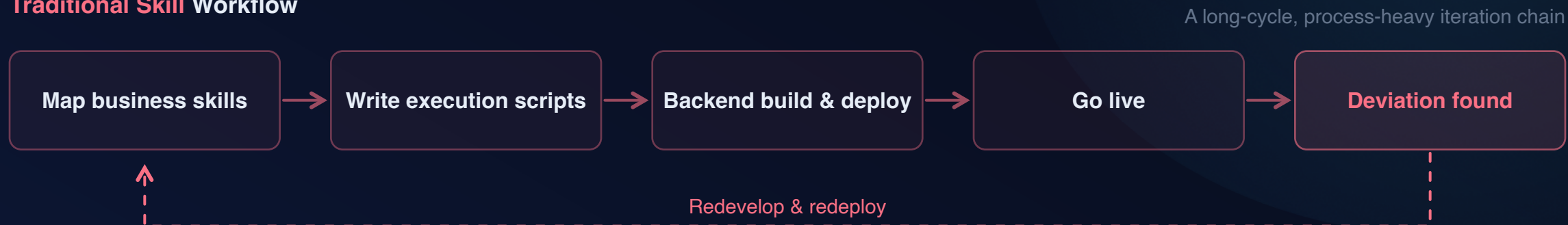


Generative UI solves **WebSkill**'s human-interaction problem on missing parameters, and renders task results

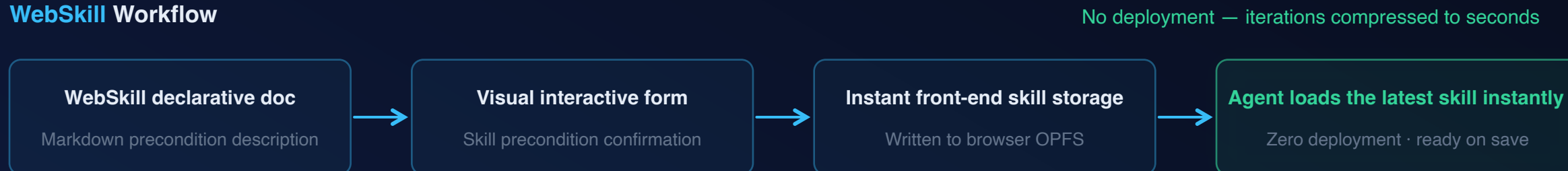




Traditional Skill Workflow

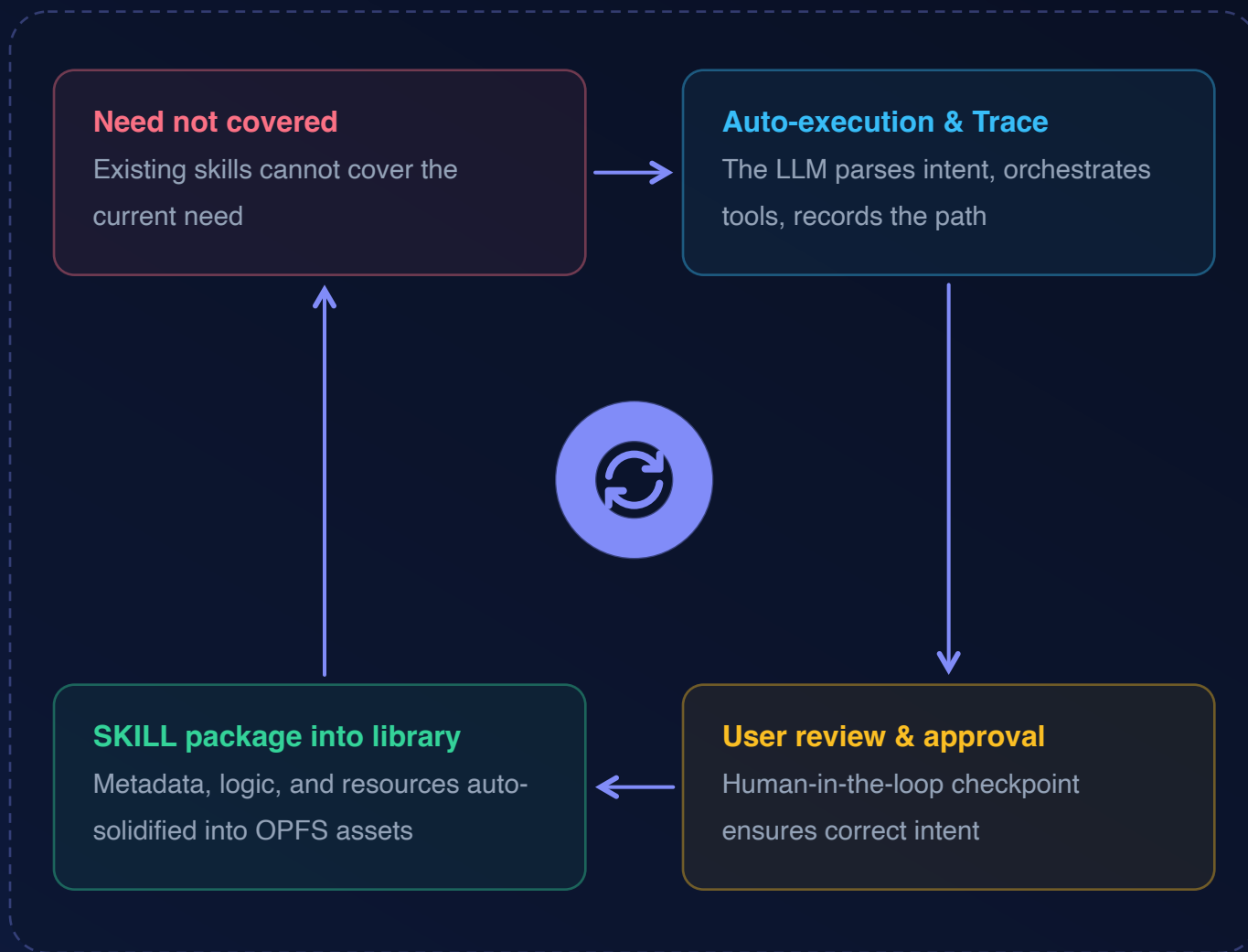


WebSkill Workflow



Iteration-chain comparison Traditional backend Skills loop through "develop → deploy → go live → rework on deviation"; WebSkill ships declarative documents with the web app — stored and used instantly on the front end, compressing Agent iteration cycles to seconds.

Private experience evolution: analyze, execute, distill skill assets



Key Stages of Auto Skill Generation



Need analysis

The LLM analyzes the need and determines existing skills fall short



Execution records

Full trace of execution: inputs, outputs, and intermediate decisions at every step



User review

After successful execution, the result becomes a skill file only with user consent (human-in-the-loop)



Skill file generation

Auto-generates SKILL.md (metadata + instructions), scripts/ (execution logic), references/ (resources)



Storage & reuse

Stored in the OPFS skill directory — directly callable for similar needs next time

Skill Lifecycle Management



Version update (REFRESHED): supports skill update, rollback, and uninstall — key operations require user confirmation



Memory & Versions

Short-term memory In-session execution context and intermediate results (in Worker memory)

Long-term memory Cross-session skill execution experience, user preferences, parameter habits

Versioning Skill update, rollback, uninstall, and more



Trace Observability

Tracking Trace every step of skill execution via the trace method

Dimensions Execution time, input params, output results, errors

Use cases Debugging, performance analysis, experience review



Audit & Evaluation System

Scoring After execution, results are scored against predefined criteria

High score Stay active; updates and upgrades allowed

Low score Triggers rollback or uninstall to stop low-quality skills

Human factor Key operations (installing external skills, deleting skills) require user confirmation

 Skill Dynamic Update Flow Skill Import Methods

Import Method	Description	Permission Required
HTTP Install	Fetch and install a skill package from a URL	User confirmation + format check
File Upload	Upload a local skill package file	User confirmation + format check
Page Sub-Route Sharing	Share skills via web app sub-routes	Permission management required
File Download Sharing	Distribute by downloading skill package files	Permission management required

 Skill Distribution Strategies**Initial skills**

Packed into build artifacts of the web app and written to OPFS on first page open

Dynamically updated skills

Via the MCP protocol, the app backend dynamically sets the page's MCP prompt / resource / tools — updated silently or with authorization

Permission defense

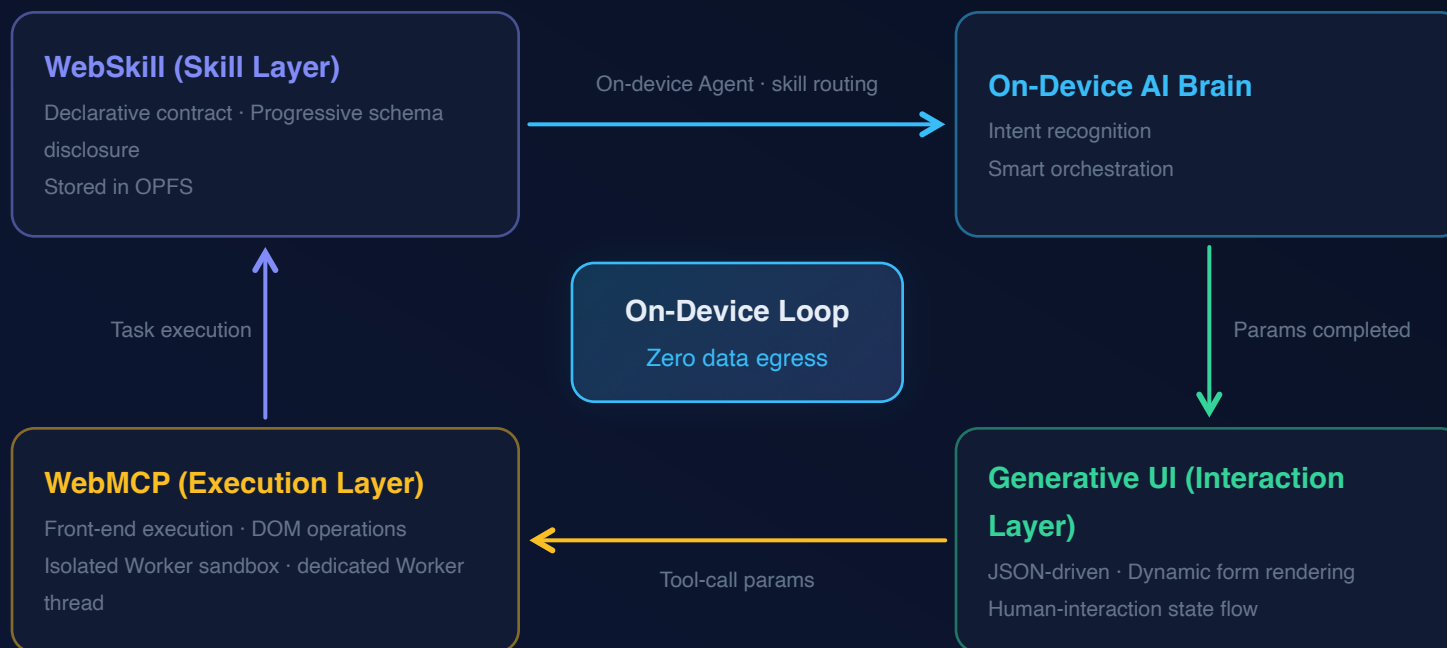
Skill sharing enforces a complete permission-verification mechanism, preventing unauthorized access

Web Industry Value

- Accelerating personal AI adoption on the Web
- Page-level progressive intelligence

**Advancing W3C Standards**

- Personalized skills
- Privacy & security
- Ecosystem win-win
- Browser / HarmonyOS

**WebSkill On-Device Closed-Loop Architecture (Agentic Web)****Personalization:
Unique for every user**

- Skill docs stored in OPFS
- Optimized with user data
- Learn from users and evolve.

WebSkill: extend skill ecosystem

- HTTP import skills
- MCP standard (page skills)
- Industry compatible skill ecosystem

THANKS

Thanks for listening · WebSkill lets skills grow in the browser

 <https://webskill.ai>

webskill.ai

HOME

IMPLEMENT

STANDARDIZATION

DOCS

中文

English

WebSkill

Web-Based Agent Skill

Running in the Browser

WebSkill turns the browser into a smart Agent execution environment. It completely solves the LLM context bloating issue, ensures absolute privacy through OPFS and Worker sandboxes, and transforms Agent capabilities into lightweight, dynamically distributed SaaS.

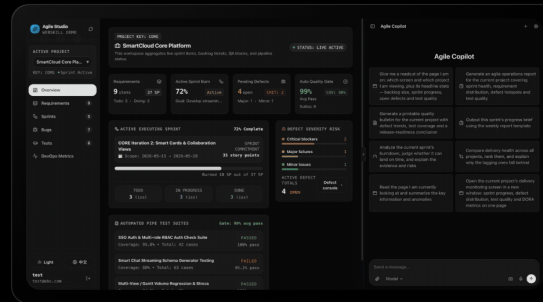
Draft Proposal

Frontend-Native

Open the WebSkill Live Demo →

What is Agent Skill?

An Agent Skill is a modular, reusable, standardized capability unit. It provides domain-specific



W3C WEB & AI IG

Presented at W3C



Next Stop — the W3C Community Group



Join the discussion