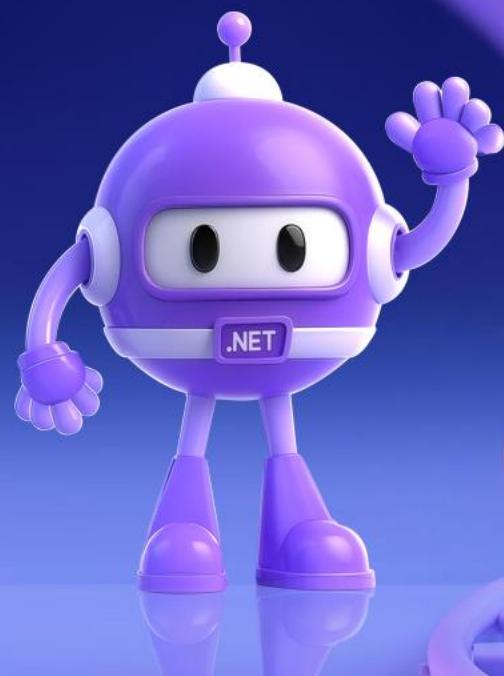


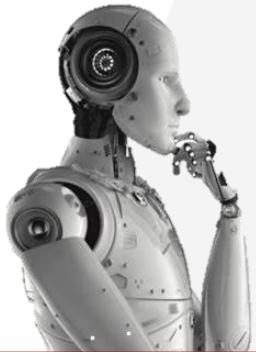
# .NET Conf China 2025

改变世界 改变自己

2025年11月30日 | 中国上海



►► .NET Conf China 2025 . 1130



从0到1用.NET  
打造 Playwright MCP

开源栈

潘淳





- ▶▶ **标准起草：**  
《苏州市银行业RPA项目实施规范》
- ▶▶ **程序开发：**  
《AntMCP.NET》：基于.NET的MCP教学工具  
《CodeEasy》：自然语言编程工具  
《ThanosOffice》：自然语言自进化智能Office插件  
《RobinRPA》：基于Ribion的RPA自动化软件

- ▶▶ **AIGC专著：**  
《一本书讲透 MCP：AI Agent 互联网新纪元》 [AntMCP.NET](#)  
《DeepSeek从企业智能化到办公自动化》 [CodeEasy](#)

## 数字化转型创新专家与 金融科技AI应用实践者

潘淳

- ▶▶ 硅基创新社区（硅创社）创始人
- ▶▶ 微软技术俱乐部（苏州）执行主席
- ▶▶ 中国DeepSeek精英讲师联合会秘书长
- ▶▶ 邮储银行总行级"潘淳创新实验室"负责人



RPA+AI 创新实验室(总行级)

# 目录



- 01 MCP 服务基座设计
- 02 MCP 服务开发实践
- 03 Playwright MCP



# 01

## PART ONE

# 上

智能体引擎的核心构建  
MCP基座设计

## ①操作云端

流式HTTP

Streamable HTTP

SSE

Server-Sent Events

http://82.156.17.205/cnstock/sse

https://mcp.amap.com/sse?key=876d9d561f2f948

https://docs.mcp.cloudflare.com/sse

删除 新增 编辑

调用工具 maps\_weather (10:29:51.525)

调用工具 maps\_weather (10:29:51.525)

调用工具 maps\_geo (10:29:51.763)

调用工具 maps\_geo (10:29:51.980)

调用工具 maps\_direction\_bicycling (10:30:02.978)

```
,"daytemp":34,"nighttemp":27,"daywind":东
,"nightwind":东,"daypower":1-3,"nightpower":1-3,"daytemp_float":34.0,"nighttemp_float":27.0},{ "date":"2025-07-25","week":5,"dayweather":多云,"nightweather":多云,"daytemp":35,"nighttemp":28,"daywind":东,"nightwind":东,"daypower":1-3,"nightpower":1-3,"daytemp_float":35.0,"nighttemp_float":28.0}]}]
```

7. 东环路高架出口
  - 沿东环路高架出口骑行26米。
8. 东环路
  - 沿东环路骑行889米, 左转。
9. 现代大道
  - 沿现代大道骑行30米, 向右前方行驶。
10. 现代大道
  - 沿现代大道向东骑行2616米, 靠右。
11. 现代大道
  - 继续沿现代大道向东骑行5485米, 右转。
12. 南施街
  - 沿南施街向南骑行556米, 左转。
13. 旺墩路
  - 沿旺墩路骑行515米, 向左后方行驶。
14. 终点: 星墩巷1号邮储银行大楼
  - 沿旺墩路骑行144米到达目的地。

总距离: 约12.4公里

预计时间: 约49分钟

苏州明天的天气如何? 再帮我规划一条从姑苏区平江路到工业园区星墩巷1号邮储银行大楼的骑行线路。

苏州明天的天气如何? 再帮我规划一条从姑苏区平江路到工业园区星墩巷1号邮储银行大楼的骑行线路。

MCP示例

- 第一步：查找电脑上"MyWorkSpace"目录，如果没有，则在D盘上创建 Standard Input/Output
- 第二步：打开这个目录
- 第三步：在该目录中，创建一个"八月报表"
- 第四步：打开这个目录
- 第四步：调研"MCP在金融领域的应用情况"，写一篇深度论文，并以论文风格起一个合适的标题
- 第五步：在"八月报表"目录中，以该标题为文件名，创建一个WORD文档，并写入论文内容
- 第六步：用WPS打开该文档
- 第七步：在WPS当中，移到文档末尾并加入一行内容："祝苏州银行业大讲堂之金融场景智能生态新范式专题培训会成功举办"

厂商: DeepSeek 模型: deepseek-chat 温度: 0.70 上限: 2048 申请Key e7f70d424e75b

你 08/22 08:26:15  
列出可用工具

A 助手 08/22 08:26:15  
我可以为您列出所有可用的工具及其功能:

## 系统信息工具

- get\_system\_summary - 获取系统摘要 (操作系统、CPU、内存、运行时间)
- get\_os\_info - 获取详细的系统信息
- get\_cpu\_info - 获取CPU信息
- get\_memory\_info - 获取内存信息
- get\_disk\_info - 获取磁盘信息
- `get [已停止]

—— 统计: 耗时=7.248 s

输入与操作 附件

列出可用工具

MCP 使用

**MCP示例**

准备：smart-campus-admin.html（网页在D盘）

- 第一步：用浏览器打开这个页面：

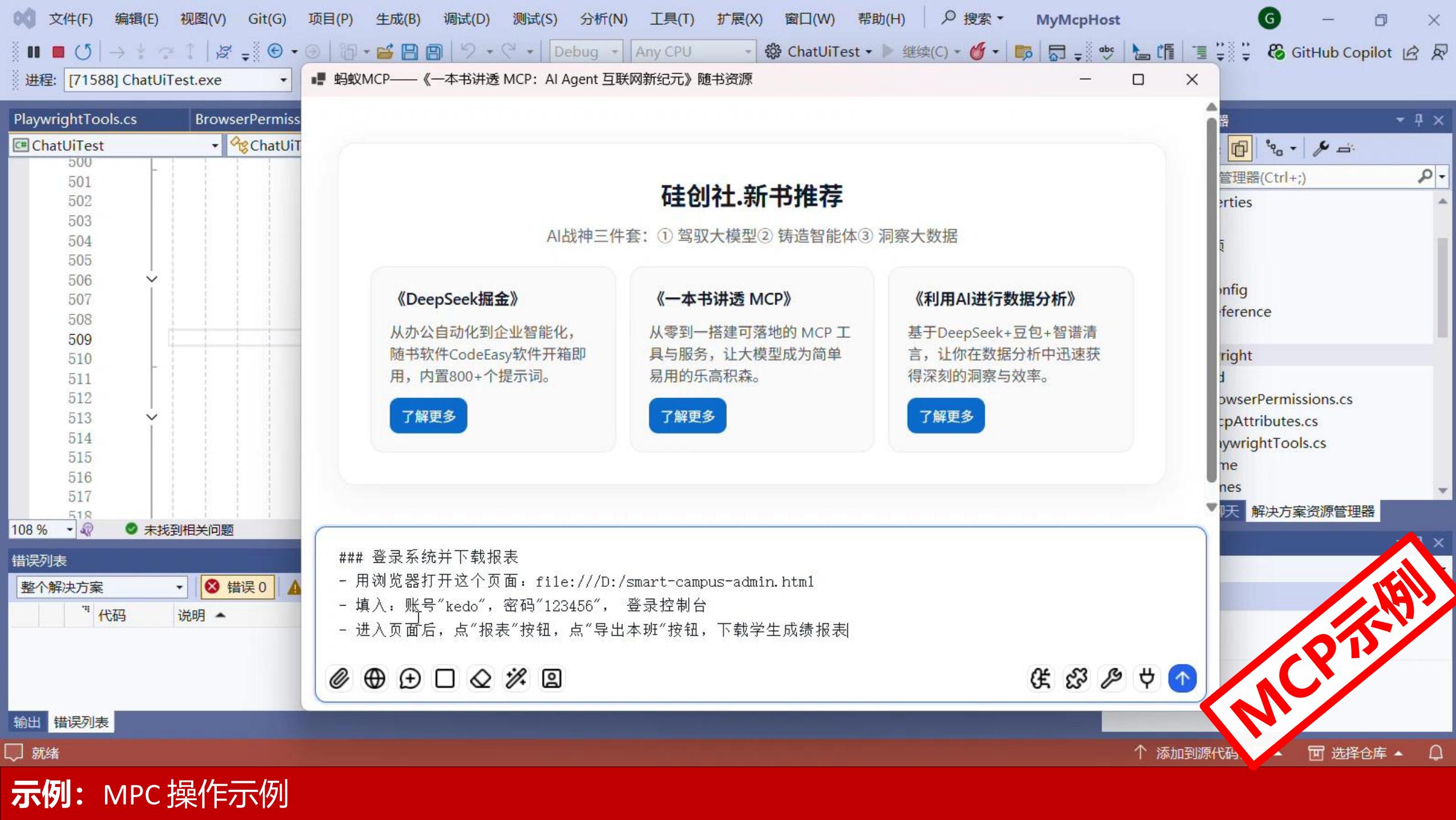
file:///C:/Users/Administrator/Desktop/AntMcp.NET/smart-campus-admin.html

- 第二步：填入信息：账号"kedo"，密码"123456"，

- 第三步：登录控制台进入页面

- 第四步：点击"报表"按钮，

- 第五步：点击"导出本班"按钮，下载学生成绩报表，显示下载路径





# 02

## PART TWO

# 02

智能体服务的构造方案  
**MCP服务开发**

写MCP服务检测IP

①简单开发

https://chatgpt.com/c/691c38c9-b9f8-8329-a8f8-bd0c415f8240

ChatGPT 5.1

新聊天 搜索聊天 库 Codex GPT 项目 新项目 ExternalBrowserWinForms snapshotForAI PlaywrightMCP Form.ts Playwright-MCP-TS 展开显示 Michael Morris Plus

FileTools.cs 文件

### 参考以上代码，写一个MCP服务，需求如下  
- 检查电脑IP地址

**(开发工具)  
AntMCP.NET**

```
csharp
using ModelContextProtocol.Server;
using System.ComponentModel;
using System.Net;
using System.Net.NetworkInformation;
using System.Net.Sockets;
using System.Text.Json;
```

询问任何问题

ChatGPT 也可能犯错。请核对重要信息。

共享

写MCP服务检测IP

ChatGPT 腾讯元宝 - 轻松工... DeepSeek | 深度搜索 QQ邮箱 运营管理部周工作... 推特 PromptUp - Create,... MCP Client Timeout... 内部局部函数

## MCP 服务编译

关闭

```
/// <summary>
/// 获取本机的 IP 地址列表（按网卡维度），默认只返回非回环的 IPv4 + IPv6。
/// </summary>
[McpServerTool, Description("Get local machine IP addresses (IPv4/IPv6) grouped by network interface.")]
public static string GetLocalIpAddresses(
    [Description("Include IPv6 addresses (default: true)")] bool includeIPv6 = true,
```

```
[info] C:\Users\Administrator\AppData\Local\ChatUiTest\Builds\IpAddress_20251118_091819618\IpAddress.cs(81,40): warning
CA1416: 可在所有平台上访问此调用站点。"UnicastIPAddressInformation.SuffixOrigin" 在 'windows' 上受支持。
(https://learn.microsoft.com/dotnet/fundamentals/code-analysis/quality-rules/ca1416)
[C:\Users\Administrator\AppData\Local\ChatUiTest\Builds\IpAddress_20251118_091819618\IpAddress.csproj]
[info] 3 个警告
[info] 0 个错误
[info]
[info] 已用时间 00:00:19.38
[info]
[info] 有可用的工作负载更新。有关详细信息，请运行 `dotnet workload list`。
编译完成
```

✓ 编译成功: C:\Users\Administrator\AppData\Local\ChatUiTest\Builds\IpAddress\_20251118\_091819618\bin\Release\net8.0\IpAddress.dll

编译为 DLL



②复杂开发

github.com/HOffice/playwright-dotnet-mcp

询问 Google

HOffice / playwright-dotnet-mcp

Type / to search

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

playwright-dotnet-mcp Public

Unpin Watch 0 Fork 3 Starred 26

main 148 Branches

HSOffice Merge pull request #142 from HOffice/codex... Add documentation for test suite last month

DetectBrowserProcess docs: translate Playwright MCP server readme to Chinese last month

PlaywrightMcpServer.Tests Allow attaching to existing browser processes last month

PlaywrightMcpServer Add top address bar to WebView2 browser last month

PlaywrightRemoteBrowserLauncher Move .NET server projects to repo root last month

WebView2BrowserHost Revert "Remove redundant LICENSE file" last month

LICENSE

Activity 26 stars 0 watching 3 forks

Releases No releases published Create a new release

【开发工具——组合】  
Codex + Github + Visual Studio

优化版本推荐 x Codex x HSOOffice/playwright-dotnet- x +

chatgpt.com/codex?tab=all

设置 文档 MM PLUS

## 接下来我们需要编写什么代码?

### 请基于现有 TypeScript (ts) 版工具实现,  
- 对照其功能与接口设计, 补全 .NET 版工具 (tools) 中的所有 TODO 部分,  
- 确保功能一致、结构对齐, 并保持跨语言的行为等价性。|

+ □ HSOOffice/playwright-... main 1x ↕ 0 ↑ ↖

任务 代码审查 归档

Add URL auto-correction and Bing search  
刚才 · HSOOffice/playwright-dotnet-mcp 已合并 +74 -4

Complete .NET version of todo  
3 分钟之前 · HSOOffice/McpOfPlaywrite 已合并 +1104 -147

更新 README.md 仓库结构部分  
4 分钟之前 · HSOOffice/playwright-dotnet-mcp 已合并 +4 -0

改进 Process 启动逻辑和提示信息  
11月10日 · HSOOffice/playwright-dotnet-mcp 已合并 +139 -46

第一步：在云端 Codex 中提交需求

优化版本推荐 Complete .NET version of todo +

chatgpt.com/codex/tasks/task\_e\_68ec3c4630c883299e712ee7242b85a0

Complete .NET version of todo  
10月13日 · HSOOffice/McpOfPlaywrite · codex/complete-.net-version-of-todo · +1104 -147

归档  查看 PR

完成.NET 的toos 的todo 部分。  
要对标ts 版本，完成相应.NET 版的

已工作 15 分 >

**Summary**

- Implemented keyboard typing and key-press actions along with mouse movement, clicks, and drags to mirror the TypeScript tool behavior, including response logging and wait handling. 0 0
- Added support for JavaScript evaluation, file uploads, and wait-for conditions, ensuring snapshot integration and argument validation consistent with the TS implementation. 0 0 0
- Completed verification, dialog handling, tab management, and tracing/install workflows so .NET tools now provide parity with their TS counterparts. 0 0 0 0 0
- Implemented screenshot/PDF exports and strengthened shared infrastructure for browser lifecycle, modal state management, and output path resolution. 0 0 0 0

**Testing**

- ⚠ dotnet test PlaywrightMcpServer.Tests/PlaywrightMcpServer.Tests.csproj \*(dotnet CLI not available in container)\*

+ 请求更改或提出问题

差异 日志

dotnet/PlaywrightTools.Actions.Common.cs +50 -14 :

```
1   1     using System;
2   2     + using System.Collections.Generic;
3   3     using System.ComponentModel;
4   4     using System.Threading;
5   5     using System.Threading.Tasks;
6   6     using ModelContextProtocol.Server;
7   7     + using Microsoft.Playwright;
8   8
9   9     namespace PlaywrightMcpServer;
10 10
11 11     public sealed partial class PlaywrightTools
12 12     {
13 13         [McpServerTool(Name = "browser_close")]
14 14         [Description("Close the page.")]
15 15         public static async Task<string> BrowserCloseAsync(
16 16             CancellationToken cancellationToken = default)
17 17         {
18 18             // TODO: Implement tool logic for closing the current browser page.
19 19             // Pseudocode:
20 20             // 1. Retrieve the active page instance.
21 21             // 2. Invoke the page close operation.
22 22             // 3. Return a serialized result confirming the page has closed.
23 23             await Task.CompletedTask;
24 24             throw new NotImplementedException();
25 25
26 26             var args = new Dictionary<string, object?>(StringComparer.OrdinalIgnoreCase);
27 27 }
```

第二步：Codex 生成代码，点击查看 PR

优化版本推荐 Complete .NET version of tool Implement .NET Playwright tool +

github.com/H5Office/playwright-dotnet-mcp/pull/70 询问 Google

Merged Implement .NET Playwright tool actions #70  
H5Office merged 1 commit into main from codex/complete-.ne... on Oct 13

dotnet/PlaywrightTools.Actions.Verify.cs

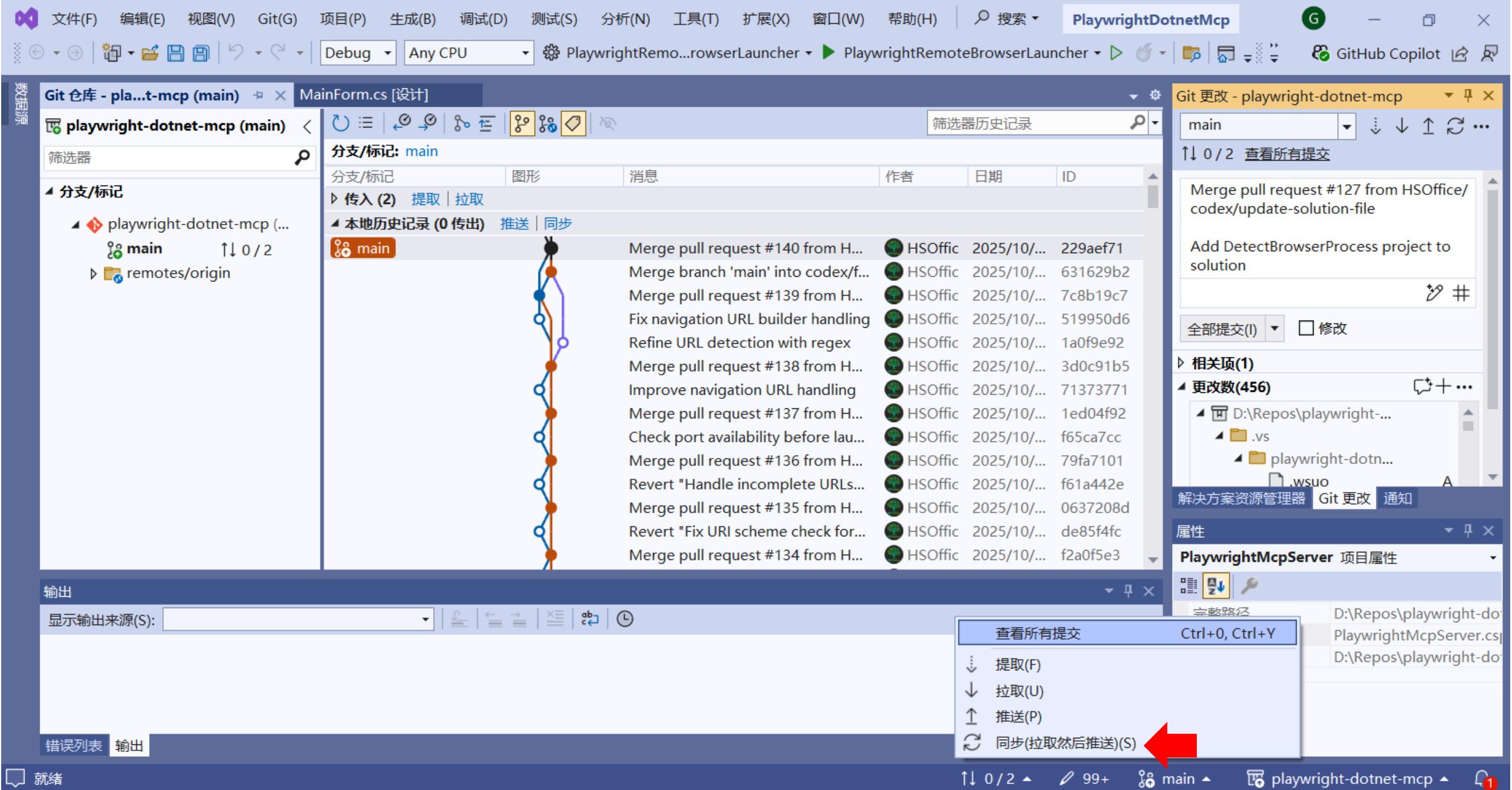
Comment on lines +94 to +108

```
94 +         for (var i = 0; i < count; i++)  
95 +     {  
96 +         var candidate = locator.Nth(i);  
97 +         if (await candidate.IsVisibleAsync().ConfigureAwait(false))  
98 +     {  
99 +         visible = true;  
100 +        break;  
101 +    }  
102 + }  
103 +  
104 + if (!visible)  
105 + {  
106 +     response.AddError("Text not found");  
107 +     return;  
108 + }
```

sourcery-ai bot on Oct 13

建议：不可见文本的错误消息可能具有误导性。

考虑对“文本不存在”和“文本存在但不可见”使用不同的错误消息，以提高清晰度。



第四步：在 Visual Studio 中同步代码，并测试（可选）



复杂智能体的设计理念  
**Playwright MCP**

PART THREE

[README](#) [MIT license](#)

# Playwright .NET MCP

浏览器自动化智能体服务器 (.NET 版) · A Model Context Protocol server toolkit powered by Microsoft Playwright for .NET

Playwright .NET MCP 为需要将浏览器能力暴露给大语言模型 (LLM) 或 Agent 宿主的应用，提供一套 .NET 实现的工具集合。项目遵循 [Model Context Protocol](#) 规范，通过 `ModelContextProtocol.Server` 包的 `McpServerTool` 特性描述工具参数与返回值，让宿主可以在不依赖视觉模型的情况下，结构化地驱动网页。

- **结构化快照：**利用可访问性树生成与 TypeScript 版兼容的 YAML/Markdown 快照格式。
- **确定性操作：**通过 Playwright 定位器完成导航、表单、鼠标、键盘等操作，避免坐标点击的不确定性。
- **Agent 友好输出：**统一的 `Response` 序列化逻辑输出结果、代码片段、标签摘要与快照文本，方便客户端解析。

## 仓库结构 | Repository Layout

路径	说明
<code>PlaywrightMcpServer/</code>	Playwright MCP 核心实现： <code>PlaywrightTools</code> 部分类、响应序列化、Tab 管理与快照构建。

如何开始

开源项目

# 代码框架

优化版本推荐 Complete .NET version of tool

github.com/HOffice/playwright-dotnet-mcp/blob/main/docs/ARCHITECTURE\_PLAN.md

询问 Google

Files

main

Go to file

- > DetectBrowserProcess
- > PlaywrightMcpServer.Tests
- > PlaywrightMcpServer
- > PlaywrightRemoteBrowserLaunch...
- > WebView2BrowserHost
- v docs
- > ts
- | ARCHITECTURE\_PLAN.md
- | PlaywrightMcp\_Analysis\_Report...
- | snapshot\_capabilities.md
- | tab-capture-snapshot.md
- | tab-comparison.md
- | LICENSE
- | PlaywrightDotnetMcp.sln

Preview Code Blame

Raw

# .NET Playwright MCP 服务代码框架规划

本文档给出在 PlaywrightMcpServer 目录下实现 Playwright MCP 服务的推荐代码框架，用于指导后续开发。规划遵循 TypeScript 版本的模块划分，确保工具名称、能力与行为一一对应，同时聚焦于“仅提供 MCP 服务，供宿主复用”这一目标。

## 目录结构

```
PlaywrightMcpServer/
├── ARCHITECTURE_PLAN.md      # 本规划文
└── mcp/
    ├── PlaywrightMcpServer.csproj
    └── Core/
        ├── BrowserServerBackend/
        │   ├── BrowserServerBackend.cs
        │   └── ToolInvocationContext.cs
        ├── Context/
        │   ├── BrowserContextFactory.cs
        │   ├── Context.cs
        │   └── Tab.cs
        └── TabEvents.cs
```

① 主体： PlaywrightMcpServer

② 主体： PlaywrightMcpServer.Tests

③ 辅助： ...BrowserLauncher

④ 辅助： WebView2BrowserHost

⑤ 辅助： DetectBrowserProcess

**浏览器连接**

可执行文件: `net8.0-windows\WebView2BrowserHost.exe` 浏览... 端口: `9223`

起始 URL: `https://www.qq.com`

**分步操作**

1-启动进程 2-DevTools 3-Playwright 4-获取 Pages 5-访问 URL 清理 / 退出

**自动化与日志**

一键运行 重置全跑 开始日志 停止日志 保存快照  导航后自动截图

**网络设置**

代理 (--proxy-server):   
 忽略 TLS 证书错误 (IgnoreHTTPSErrors)

**实用工具**

导出 /json/list 导出 /json/protocol  
刷新页面列表 新开 Tab  
<https://example.com>

**页面与日志**

页面 (Pages):

Page-1 - 腾讯网  
Page-2 - Example Domain  
Page-3 - Example Domain

日志输出:

```
[11:05:57] [Console][Page-1][warning] Mixed Content: The page at 'https://www.qq.com/' was loaded over HTTP:  

[11:05:57] [Console][Page-1][warning] Mixed Content: The page at 'https://www.qq.com/' was loaded over HTTPS:  

[11:05:57] [Console][Page-1][warning] Mixed Content: The page at 'https://www.qq.com/' was loaded over HTTPS:  

[11:06:06] 截图超时: Timeout 60000ms exceeded.Call log: - taking page screenshot - waiting for fonts to load...
[11:06:08] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:11] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:14] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:17] 切换当前页面为: Page-2  

[11:06:18] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:20] [Console][Page-1][log] frame on registerListener  

[11:06:20] [Console][Page-1][debug] channel received on listener[invoke]  

[11:06:20] [Console][Page-1][debug] channel received on listener[sync_method]  

[11:06:20] [Console][Page-1][log] frame on sync_method {getGameSession: }  

[11:06:21] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:23] [Console][Page-1][log] frame on invoke {apiName: getGameSession, args: Object}  

[11:06:24] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:26] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:26] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:26] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:27] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:27] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:28] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:28] [Console][Page-1][error] Failed to load resource: net::ERR_CONNECTION_CLOSED  

[11:06:28] [Console][Page-1][error] Error: timeout at https://game.weixin.qq.com/cgi-bin/minigame/static/pc-g
[11:06:34] 切换当前页面为: Page-1  

[11:06:38] 截图超时: Timeout 60000ms exceeded.Call log: - taking page screenshot - waiting for fonts to load...
```

**ARIA 机制优化**

**浏览器 HOOK**

能力分类	文件名	Tool 名称	功能简介
core 核心操作	common.ts	browser_close	关闭浏览器页面
		browser_resize	调整浏览器窗口大小
	navigate.ts	browser_navigate	跳转到指定 URL
		browser_navigate_back	返回上一页
	files.ts	browser_file_upload	上传文件
	dialogs.ts	browser_handle_dialog	处理对话框
	evaluate.ts	browser_evaluate	在页面或元素上执行 JS 代码
	wait.ts	browser_wait_for	等待文本出现/消失或等待时间
	form.ts	browser_fill_form	填写多个表单字段
	keyboard.ts	browser_press_key	模拟键盘按键
		browser_type	在元素中输入文本
	snapshot.ts	browser_click	点击元素（可双击/改键/带修饰键）
		browser_drag	在两个元素间拖拽
		browser_hover	悬停元素
		browser_select_option	在下拉框中选择选项
		browser_snapshot	页面无障碍快照
		browser_generate_locator	生成测试定位器
	network.ts	browser_network_requests	列出网络请求
	console.ts	browser_console_messages	获取控制台消息
	screenshot.ts	browser_take_screenshot	截图

Playwright

能力分类	文件名	Tool 名称	功能简介
vision 坐标操作	mouse.ts	browser_mouse_move_xy	移动鼠标到指定坐标
		browser_mouse_click_xy	在指定坐标点击
		browser_mouse_drag_xy	在坐标间拖拽
testing 测试断言	verify.ts	browser_verify_element_visible	验证元素（角色+名称）可见
		browser_verify_text_visible	验证文本可见
		browser_verify_list_visible	验证列表及其条目
		browser_verify_value	验证元素值/状态
core-tabs 标签页管理	tabs.ts	browser_tabs	管理标签页（列出/新建/关闭/切换）
pdf PDF导出	pdf.ts	browser_pdf_save	保存页面为 PDF
tracing 追踪管理	tracing.ts	browser_start_tracing	开始性能追踪
		browser_stop_tracing	停止性能追踪
core-install 浏览器安装	install.ts	browser_install	安装配置的浏览器

Playwright

**核心操作** **坐标操作** **测试断言** **标签页管理** **PDF导出** **追踪管理** **浏览器安装**

.NET Conf China 2025

改变世界 改变自己



# THANK YOU

MCP建立起了大模型和已有数字系统之间的高效连接，为未来大模型和物理世界的连接奠定了基础。占总的书是我看过的有关MCP最全面以及实战性最好的读本，推荐给大家！

邱达 钉钉副总裁

MCP让大模型成为简单易用的乐高积木，任何人都能利用它组建自己的AI应用。掌握MCP，是所有人在Agent时代的必备素养。这不是可选的技术革命，而是人类生产范式的根本转变。那些能熟练运用MCP的人将成为超级个体，错过这波浪潮很可能在竞争中掉队。让我们一起通过这本书读懂MCP、实操MCP，抓住升级为超级个体的机会！

林旅强 MCP中国开发者联盟发起人&  
开源社联合创始人

创业者切莫错过MCP带来的崭新机遇，它能助力你打破传统业务局限，开拓全新的智能化应用局面。单人AI超级应用公司的时代即将到来，它会催生出大量轻量级的创业团队，用极低成本就能打造出有竞争力的AI应用。投资人需敏锐察觉MCP的潜力，它就是开启下一轮财富增长的密钥。这本书将助你在这片新蓝海中找准方向，实现价值飞跃。

李明顺 行行AI董事长&顺福资本管理合伙人

这本书系统地介绍了MCP的技术原理、生态构建和实际应用，既有深度的技术解析，也有丰富的实战案例。作者在保持技术深度的同时，注重内容的易读性，让技术和非技术背景的读者都能从中受益。对于想要了解AI Agent互联发展趋势，或希望在实际项目中应用MCP技术的朋友来说，这是一本很有价值的参考书。

AJ 通往AGI之路社区发起人

本书不仅是一本技术指南，更是通向未来AI生态系统的权威路线图。几位作者是我在AI领域的挚友，也是国内AI应用领域实践与理论并重的资深专家，对MCP有着独到而深刻的前沿洞察。从AI Agent演进到MCP核心，从生态构建到实战开发，本书系统构建了完整知识框架。这本书不仅会帮你掌握MCP的技术精髓，还会帮助你在AI Agent互联网新纪元中找准定位，发现属于自己的机遇。

云中江树 LangGPT社区创始人/  
畅销书《智能体设计指南》作者

客服电话：(010) 88361066 88379833 68326294

MCP之于AI Agent，正如HTTP之于万维网、USB-C之于硬件生态，通过促进AI应用的标准化和模块化，降低AI开发门槛，繁荣AI生态。冰强长期耕耘在AI产业一线，我一直从他的分享中受益良多。本书从实践出发，图文并茂地探讨了MCP的核心功能、技术实现及其在生态系统中的应用，相信一定能满足相关读者对MCP知识的渴求。

高飞 至顶科技CEO兼总编辑

这不是又一本技术科普书，而是一份产业变革的情报。从MCP核心到生态构建，从实战案例到商业机会，作者以罕见的前瞻性，解码了AI应用标准化革命的全貌。当所有人还在讨论大模型时，真正的机会已转向Agent互联。错过这波浪潮，你将与下一个十年的AI红利失之交臂，这是一本能让你站在风口的书。

易亚婷 《易论AI》主理人

本书脉络清晰地梳理了AI Agent互联网的演进，剖析了MCP如何在标准化、安全与互操作性三大维度提供最优解，兼具技术深度与易读性，能为企业与Agent开发者打开智能体互联互通的新视野！

郭政纲 AIGC开放社区发起人

冰强作为业内颇具洞察力的布道者，在本书中抽丝剥茧地解析了MCP的技术架构，捕捉到了协议层创新带来的商业模式变革。书中展现的“积木式创新”思维带来了高扩展性的开发范式，或将催生智能体经济的爆发式增长。作为AIGCLink社区的发起人，冰强长期致力于构建技术普惠的开放生态。本书正是社区精神的延续：既可指引从业者穿越迷雾，又能凝聚行业共识。

董慧智 中国人工智能学会智工委常委&  
零氪云CEO



## 一本书讲透MCP

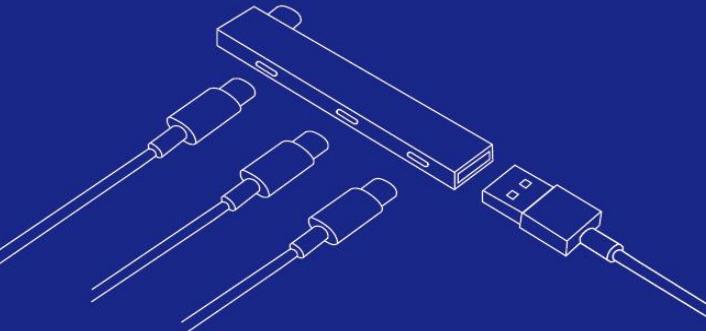
占冰强  
郭美青  
莫欣  
潘淳  
覃睿  
◎著

机械工业出版社  
CHINA MACHINE PRESS

# 一本书讲透 MCP

## AI Agent互联网新纪元

占冰强 郭美青 莫欣 潘淳 覃睿 ◎著



资深AI专家、MCP布道者撰写，钉钉副总裁、MCP中国开发者联盟发起人、  
通往AGI之路社区发起人等近10位专家推荐

从宏观到微观，从理论到实践，从技术到商业，从平台到生态，从现在到未来，围绕10余个主题对MCP做了全面、透彻、深入的讲解

机械工业出版社  
CHINA MACHINE PRESS

《智能体设计指南——成为提示词高手和AI Agent设计师》

《DeepSeek掘金——从企业智能化到办公自动化》

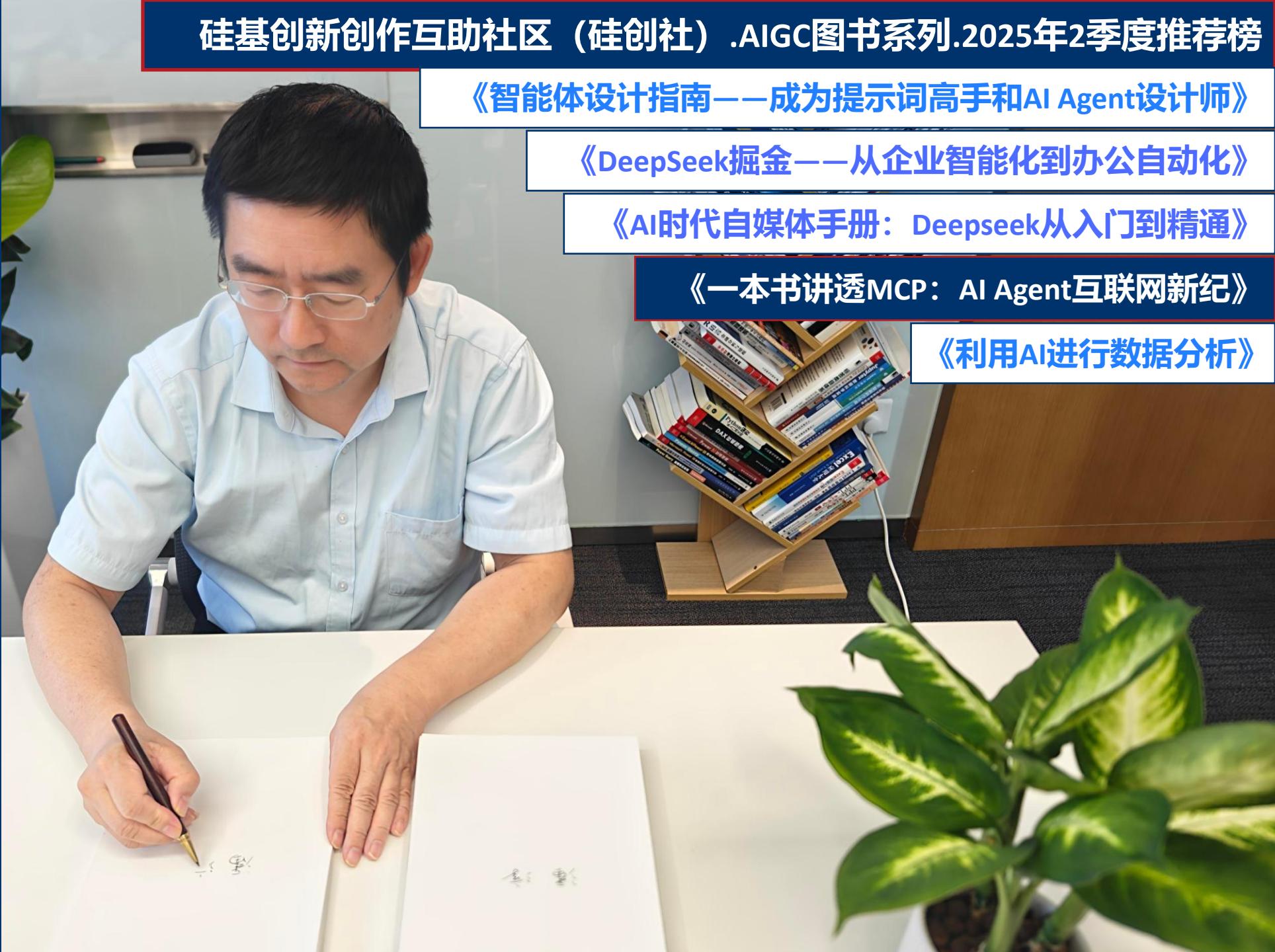
《AI时代自媒体手册：Deepseek从入门到精通》

《一本书讲透MCP：AI Agent互联网新纪》

《利用AI进行数据分析》



一本书讲透MCP



# .NET Conf China 2025

改变世界 改变自己



.NET Conf China 2025

改变世界 改变自己







