

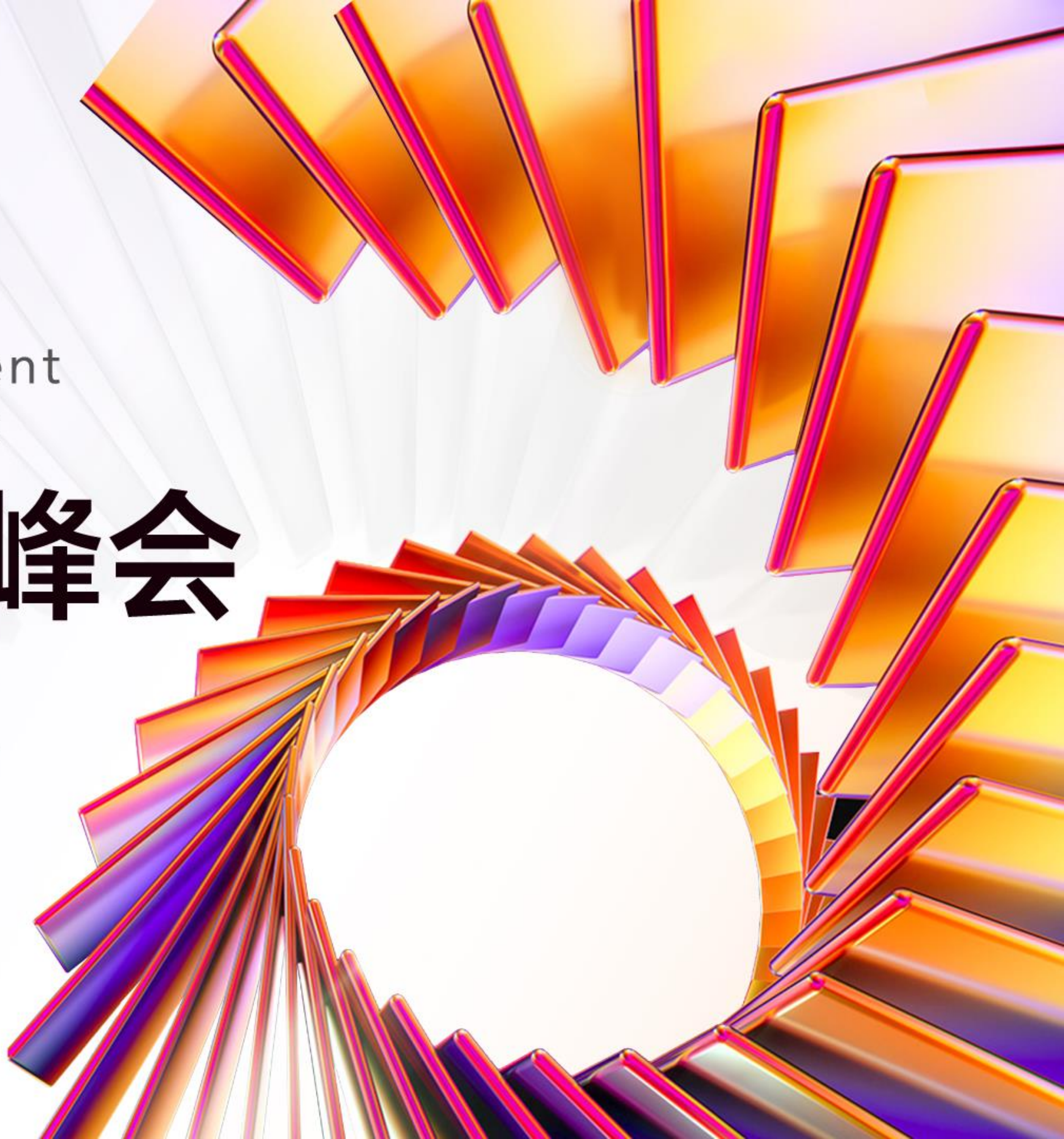


第8届 AI+ Development
Digital Summit

AI+研发数字峰会

拥抱AI 重塑研发

11月14-15日 | 深圳





2026 AI+PRODUCT INNOVATION SUMMIT

01.16-17 · ShangHai

AI+产品创新峰会



Track 1: AI 产品战略与创新设计

从0到1的AI原生产品构建

论坛1: AI时代的用户洞察与需求发现

论坛2: AI原生产品战略与商业模式重构

论坛3: Agentic AI产品创新与交互设计

2-hour Speech: 回归本质



用户洞察的第一性

——2小时思维与方法论工作坊

在数字爆炸、AI快速发展的时代，
仍然考验“看见”的“同理心”



Track 2: AI 产品开发与工程实践

从1到10的工程化落地实践

论坛1: 面向Agent智能体的产品开发

论坛2: 具身智能与AI硬件产品

论坛3: AI产品出海与本地化开发

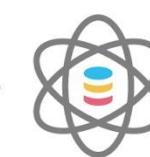
Panel 1: 出海前瞻



“出海避坑地图”圆桌对话

——不止于翻译:

AI时代的出海新范式



Track 3: AI 产品运营与智能演化

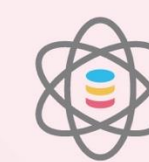
从10到100的AI产品运营

论坛1: AI赋能产品运营与增长黑客

论坛2: AI产品的数据飞轮与智能演化

论坛3: 行业爆款AI产品案例拆解

Panel 2: 失败复盘



为什么很多AI产品“叫好不叫座”?

——从伪需求到真价值:

AI产品商业化落地的关键挑战

智能重构产品 数据驱动增长

Reinventing Products with Intelligence, Driven by Data

80+

业界大咖

汇聚产品大咖的真知灼见

40+

创新案例

开拓全新AI+产品视角

10+

分论坛

涵盖产品到商业增长的全链路

800+

听众规模

共同探索产品的智能重构



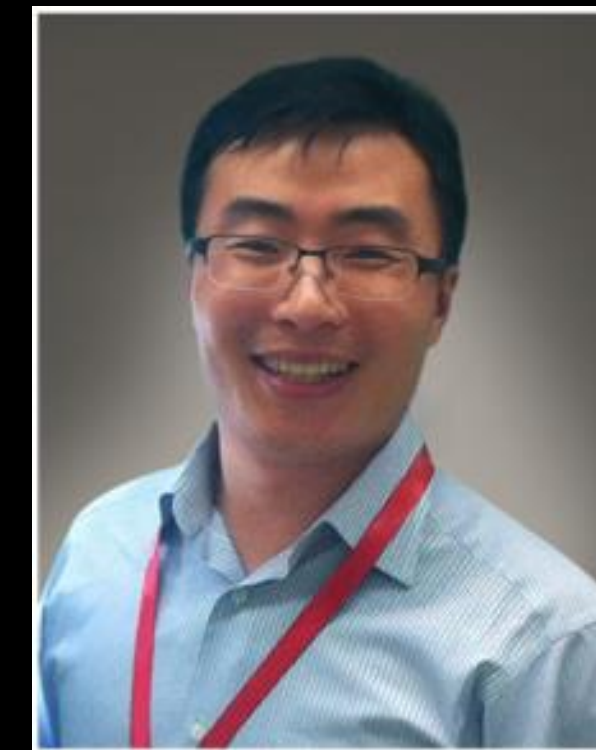
扫码查看会议详情



Qoder CLI 企业研发实践： 让团队开发更便捷、更安全、更高效

大家好，我是行云老马

- 25年软件产品研发和管理经验，20年云计算、AI领域专业经验
- 连续10任微软最有价值专家MVP称号，MVP区域总监
- 参与信通院云原生成熟度模型、开发上云等相关标准定义
- DevOps凤凰沙盘官方认证教练
- 曾在Rancher Labs, Citrix Systems, Oracle R&D等公司担任总架构师、研发经理、技术总监等职务

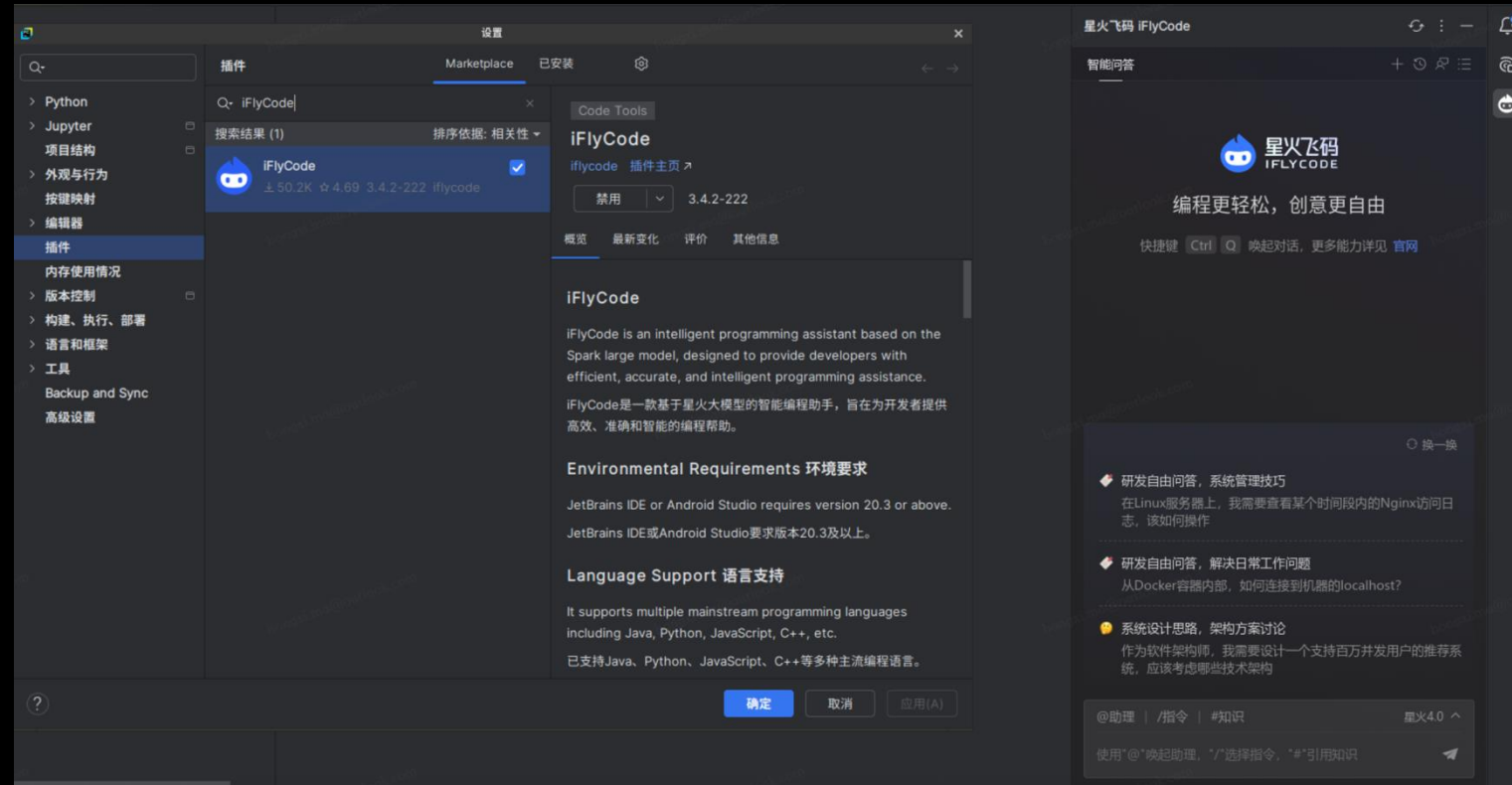


马洪喜

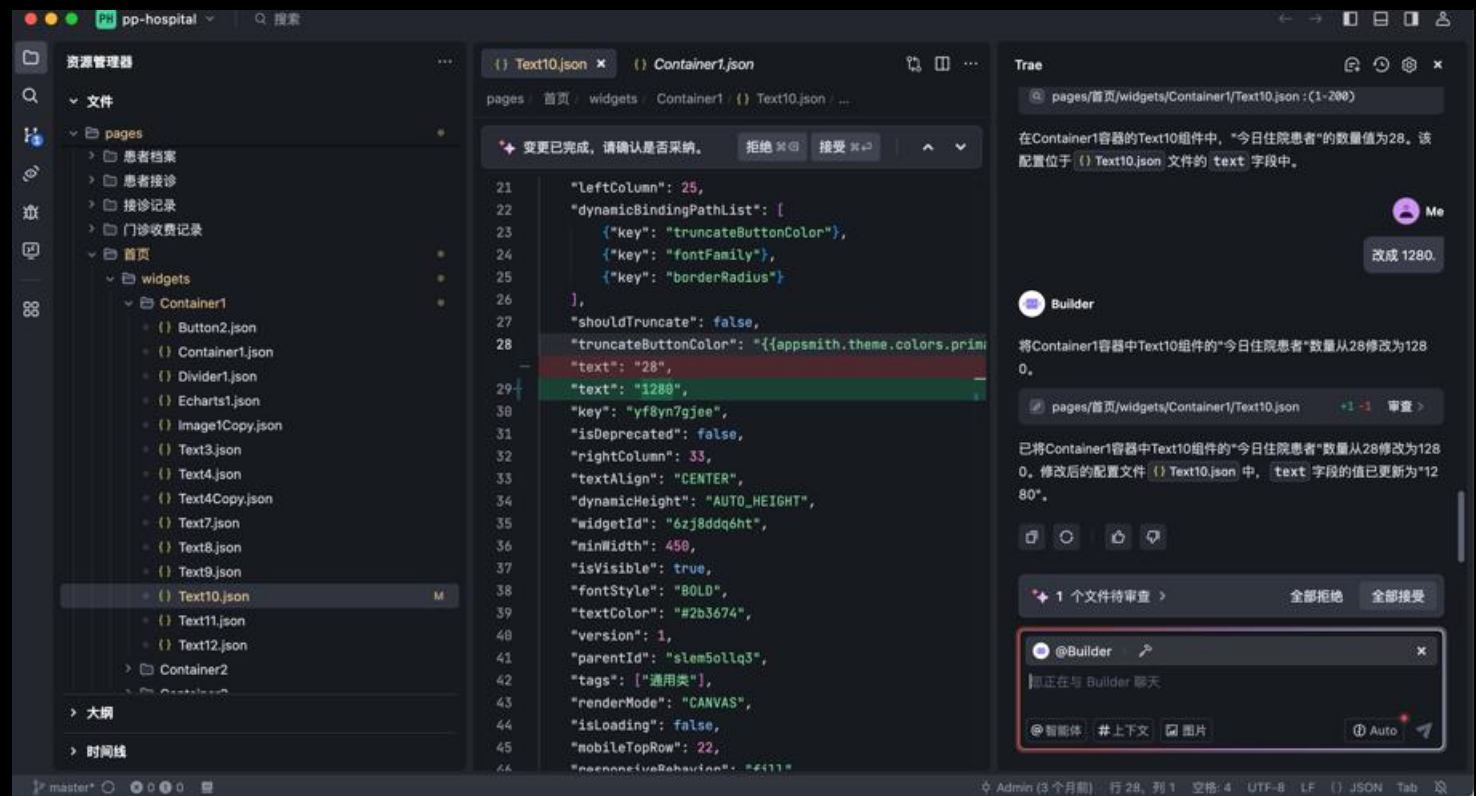
深圳行云创新CEO
云原生和AI原生架构师
微软最有价值专家MVP, MVPRD

Qoder CLI 还有各种花样玩法

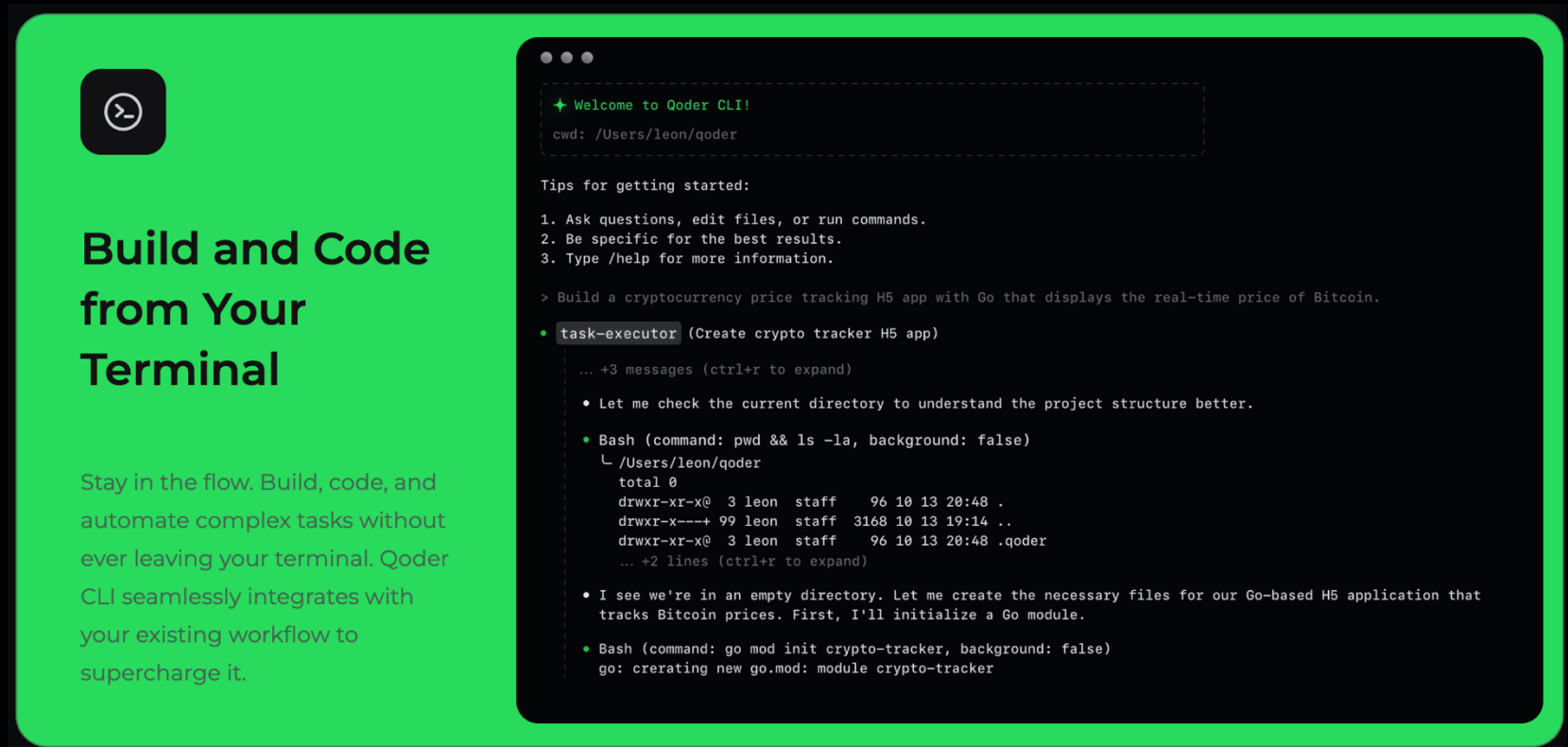
为什么我们喜爱 Qoder CLI



传统插件式



魔改 VS Code



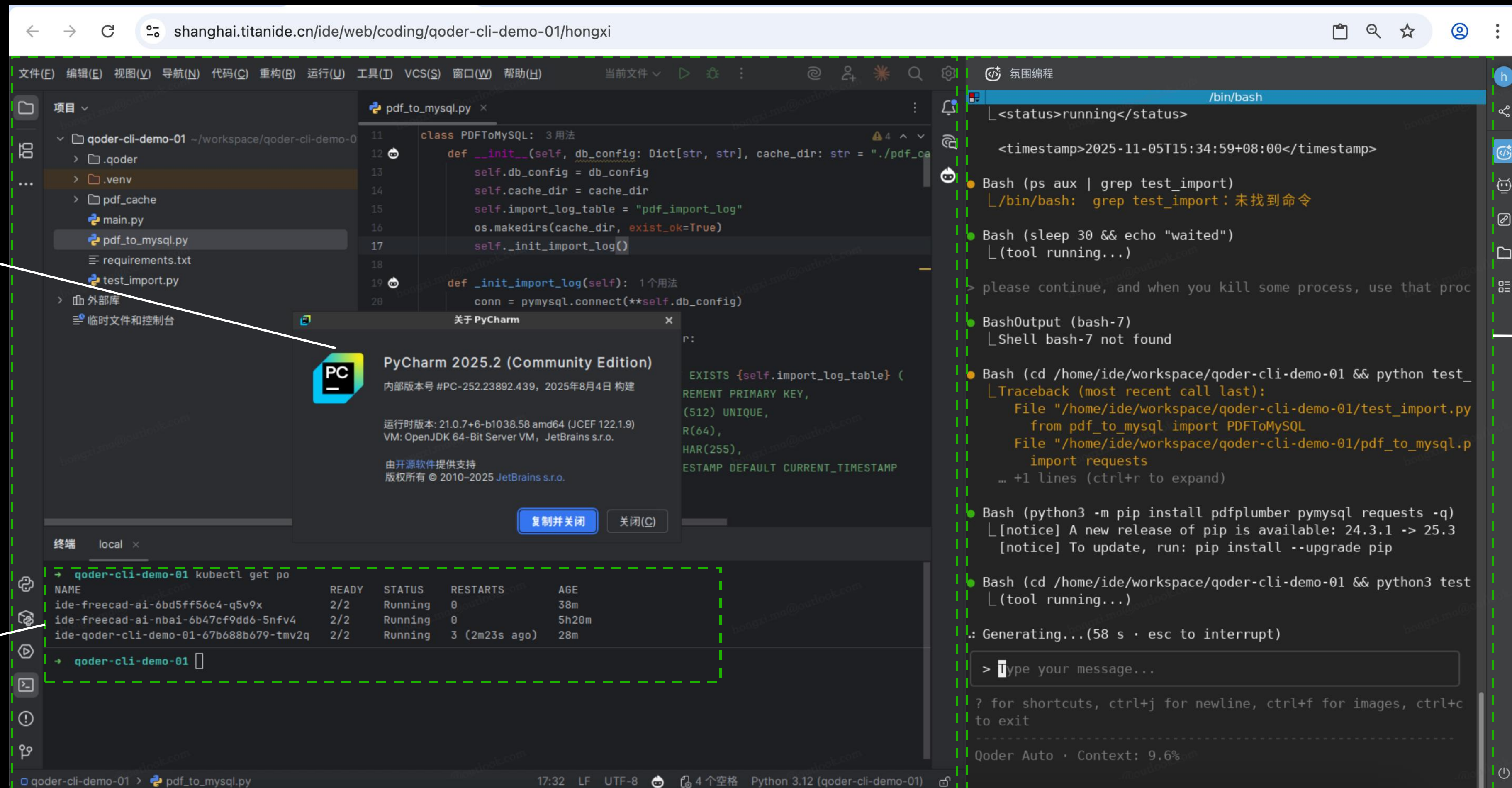
Qoder CLI

更重要的是Qoder CLI 可以在 K8s 和 Cloud IDE 里完美运行

喜爱的IDE

Qoder CLI

在K8s Pod里
安全运行



打开Chrome就能使用Qoder CLI和喜爱的IDE， 极度便捷

为什么我们在 K8s 和 Cloud IDE 里运行 Qoder CLI

采用本地IDE开发给我们带来的四大挑战

典型Agent任务耗时：
- 代码重构：2-6小时
- 全量测试：1-3小时
- 安全扫描：持续运行
- 多Agent协同：数小时到数天

开发者笔记本开机时间：
- 平均：8小时/天
- 休眠时间：16小时/天
- **任务被中断概率 >60%**

挑战1：时间维度冲突

如果允许插件随意执行命令：
→ 恶意插件可以窃取SSH密钥
→ 可以读取浏览器密码
→ 可以安装后门程序
→ 可以删除系统文件

所以：**本地IDE插件必须在沙箱中运行**

挑战2：权限维度冲突

任务：运行10,000个测试用例

本地笔记本方案：
- 8核CPU串行运行
- 耗时：5小时
- 期间电脑无法使用
- 风扇狂转，烫手

Cloud IDE方案：
- 申请100核资源
- 并行运行（100个进程）
- 耗时：3分钟
- 用户浏览器毫无感知

差距：100倍

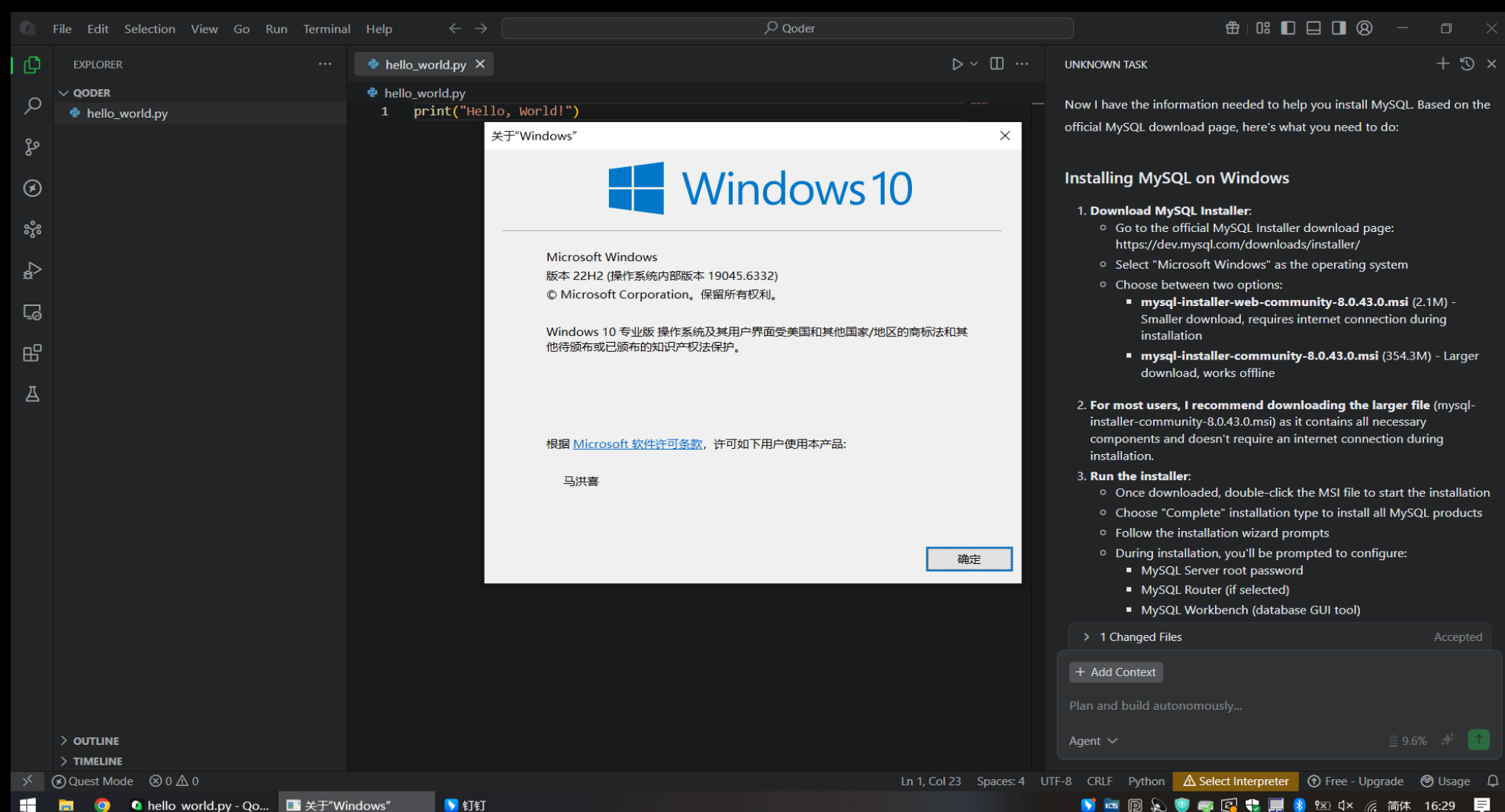
挑战3：资源维度冲突

运行3个Agent：
- 前端Agent（需要8GB内存）
- 后端Agent（需要8GB内存）
- 测试Agent（需要8GB内存）

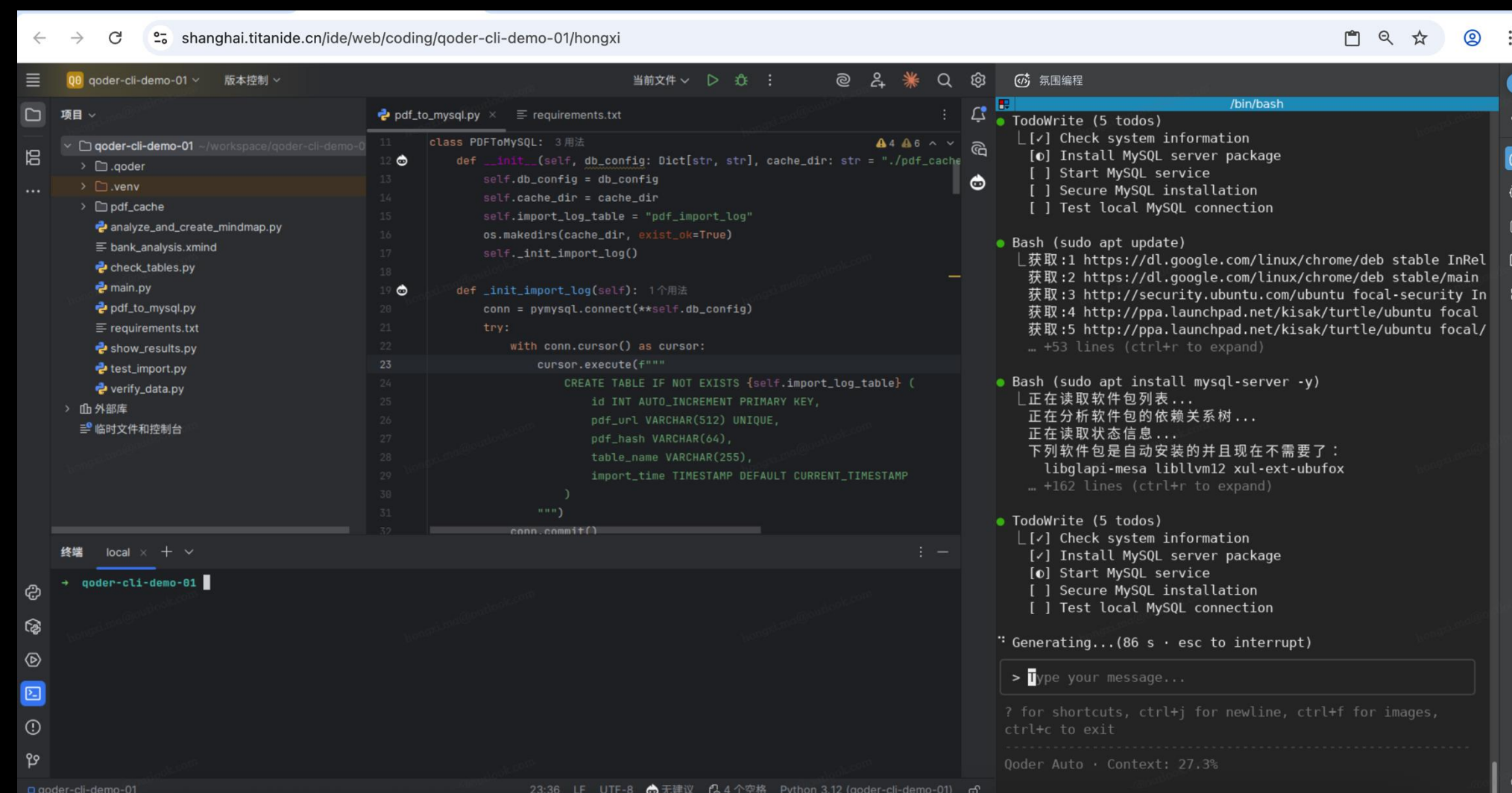
总需求：24GB内存
普通笔记本：16GB内存
→ **物理上不可能**

挑战4：协同维度冲突

我们更喜欢在 Linux 环境使用 Qoder CLI



Windows PC上直接使用AI来安装MySQL，搞不定！



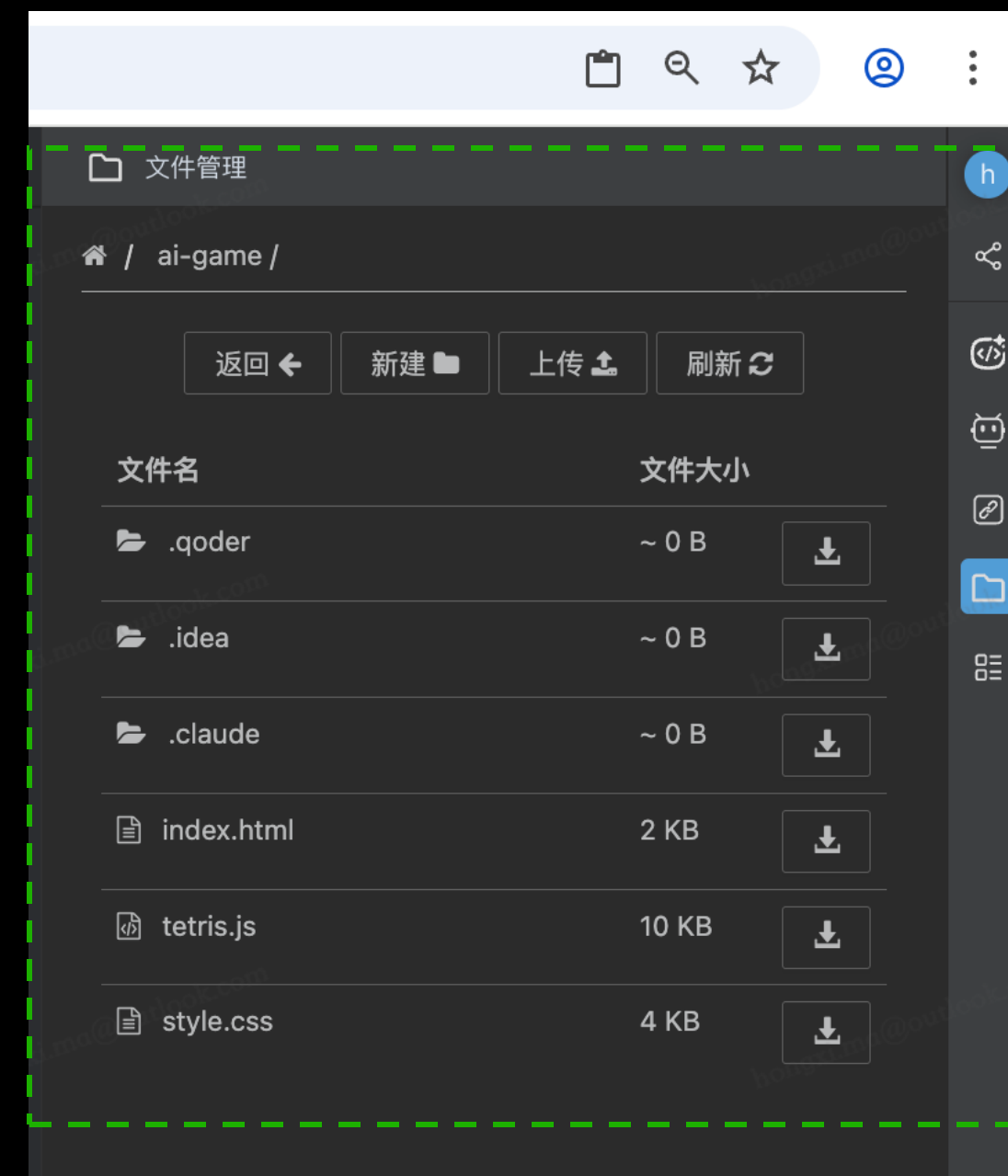
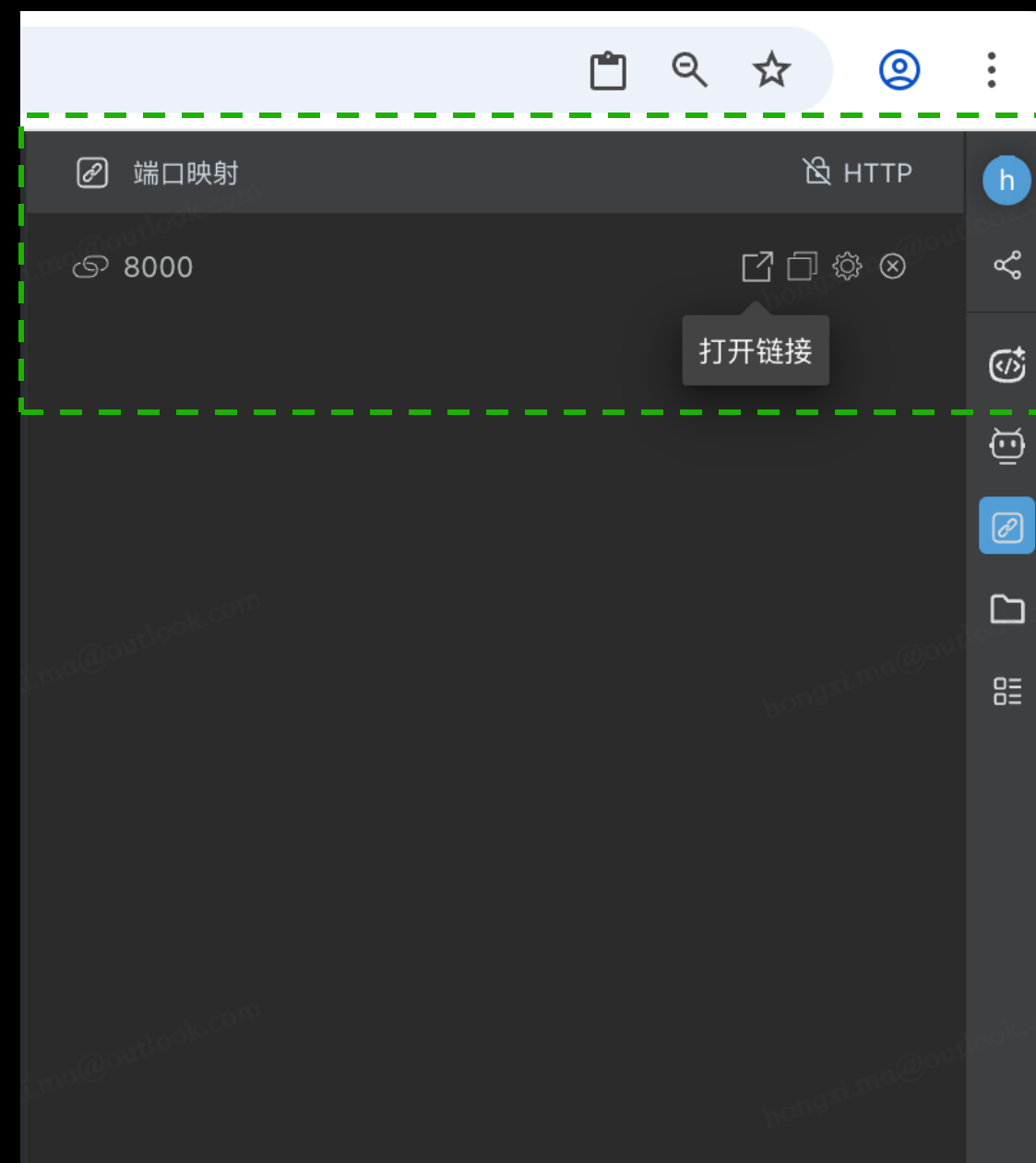
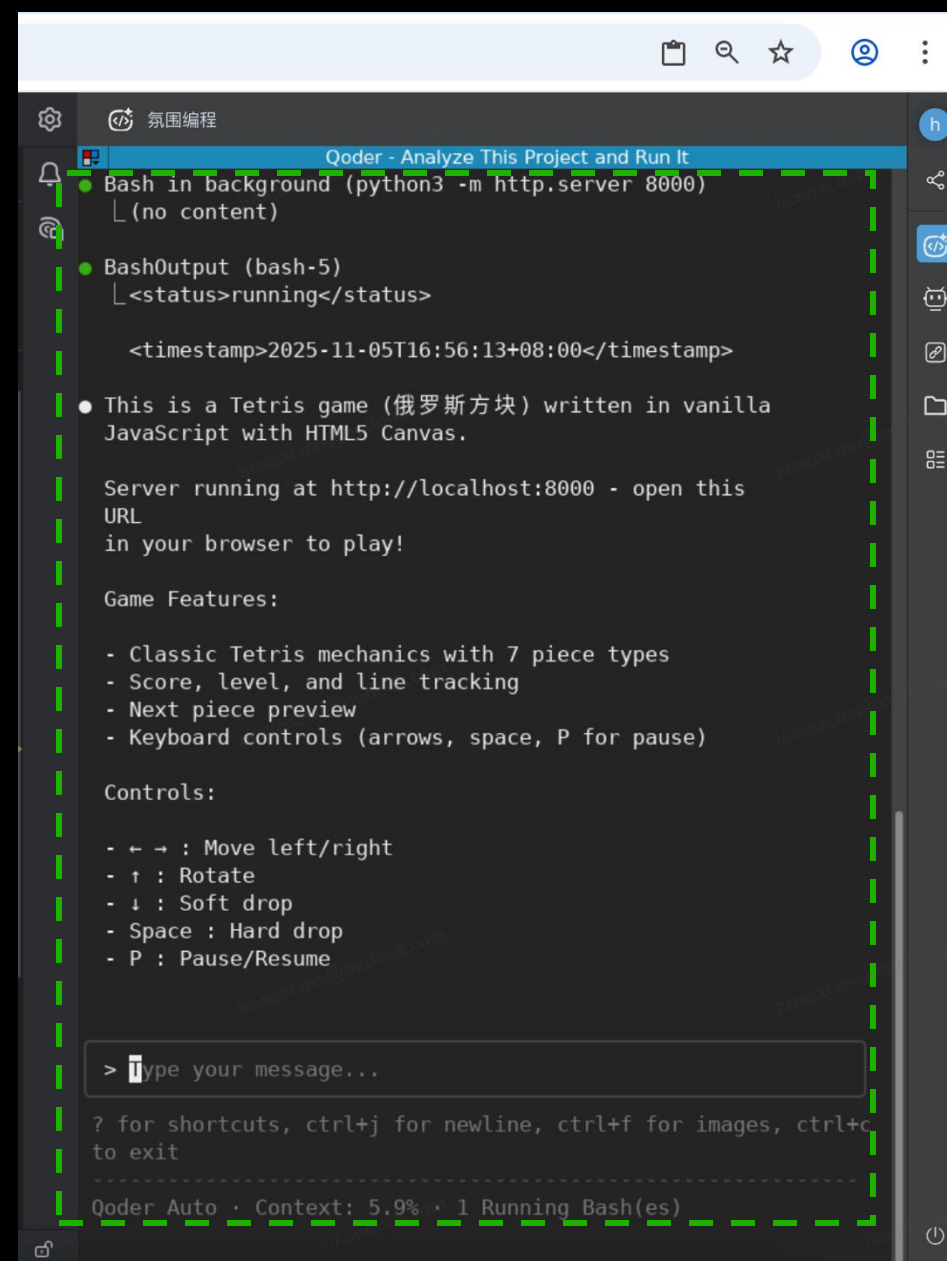
Linux上一句话的事儿，不用担心搞乱: 这是Cloud IDE

深度分析其产生原因

- LINUX和MacOS是命令行起家，核心操作模式是CLI驱动，所以互联网上有大量的相关指令的语料，如：apt install mysql-server；
- Windows 是GUI起家，核心操作模式是鼠标点击，互联网上与其相关的CMD/PowerShell语料极少，就算是有的，但也缺少apt-source这样的配套库，所以得到的是HOWTO INSTALL，这不能怪AI工具，是大模型“学不到”。

K8s里运行 Qoder CLI 要关注的一些技术问题

以网络端口暴露K8s Service & Ingress和文件上传为例



1、生成游戏

2、网络映射

3、测试效果

4、上传底图

The image shows a VS Code editor interface with a Go project controller and a Qoder log. A green arrow points to the Go code with the text "帮你解BUG".

Go Code (project_controller.go):

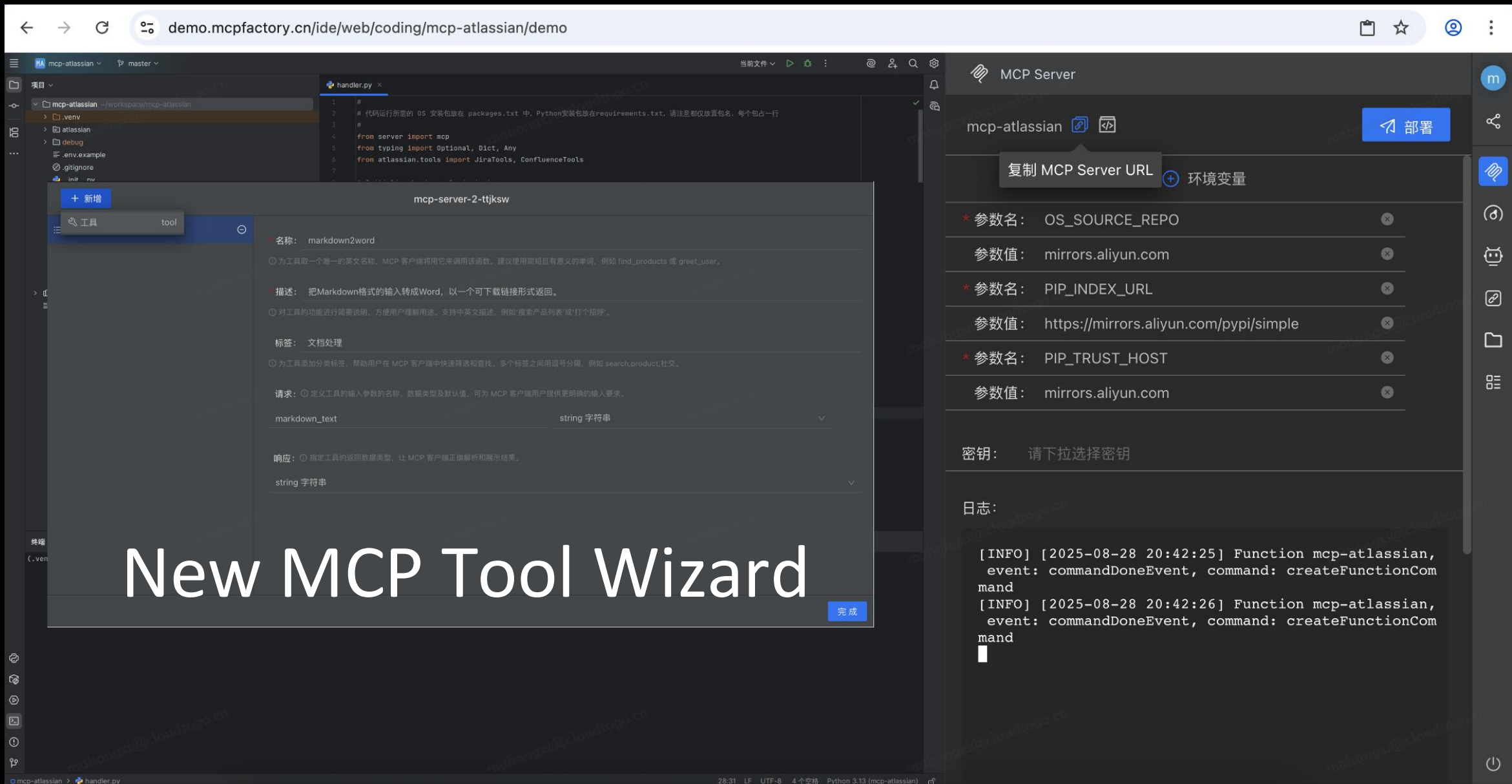
```
10 import (  
11     corev1 "k8s.io/api/core/v1"  
12     storagev1 "k8s.io/api/storage/v1"  
13     "k8s.io/apimachinery/pkg/api/errors"  
14     "k8s.io/apimachinery/pkg/types"  
15     "k8s.io/apimachinery/pkg/util/wait"  
16     ctrl "sigs.k8s.io/controller-runtime"  
17     "sigs.k8s.io/controller-runtime/pkg/client"  
18     "sigs.k8s.io/controller-runtime/pkg/log"  
19 )  
20  
21 "titanide.io/api/v1alpha1"  
22  
23 // ProjectReconciler reconciles a Project object  
24 type ProjectReconciler struct {  
25     client.Client  
26     LockManager *LockManager  
27 }  
28  
29 // LockManager provides distributed locking for project operations  
30 type LockManager struct {  
31     locks sync.Map  
32 }  
33  
34 // acquireLock attempts to acquire a lock for the given key  
35 func (lm *LockManager) acquireLock(key string) bool {  
36     _, loaded := lm.locks.LoadOrStore(key, struct{}{})  
37     return !loaded  
38 }  
39  
40 // releaseLock releases the lock for the given key  
41 func (lm *LockManager) releaseLock(key string) {  
42     lm.locks.Delete(key)  
43 }  
44  
45 // Reconcile handles the reconciliation logic for Project resources  
46 func (r *ProjectReconciler) Reconcile(ctx context.Context, req ctrl.Request) {
```

Qoder Log:

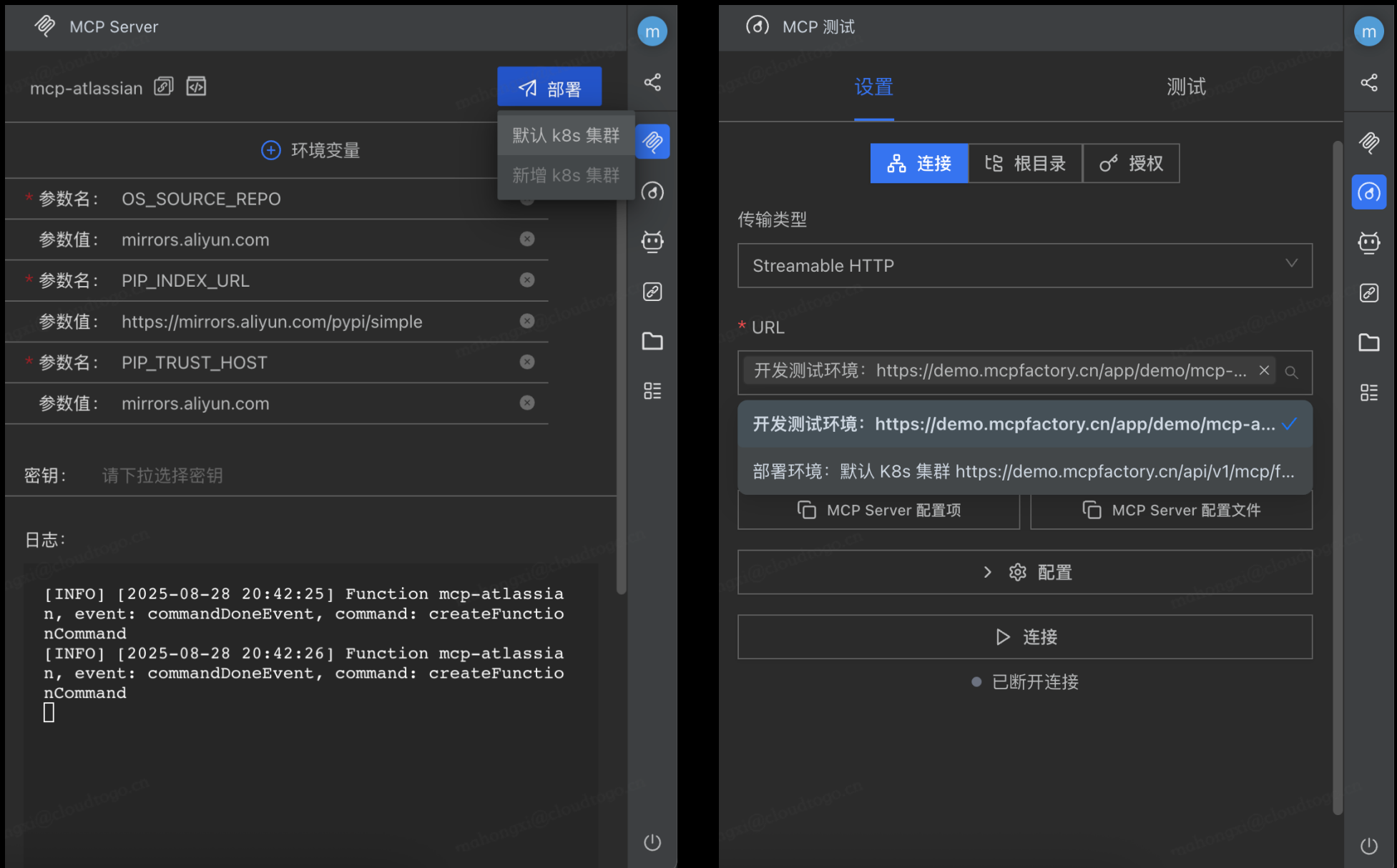
```
Qoder - TitanIDE Open状态故障单  
给出问题解决思路和相关代码修复。  
# IDE-1605 问题解决思路和代码修复方案  
## 问题分析  
核心问题：  
项目关闭后再次启动时，概率性出现挂载存储失败，  
且后台出现两个pod运行  
可能原因：  
1. Pod未完全删除就创建了新Pod，导致存储卷被占用  
2. PVC/PV释放不及时，新Pod无法挂载  
3. StatefulSet或Deployment控制器状态不一致  
4. 存储卷的访问模式 (ReadWriteOnce) 冲突  
5. Pod优雅终止时间不足，强制删除导致卷未正常卸载  
## 解决思路  
### 1. 确保Pod完全删除后再创建  
// pkg/controller/project/project_controller.go  
func (r *ProjectReconciler) stopProject(ctx context.  
Context, project *v1alpha1.Project) error {  
    log := log.FromContext(ctx)  
  
    // 1. 获取当前运行的Pod  
    podList := &corev1.PodList{}  
    err := r.List(ctx, podList, client.  
InNamespace(project.Namespace),  
    client.MatchingLabels{  
        "app.kubernetes.io/name": project.Name,  
        "app.kubernetes.io/instance": project.Spec.  
Owner,  
    })  
    if err != nil {  
        return fmt.Errorf("failed to list pods: %w",  
err)
```

给出问题解决思路和相关代码修复。

But MCP Servers 从哪儿来?



云函数 + Qoder CLI 构建的MCP专有开发工具



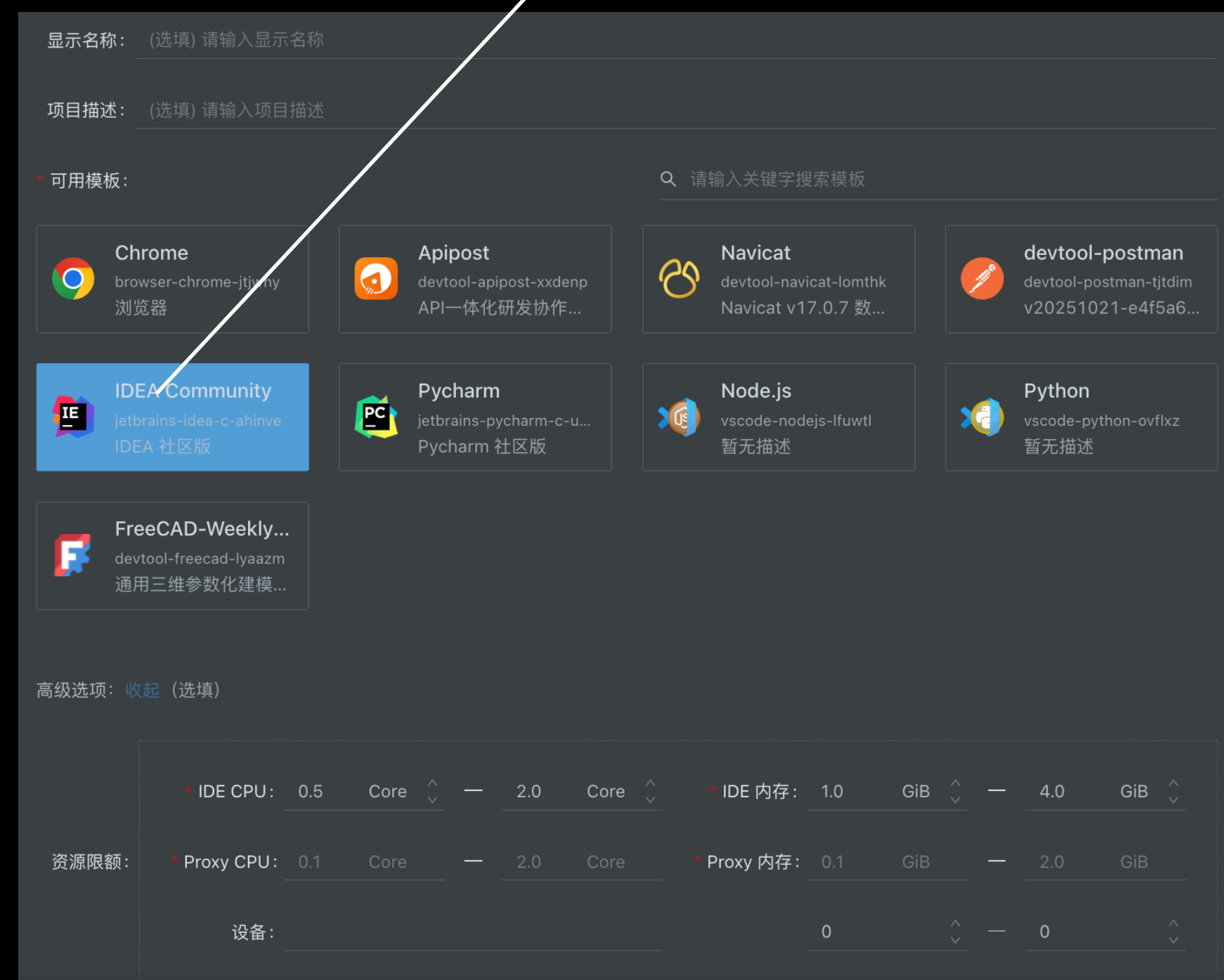
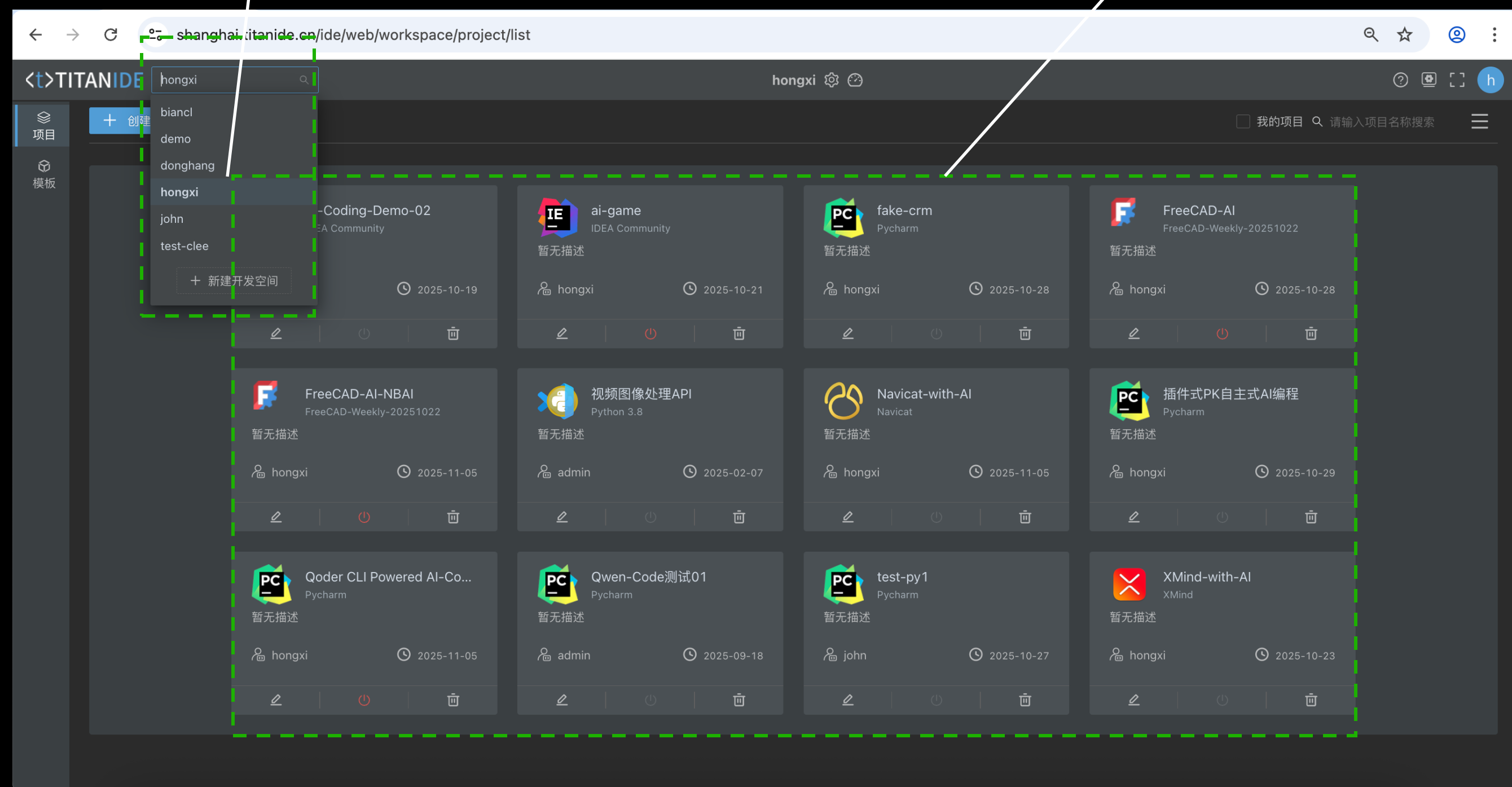
集MCP Servers的部署、测试、运营于一体

团队一起使用 Qoder CLI，我们采用统筹管理的方案

在K8s 不同的Namespace下
协同或是独立开发

不同角色的各类工具
都配备了Qoder CLI

在开发模板中内置了Qoder CLI和
开发习惯、约束的提示词以及MCP Servers



AI 生成的代码也需要被保护

shanghai.titanide.cn/ide/web/workspace/project/list

安全

shanghai.titanide.cn

连接是安全的
您发送给这个网站的信息（例如密码或信用卡号）不会外泄。[了解详情](#)

证书有效

Go语言开发-交易接口 2024-02-02 11:26:12

addr eq 122.112.252.95 and ip.addr eq 192.168.110.66

Time	Source	Destination	Protocol	Length	Info
257 8.135115	122.112.252.95	192.168.110.66	TCP	60	443
258 8.300101	192.168.110.66	122.112.252.95	TLSv1.2	122	App
259 8.334459	122.112.252.95	192.168.110.66	TLSv1.2	113	App
260 8.334459	122.112.252.95	192.168.110.66	TLSv1.2	1188	App
261 8.334515	192.168.110.66	122.112.252.95	TCP	54	50090

0000 68 14 01 42 34 b1 c8 c4 65 79 98 af 08 00 45 00 h..B4...ey...E-

0010 04 96 55 8e 40 00 29 06 52 19 7a 70 fc 5f c0 a8 ..U.@.) R-zp_...

0020 6e 42 01 bb c3 aa aa 79 88 54 be 96 81 76 50 18 nB...y T...vP-

0030 03 26 8a 43 00 00 17 03 03 04 69 fd 44 1c 41 0b .&C...iD-A-

0040 94 bb 3b 42 50 1f e6 40 10 ac 1b 5a 45 56 cc b3 .;BP...@...ZEV-

0050 a4 ae 49 15 25 c8 e3 06 23 f9 71 ee 4a 4d 4f e2 ..I.%...#-qJMO-

0060 59 80 60 fe 7c 76 2f 97 48 41 bb 09 d7 bc 65 d9 Y...|v/ HA...e-

0070 bf 30 69 c3 10 b6 87 de 1c 23 0e 9c 1b 31 b8 b6 .0i...-#...1..

0080 82 53 d2 3b f0 cc 5c 78 44 b2 85 b5 98 40 79 63 .S;...x D...@yc

传输信道安全：
采用 TLS RSA4K加密传输，杜绝了中间人拦截以及浏览器外部破解数据的可能性。

helloworld.go

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     fmt.Println(a... "Hello, World!")
7 }
8
```

Name

active

data: image/webp;ba...

data: image/webp;ba...

data: image/webp;ba...

data: image/webp;ba...

data: image/webp;ba...

data: image/webp;ba...

data: image/webp;ba...

Headers

Preview

func main() {

传输内容安全：
在服务器侧对内容进行图形切片，采用webp 协议传输变量部分。到达用户侧是图型切块，不是文字。

helloworld.go

```
1 package main
2
3 import "fmt"
4
5 func main() {
6     fmt.Println(a... "Hello, World!")
7 }
8
```

显示上下文操作 Alt+Enter

剪切(I) Ctrl+X

复制(C) Ctrl+C

粘贴(P) Ctrl+V

复制/粘贴特殊

列选择模式(M) Alt+Shift+Insert

在文件中查找

重构(R)

未命名 - 已编辑

Helvetica 常规体 12

5DBE5E1F258B770DD0CAD7ABDACECF7848314B7C1EB7B0179156

B646EE13470AD31E36B71489CD3A88649E28ACF722AEA5DCA9492

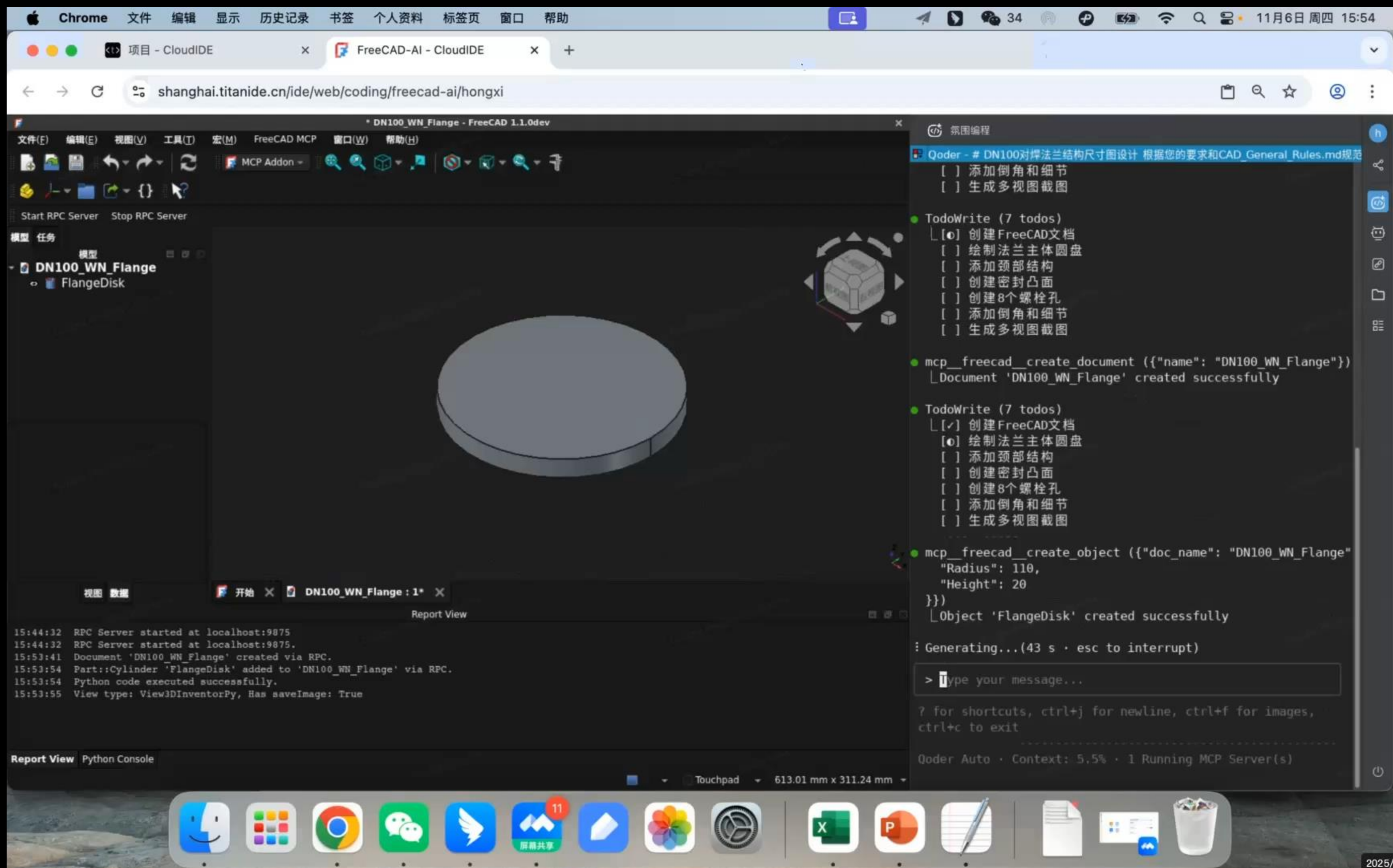
3704283D2F9C1C563401ADEF419FFA715F939F941AF9F8F0DA00A4

6C3%8A%C3%96%C3%A1%C3%8C

复制粘贴安全：
打开相应开关后，IDE 内的代码等数据向外部复制时，会经由服务器端加密逻辑形成乱码。

除前述机制外，还包括文件下载安全、调试访问安全、开发空间安全、登录认证安全等，以及配套的审计机制。

有趣的扩展，Qoder CLI 在 CAD 领域也能大展身手？！



Thank You!

欢迎一起交流学习



科技生态圈峰会 + 深度研习



——1000+ 技术团队的选择

NiDD 峰会

NiDD 峰会 上海站

AI+研发数字峰会

时间: 2026.05.22-23

NiDD 峰会 北京站

AI+研发数字峰会

时间: 2026.08.21-22

NiDD 峰会 深圳站

AI+研发数字峰会

时间: 2026.11.20-21



AiDD峰会详情

NiDD × AI+产品峰会

NiDD × 上海站

AI+产品创新峰会

时间: 2026.01.16-17

NiDD × 北京站

AI+产品创新峰会

时间: 2026.08.23-24



产品峰会详情



2026 AI+PRODUCT INNOVATION SUMMIT

01.16-17 · ShangHai

AI+产品创新峰会



Track 1: AI 产品战略与创新设计

从0到1的AI原生产品构建

论坛1: AI时代的用户洞察与需求发现

论坛2: AI原生产品战略与商业模式重构

论坛3: Agentic AI产品创新与交互设计

2-hour Speech: 回归本质



用户洞察的第一性

——2小时思维与方法论工作坊

在数字爆炸、AI快速发展的时代，
仍然考验“看见”的“同理心”



Track 2: AI 产品开发与工程实践

从1到10的工程化落地实践

论坛1: 面向Agent智能体的产品开发

论坛2: 具身智能与AI硬件产品

论坛3: AI产品出海与本地化开发

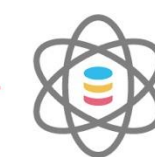
Panel 1: 出海前瞻



“出海避坑地图”圆桌对话

——不止于翻译：

AI时代的出海新范式



Track 3: AI 产品运营与智能演化

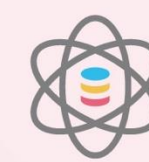
从10到100的AI产品运营

论坛1: AI赋能产品运营与增长黑客

论坛2: AI产品的数据飞轮与智能演化

论坛3: 行业爆款AI产品案例拆解

Panel 2: 失败复盘



为什么很多AI产品“叫好不叫座”？

——从伪需求到真价值：

AI产品商业化落地的关键挑战

智能重构产品 数据驱动增长

Reinventing Products with Intelligence, Driven by Data

80+

业界大咖

汇聚产品大咖的真知灼见

40+

创新案例

开拓全新AI+产品视角

10+

分论坛

涵盖产品到商业增长的全链路

800+

听众规模

共同探索产品的智能重构



扫码查看会议详情



第8届 AI+研发数字峰会

AI+ Development Digital Summit

感谢聆听!

扫码领取会议PPT资料

