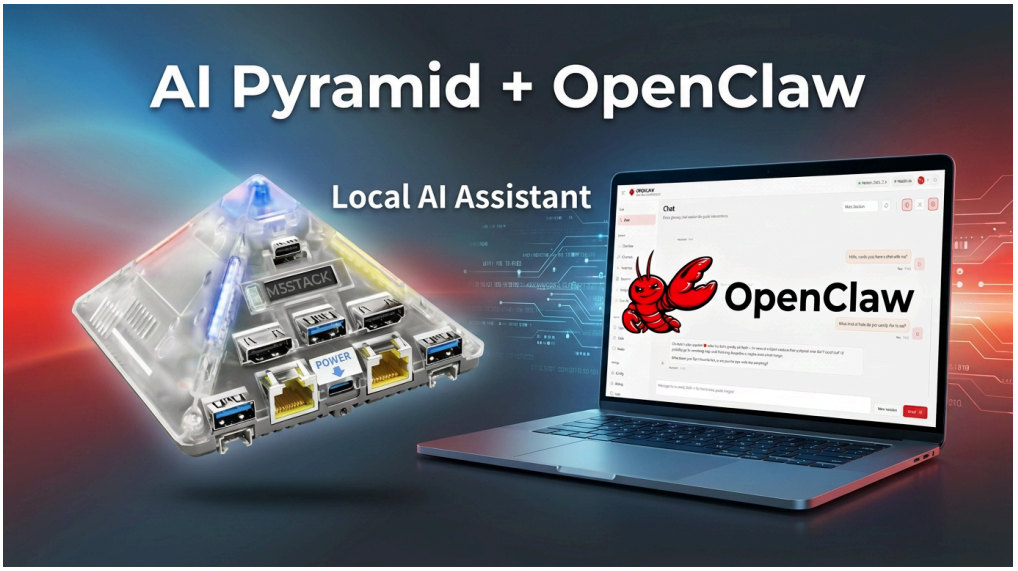
AI Pyramid Deployment of OpenClaw

1. Introduction

OpenClaw is an open-source AI Agent framework that can connect to local or cloud Large Language Models (LLMs) to achieve automated tasks and intelligent interaction. This document will introduce how to perform image flashing, initial configuration, and initial access for OpenClaw on **AI Pyramid**.



2. OpenClaw Image Burning

Download the OpenClaw image file below and refer to AI Pyramid Image Burning to complete the burning.

Firmware Version	Download Link
AI_Pyramid_openclaw_emmc_ubuntu_rootfs_desktop_V3.6.4	Download

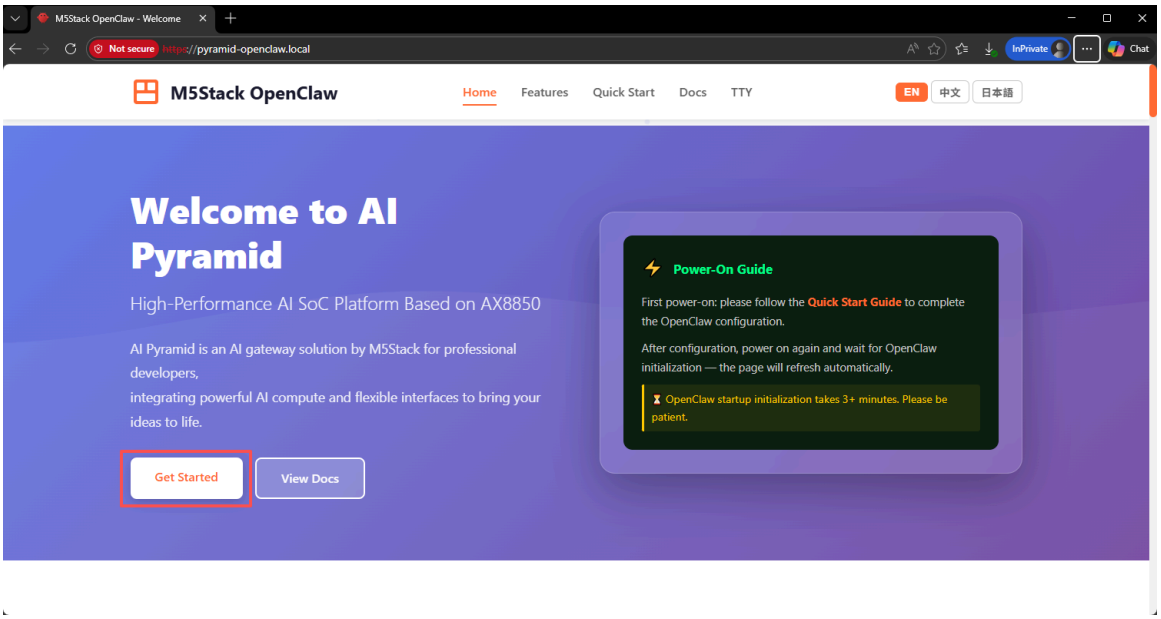
3. Logging into the Device

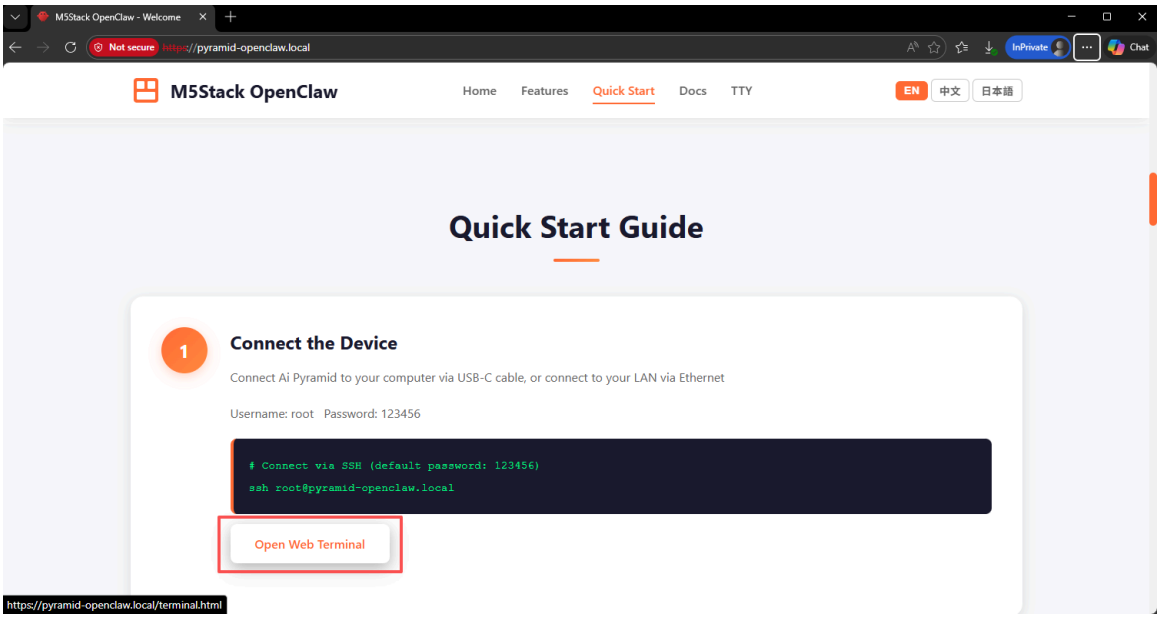
After completing the image burning, the device will automatically enable the OpenClaw initialization boot page (only displayed when the initialization is not completed).

3.1 Entering the Web Terminal

LAN Access (Recommended)

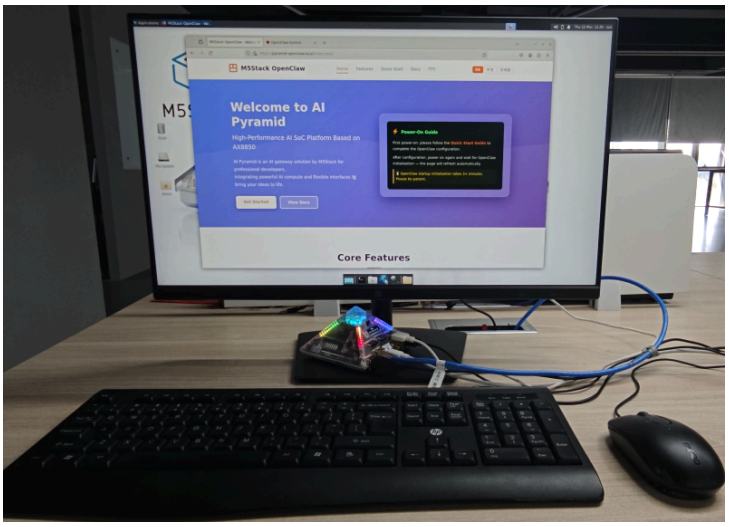
1. Power on the AI Pyramid and connect it to the Ethernet.
2. On a computer within the same local area network, access `https://pyramid-openclaw.local` through a browser.
3. Scroll down to `Get Started` and click `Open Web Terminal` to enter the terminal.





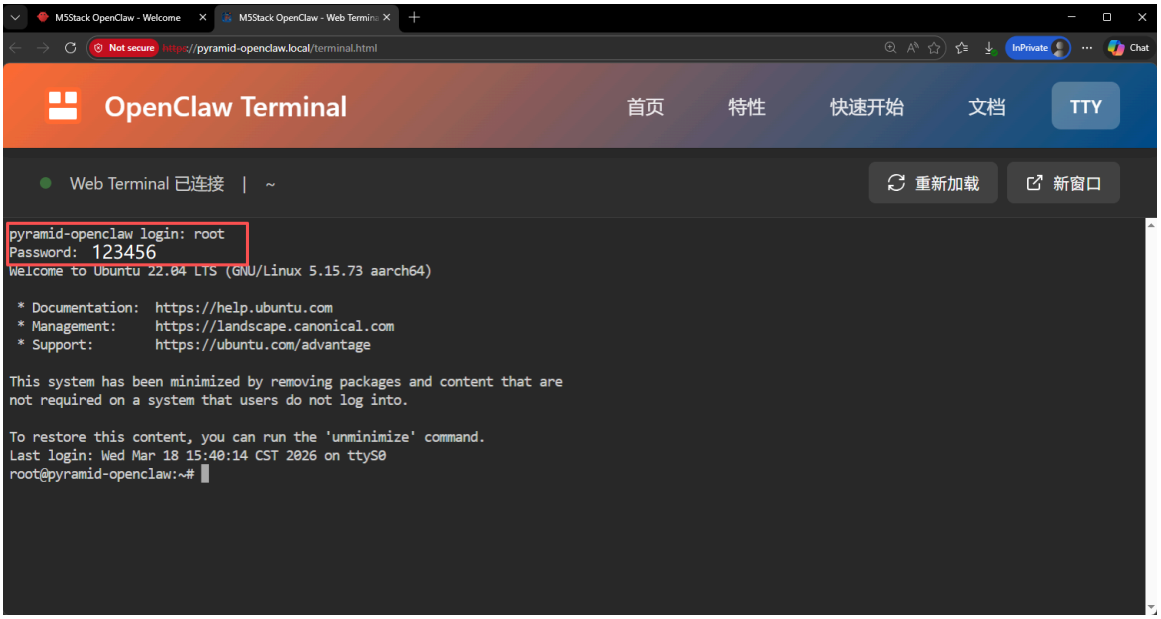
Local Desktop Access

Connect a monitor, mouse, and keyboard to the AI Pyramid. In the desktop browser, access `https://pyramid-openclaw.local`. The operation method is the same as LAN access.



3.2 Logging into the Terminal

In the Web Terminal, enter the username `root` and password `123456` to log into the device terminal:



4. OpenClaw Initialization

Take the QuickStart configuration method to access the `Qwen` API as an example. Directly run the following commands to start the initialization (the device is pre - installed with the OpenClaw service):

1. Execute the initialization command.

```
openclaw onboard
```

Tip: Initialization usually takes about 2 minutes. Please be patient.

2. Agree to the security prompt: After reading the security prompt, select `Yes` to continue:

```
◆ I understand this is personal - by - default and shared/multi - user use requires lock - down. Continue?
● Yes / ○ No
```

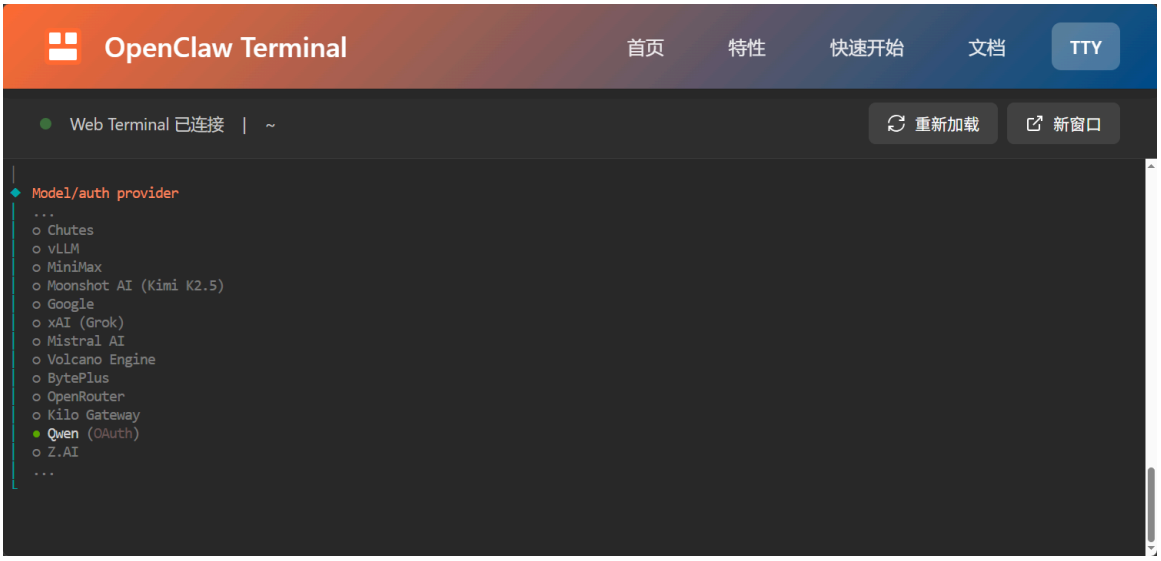
3. Select the configuration mode: Select `QuickStart`:

```
◆ Onboarding mode
│ ● QuickStart (Configure details later via openclaw configure.)
│ ○ Manual
```

4. Select the AI model provider.

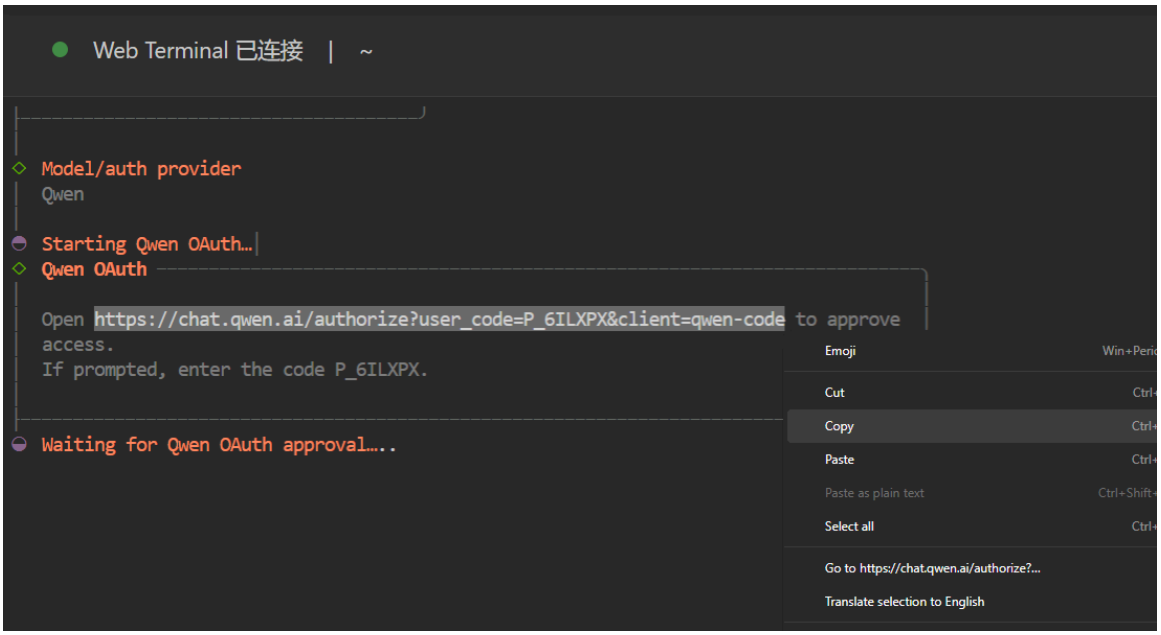
In this case, `Qwen (OAuth)` is used as an example. You can also choose other providers according to your subscription:

```
◆ Model/auth provider
│ ○ OpenAI
│ ○ Anthropic
│ ● Qwen (OAuth)
│ ○ OpenRouter
│ ○ Skip for now
```



5. Complete the OAuth authentication: Copy the displayed link and open it in a browser to complete the Qwen OAuth authentication (register first if you don't have an account):

```
◇ Qwen OAuth
│ Open https://chat.qwen.ai/authorize?user_code = 1 - _VFKMX&client = qwen - code to approve access.
│ If prompted, enter the code 1 - _VFKMX.
```



6. Confirm the default model: After successful authentication, select `Keep current` to keep the default model:

```
◆ Default model
│ ● Keep current (qwen - portal/coder - model)
│ ○ Enter model manually
```

7. Skip the channel configuration: Select `Skip for now` to skip the communication channel configuration:

```
◆ Select channel (QuickStart)
│ ○ Telegram (Bot API)
│ ○ Discord (Bot API)
│ ● Skip for now
```

8. Skip the Skills installation: Select `No` to not install Skills for now:

```
◆ Configure skills now? (recommended)
│ ○ Yes / ● No
```

9. Skip the Hooks configuration: Press the spacebar to select `Skip for now` and press Enter to confirm:

```
◆ Enable hooks?
│ ■ Skip for now
│ □ 🚀 boot - md
│ □ 📝 command - logger
```

10. Save the Token: After the initialization is completed, be sure to save the displayed `token` value. This is the credential for accessing the OpenClaw console later:

```
◇  Dashboard ready
│ Dashboard link (with token):
│ http://127.0.0.1:18789/#token = 0c4a936a196d3be5812e7624d5cb419b0b364c308f68b799
│ Copy/paste this URL in a browser on this machine to control OpenClaw.
```

11. Configure LAN access: Execute the following commands in sequence in the terminal to enable LAN access:

```
openclaw config set gateway.bind lan
openclaw config set gateway.controlUi.allowInsecureAuth true
openclaw config set gateway.controlUi.allowedOrigins '["http://127.0.0.1","http://127.0.0.1:18789","https://
openclaw config set gateway.controlUi.enabled true
```

12. Access the OpenClaw console: Open the console address in a computer browser (replace `YOUR_TOKEN_VALUE` with the token saved in step 10):

Within the same local area network, access through the following domain name plus the token parameter:

```
https://pyramid-openclaw.local#token = YOUR_TOKEN_VALUE
```

Precautions

Currently, accessing the console through an IP address is not supported. Only domain name access is supported. If the browser prompts "Not a private connection" or other security warnings, you can click the advanced options and choose to continue accessing.

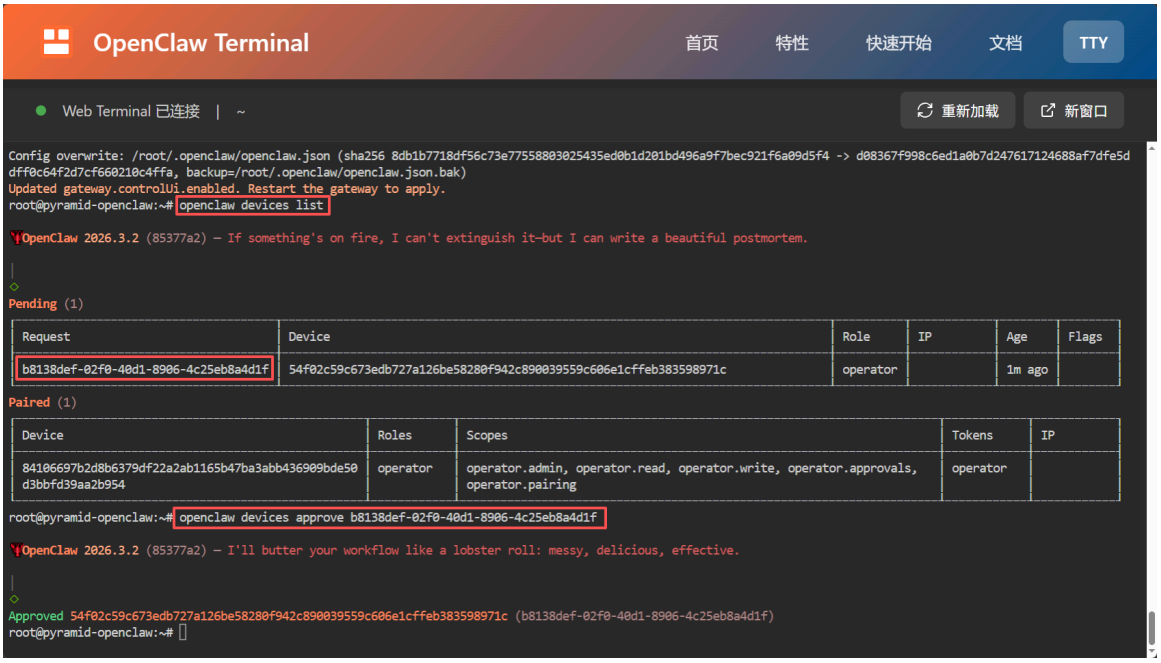
13. Device pairing: The first time you access the console, you need to complete device pairing. Execute the following command in the device terminal to view the devices to be paired:

```
openclaw devices list
```

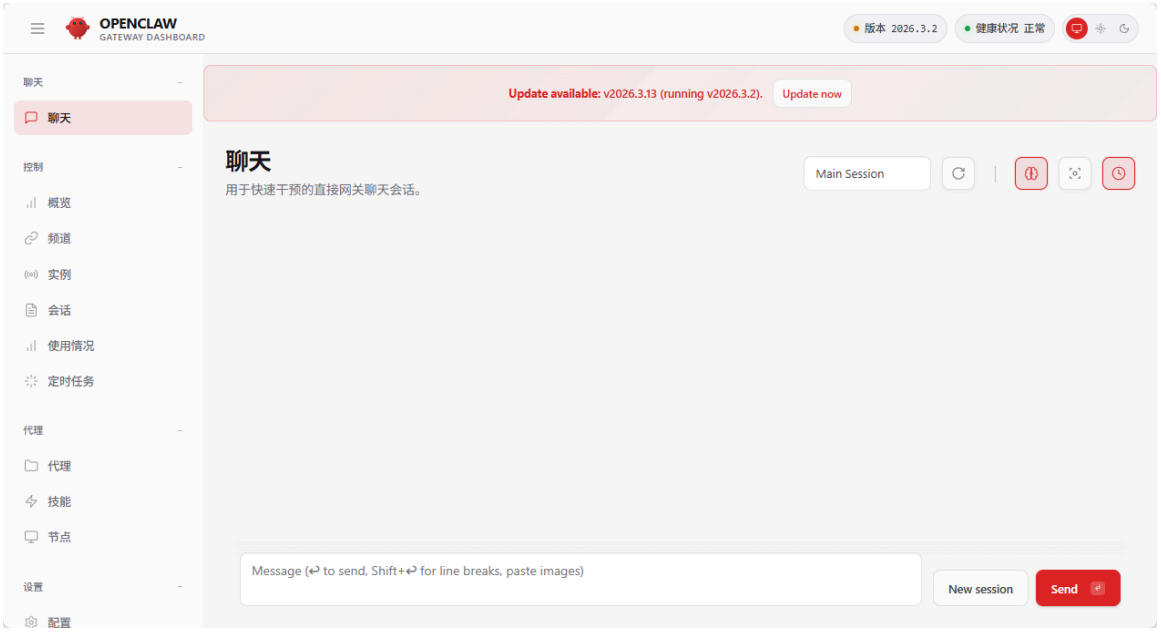
Then approve the pairing request (replace `<requestId>` with the actual device ID):

Note: The requestId is only valid for the current browser. If you change the device or browser, you need to view the requestId again and re - initiate the pairing.

```
openclaw devices approve <requestId>
```



After the configuration is completed, the console can be accessed normally on the web - side.

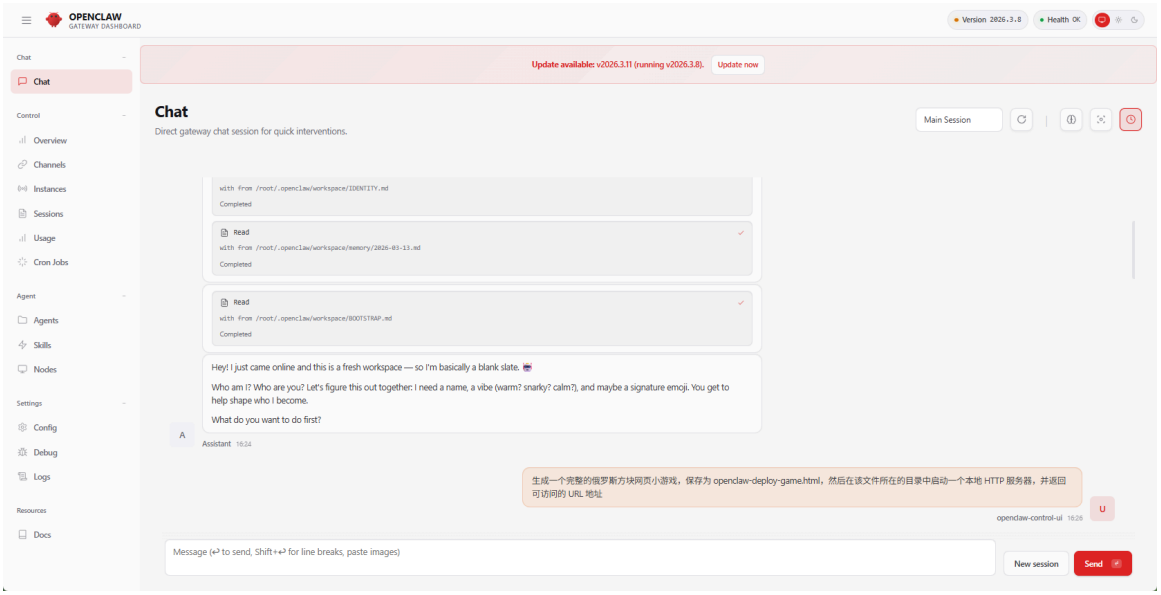


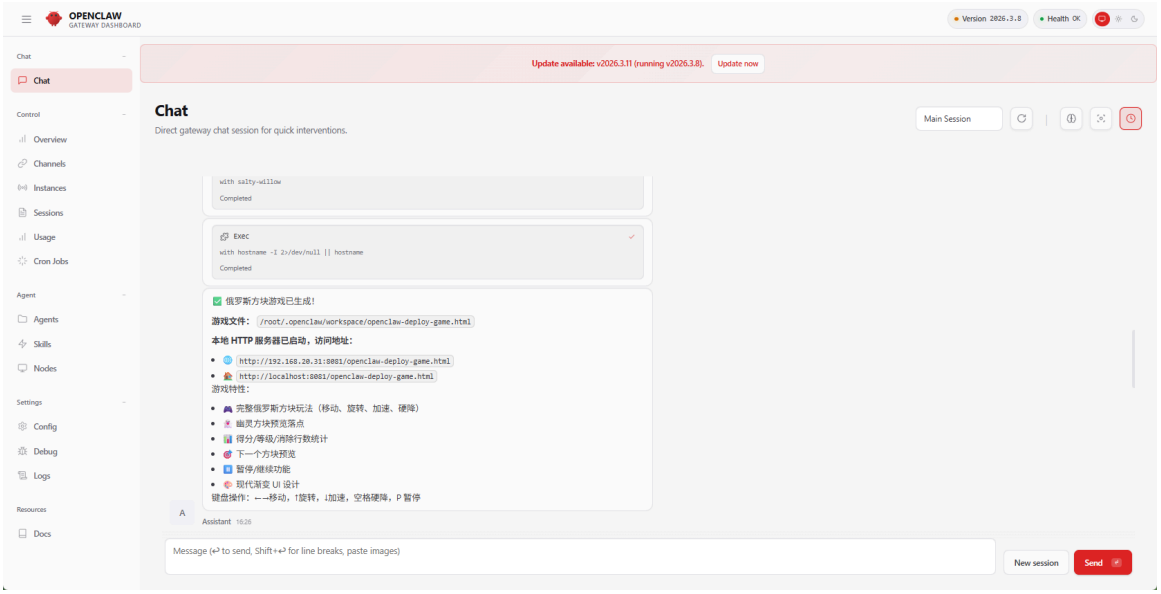
5. Effect Demonstration

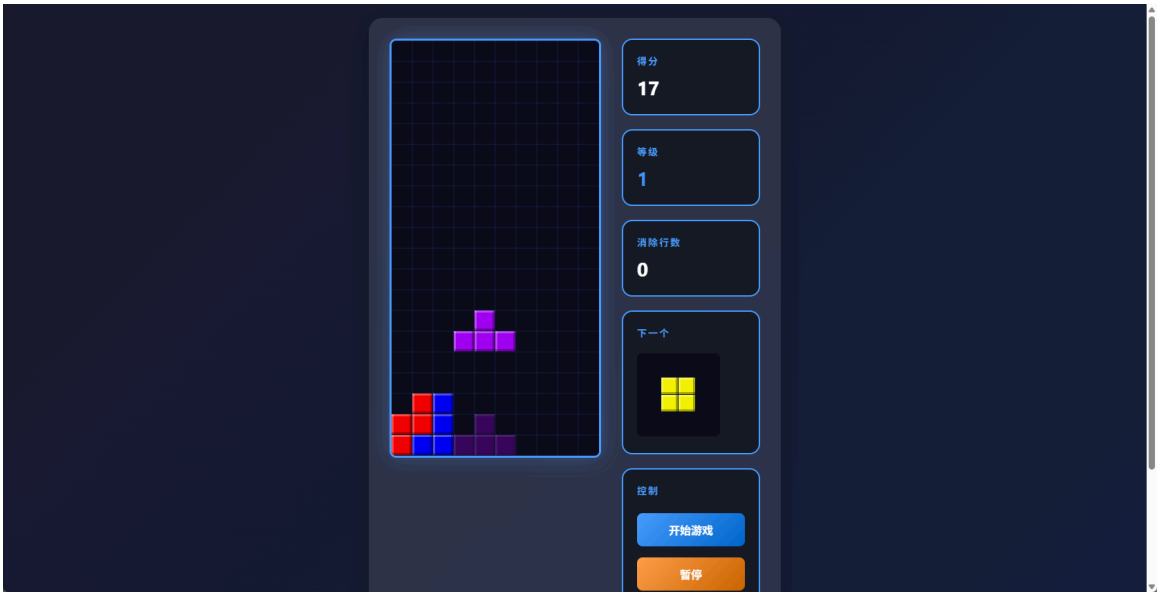
Submit the following task in the OpenClaw console to verify the deployment effect:

> Generate a complete Tetris web mini - game, save it as openclaw - deploy - game.html, then start a local HTTP server in the directory where the file is located, and return the accessible URL address.

After OpenClaw completes the task, copy the returned URL and open it in a browser to view the mini - game effect.


OPENCLAW
GATEWAY DASHBOARD
Version 2026.3.8
Health OK
Update available: v2026.3.11 (running v2026.3.8). Update now
Chat
Direct gateway chat session for quick interventions.
Main Session
Chat
Control
Overview
Channels
Instances
Sessions
Usage
Cron Jobs
Agent
Agents
Skills
Nodes
Settings
Config
Debug
Logs
Resources
Docs
with from /root/.openclaw/workspace/IDENTITY.md
Completed
Read
with from /root/.openclaw/workspace/memory/2026-03-13.md
Completed
Read
with from /root/.openclaw/workspace/BOOTSTRAP.md
Completed
Hey! I just came online and this is a fresh workspace — so I'm basically a blank slate.
Who am I? Who are you? Let's figure this out together: I need a name, a vibe (warm? snarky? calm?), and maybe a signature emoji. You get to help shape who I become.
What do you want to do first?
Assistant 16:24
生成一个完整的俄罗斯方块网页小游戏，保存为 openclaw-deploy-game.html，然后在该文件所在的目录中启动一个本地 HTTP 服务器，并返回可访问的 URL 地址
openclaw-control-ui 16:26
Message (↩ to send, Shift+↩ for line breaks, paste images)
New session
Send


OPENCLAW
GATEWAY DASHBOARD
Version 2026.3.8
Health OK
Update available: v2026.3.11 (running v2026.3.8). Update now
Chat
Direct gateway chat session for quick interventions.
Main Session
with salty-willow
Completed
Exec
with hostname -I 2>/dev/null || hostname
Completed
俄罗斯方块游戏已生成！
游戏文件： /root/.openclaw/workspace/openclaw-deploy-game.html
本地 HTTP 服务器已启动，访问地址：
http://192.168.20.31:8081/openclaw-deploy-game.html
http://localhost:8081/openclaw-deploy-game.html
游戏特性：
完整俄罗斯方块玩法（移动、旋转、加速、硬降）
幽灵方块预览落点
得分/等级/消除行数统计
下一个方块预览
暂停/继续功能
现代渐变 UI 设计
键盘操作：←→移动，↑旋转，↓加速，空格硬降，P 暂停
Assistant 16:26
Message (↩ to send, Shift+↩ for line breaks, paste images)
New session
Send


得分
17
等级
1
消除行数
0
下一个
控制
开始游戏
暂停

6. Video