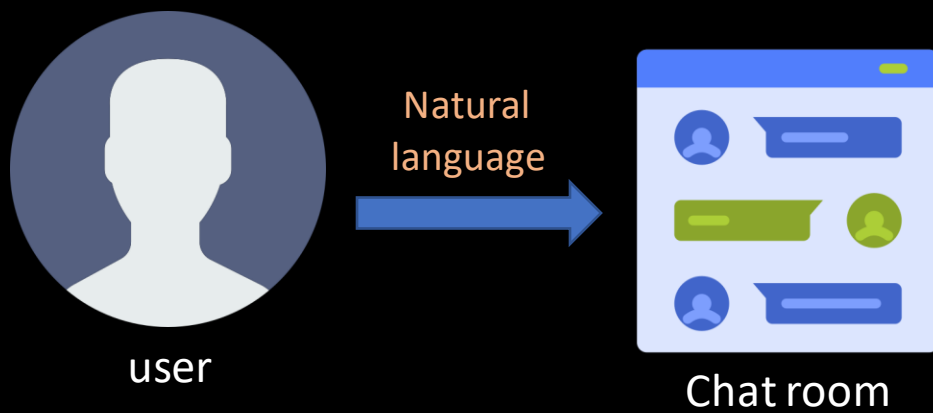
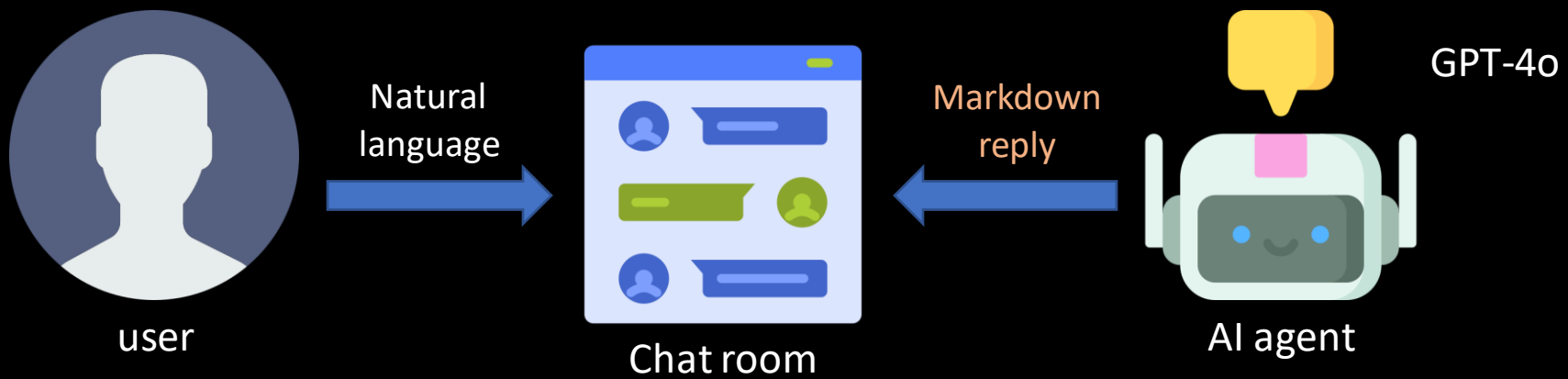


AI_Agent_Remote_Control _RaspberryPi

Github: https://github.com/CrowHatter/AI_Agent_Remote_Control_RaspberryPi





- 聊天室 1
- 聊天室 2

在桌面建立 Python 檔案

首先，我們需要在桌面建立一個 Python 檔案，並在檔案中寫入程式碼來印出 "Hello, World!" 。

1. 首先，切換至桌面目錄：

BASHCopy

```
cd ~/Desktop
```

2. 接著，使用以下指令建立一個新的 Python 檔案 (例如 `hello.py`)：

BASHCopy

```
echo "print('Hello, World!')" > hello.py
```

編輯 Python 檔案

現在我們已經在桌面建立了一個 Python 檔案 `hello.py`，接下來我們可以編輯這個檔案，讓它真正印出 "Hello, World!" 。

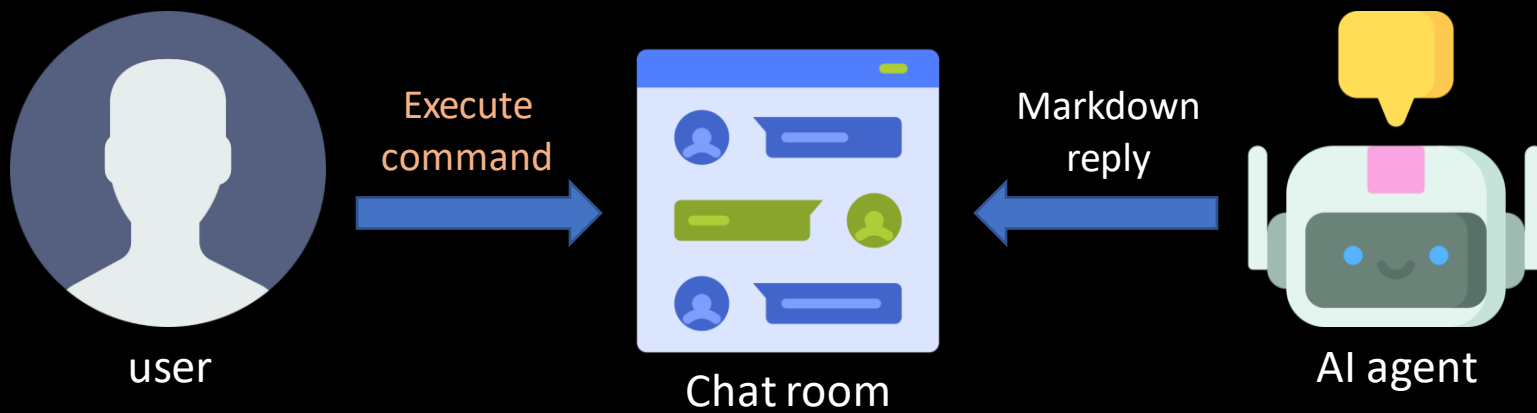
1. 使用以下指令編輯 `hello.py` 檔案：

BASHCopy

```
nano hello.py
```

2. 在 Nano 編輯器中，將游標移至 `print('Hello, World!')` 這一行，修改程式碼或做任何其他必要的編輯。





- 聊天室 1
- 聊天室 2

```
echo "print('Hello, World!')" > hello.py
```

編輯 Python 檔案

現在我們已經在桌面建立了一個 Python 檔案 `hello.py`，接下來我們可以編輯這個檔案，讓它真正印出 "Hello, World!"。

1. 使用以下指令編輯 `hello.py` 檔案：

BASH

```
nano hello.py
```

2. 在 Nano 編輯器中，將游標移至 `print('Hello, World!')` 這一行，修改程式碼或做任何其他必要的編輯。

3. 按下 `Ctrl + X`，再依序按下 `y` 確認儲存，最後按下 `Enter` 離開 Nano 編輯器。

執行 Python 檔案

最後，我們可以執行剛剛建立的 Python 檔案 `hello.py`，來印出 "Hello, World!"。

1. 使用以下指令執行 `hello.py` 檔案：

BASH

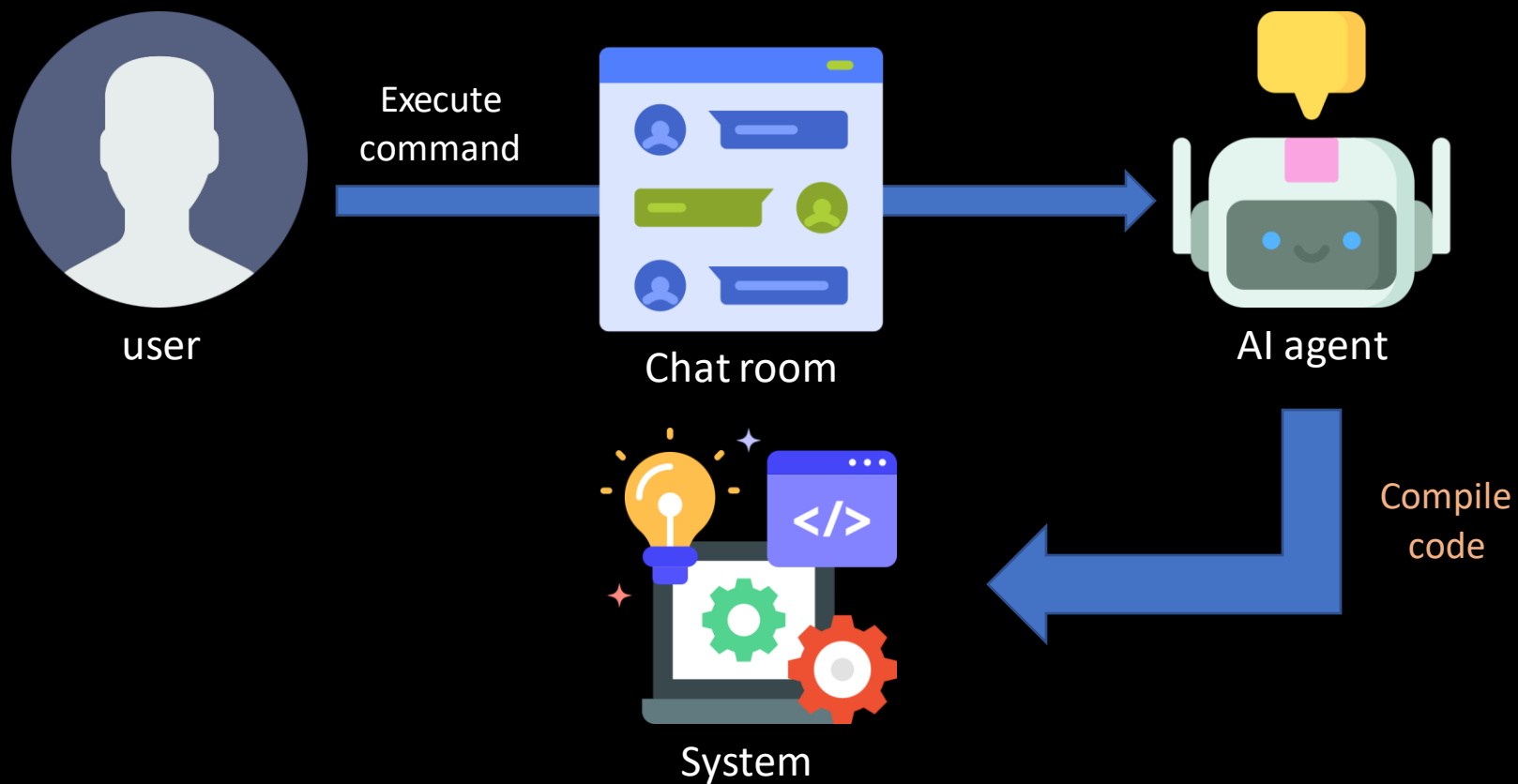
```
python3 hello.py
```

透過以上步驟，您已在桌面建立了一個能印出 "Hello, World!" 的 Python 檔案並成功執行。如果有任何問題或需要進一步協助，請隨時讓我知道！

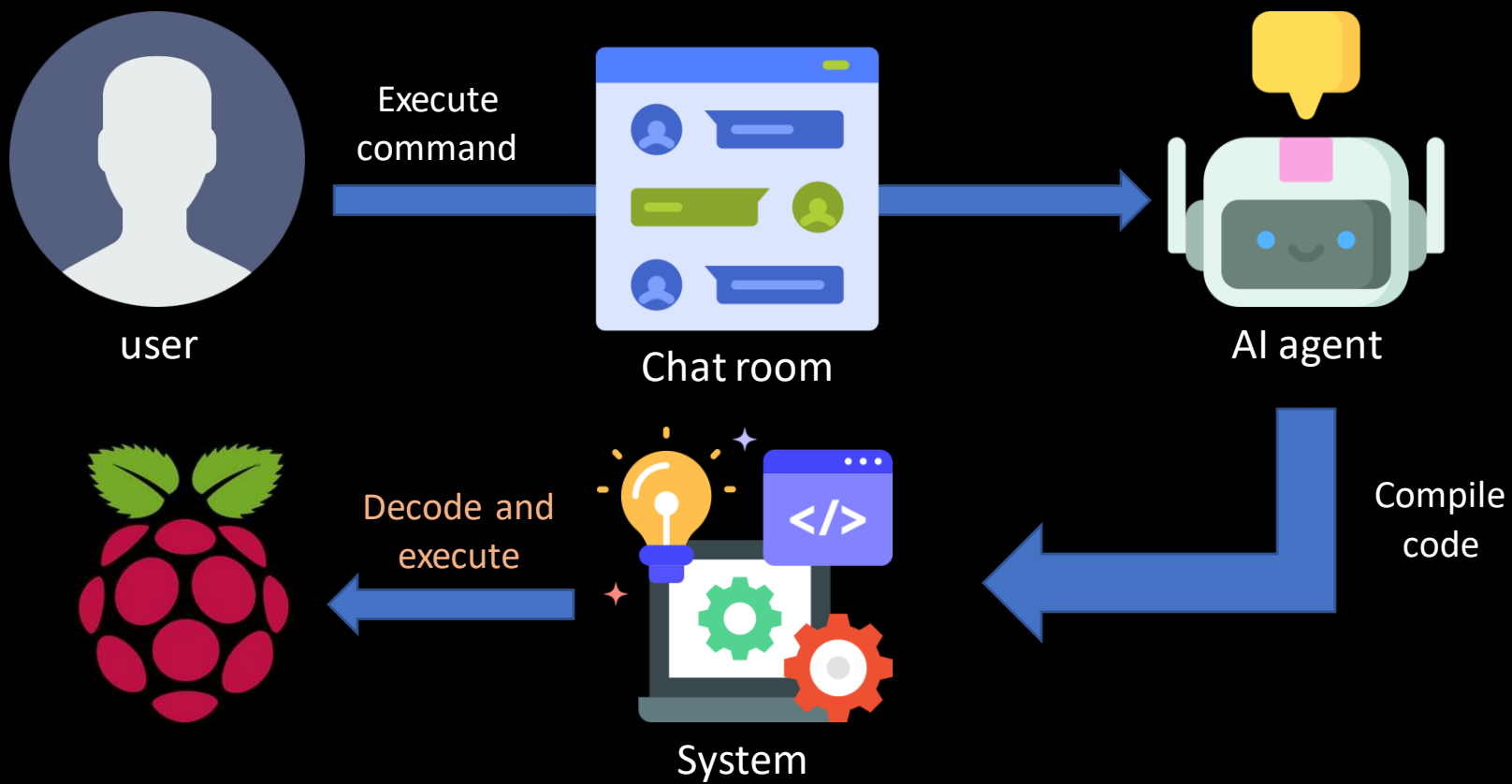
請輸入訊息...

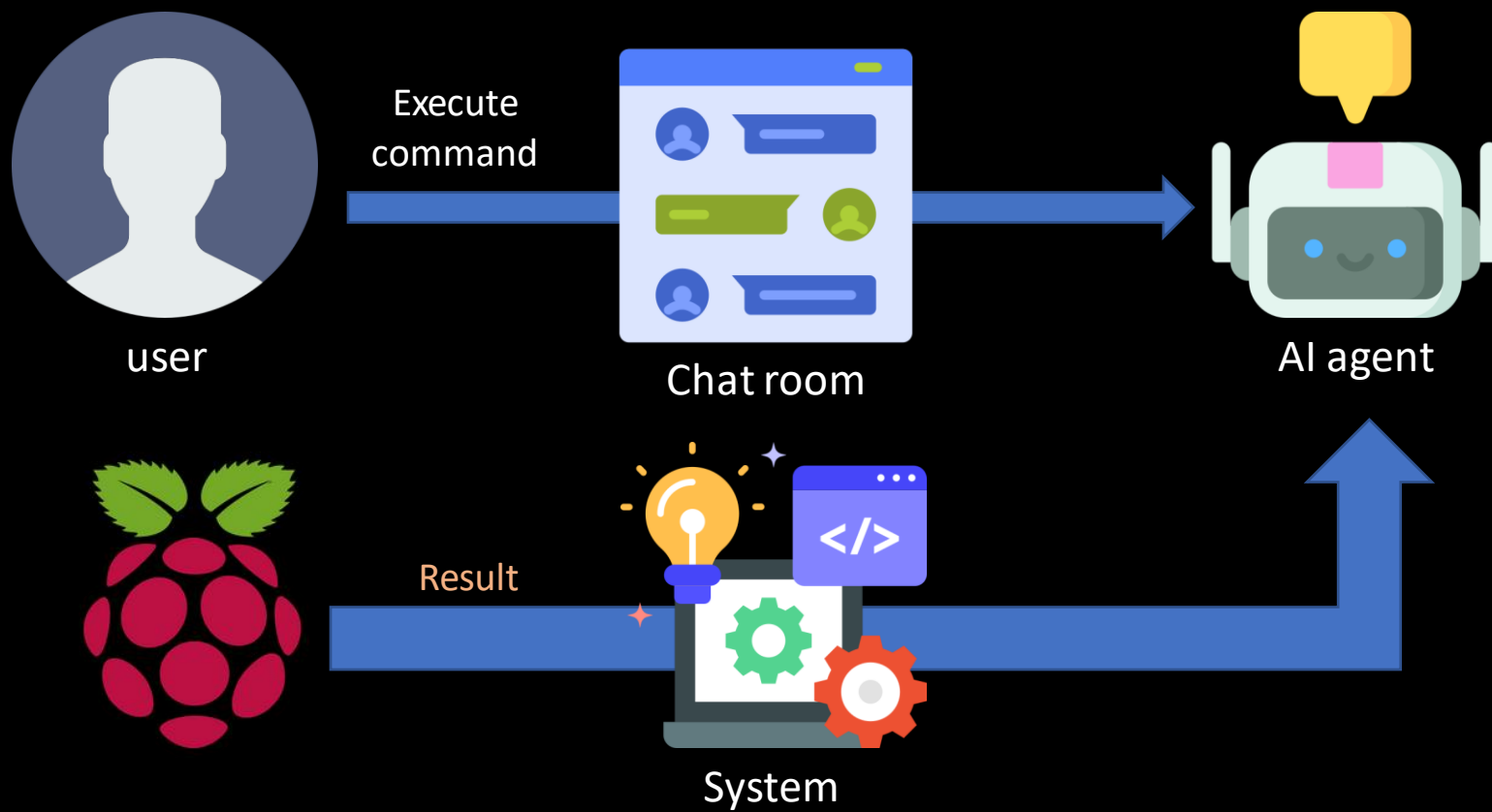
▶

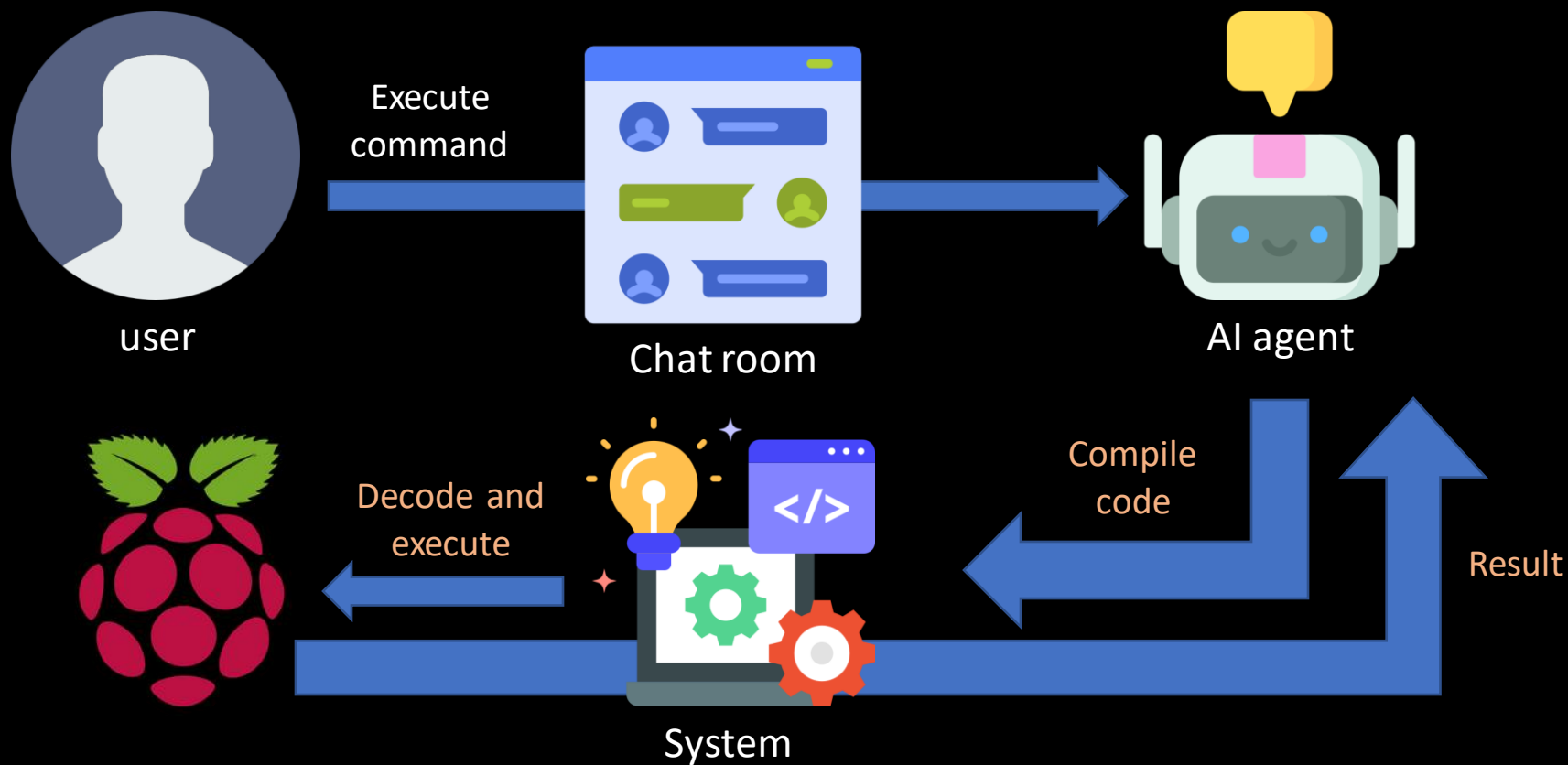




```
{"ExecuteCommand": "<打算執行之CLI指令>"  
{"ExecuteInvokeShellCommand": "<打算執行之指令>"}
```







Loop multiple times

- 聊天室 1
- 聊天室 2

```
echo "print('Hello, World!')" > hello.py
```

編輯 Python 檔案

現在我們已經在桌面建立了一個 Python 檔案 `hello.py`，接下來我們可以編輯這個檔案，讓它真正印出 "Hello, World!"。

- 使用以下指令編輯 `hello.py` 檔案：

BASH

nano hello.py

Copy

- 在 Nano 編輯器中，將游標移至 `print('Hello, World!')` 這一行，修改程式碼或做任何其他必要的編輯。

- 按下 `Ctrl + X`，再依序按下 `v` 確認儲存，最後按下 `Enter` 離開 Nano 編輯器。

執行 Python 檔案

最後，我們可以執行剛剛建立的 Python 檔案 `hello.py`，來印出 "Hello, World!"。

- 使用以下指令執行 `hello.py` 檔案：

BASH

python3 hello.py

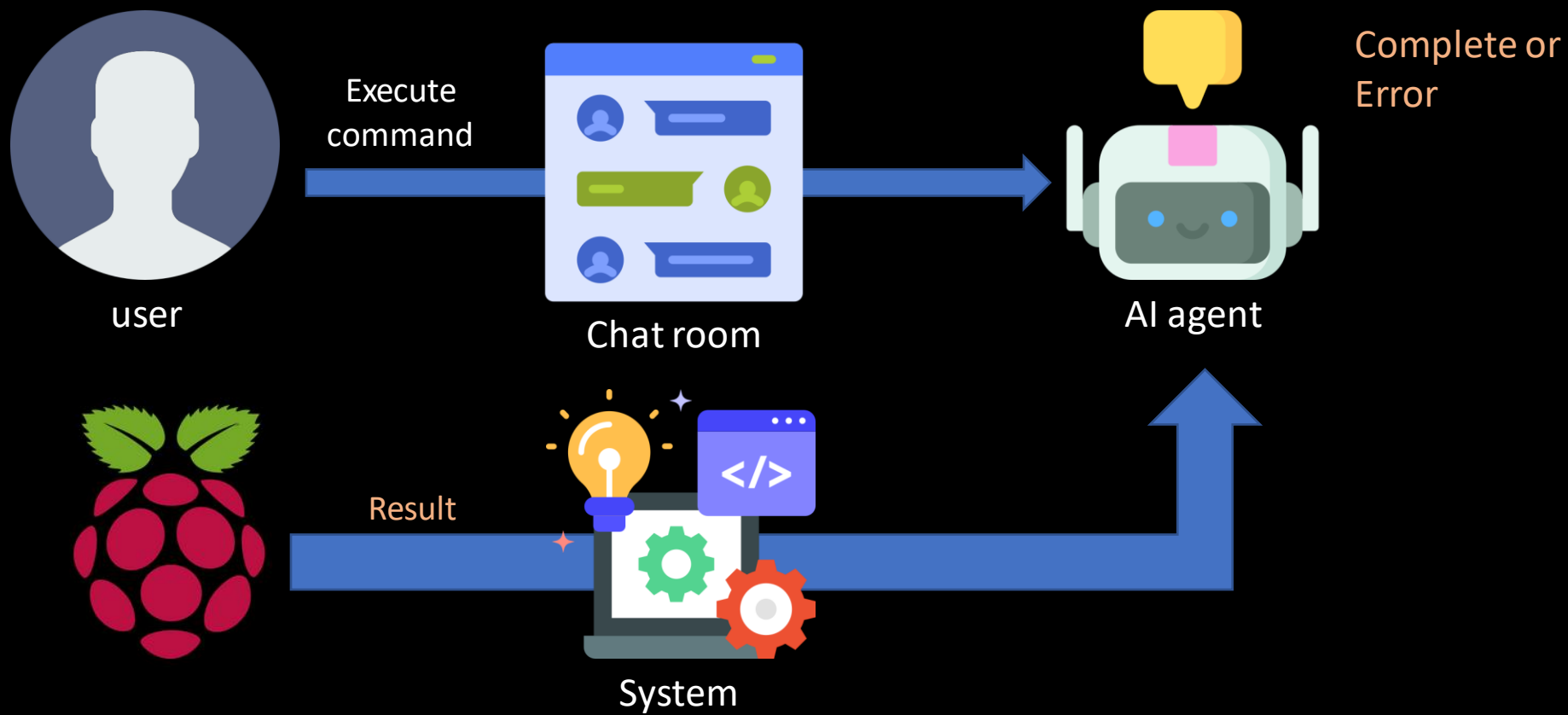
Copy

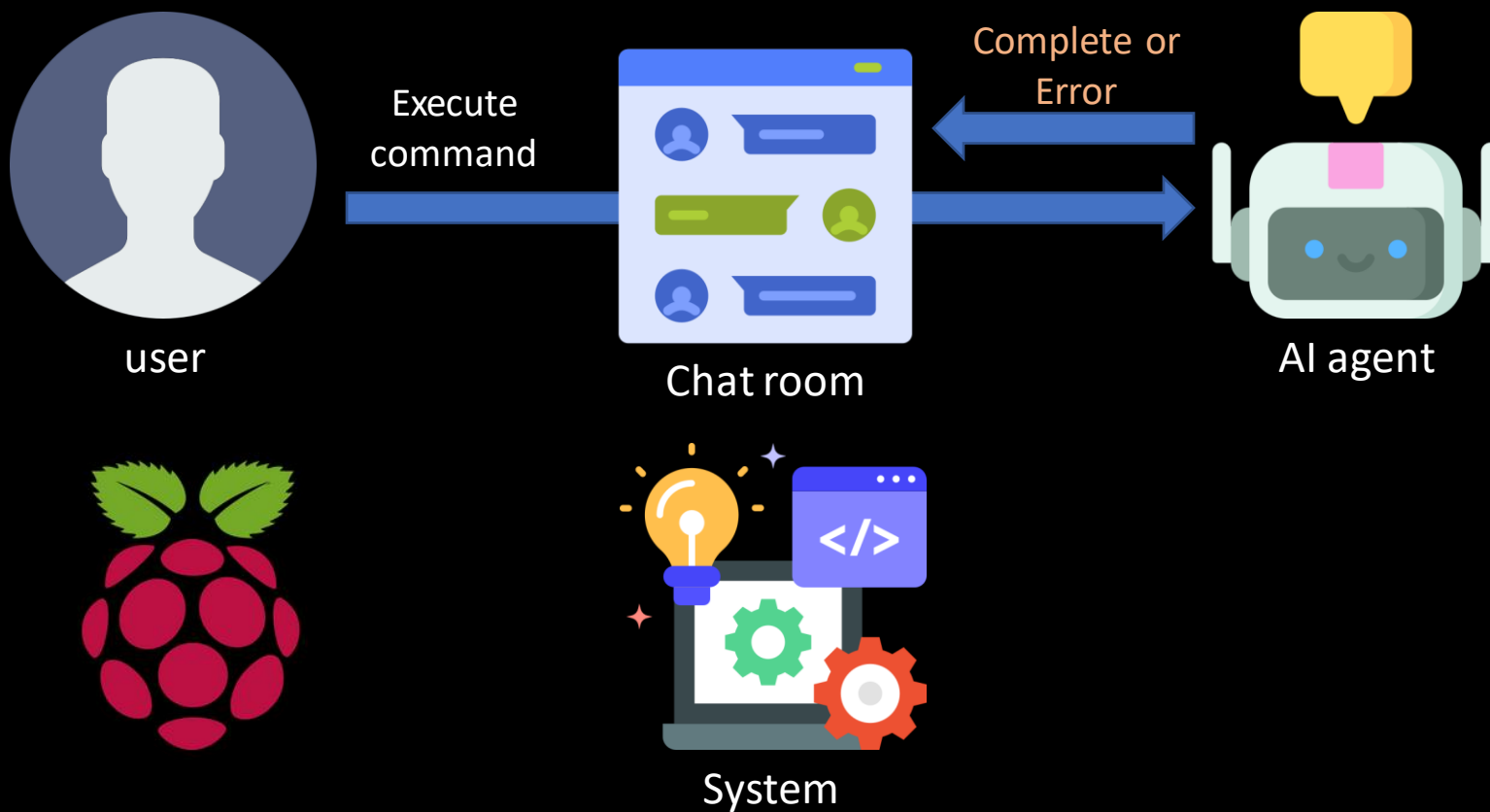
透過以上步驟，您已在桌面建立了一個能印出 "Hello, World!" 的 Python 檔案並成功執行。如果有任何問題或需要進一步協助，請隨時讓我
知道！

Executing

請輸入訊息...







```
{"Error": {  
  "ExecutedCommand": "<剛才執行的指令>",  
  "RaspberryPiOutput": "<實際輸出結果>",  
  "ExpectedBehavior": "<本應該看到或期望出現的結果描述>"  
}}
```

```
{"Complete": "All commands executed  
successfully."}
```

- 聊天室 1
- 聊天室 2

1. 使用以下指令編輯 `hello.py` 檔案：

BASH

`nano hello.py`

Copy

2. 在 Nano 編輯器中，將游標移至 `print('Hello, World!')` 這一行，修改程式碼或做任何其他必要的編輯。

3. 按下 `Ctrl + X`，再依序按下 `Y` 確認儲存，最後按下 `Enter` 離開 Nano 編輯器。

執行 Python 檔案

最後，我們可以執行剛剛建立的 Python 檔案 `hello.py`，來印出 "Hello, World!"。

1. 使用以下指令執行 `hello.py` 檔案：

BASH

`python3 hello.py`

Copy

透過以上步驟，您已在桌面建立了一個能印出 "Hello, World!" 的 Python 檔案並成功執行。如果有任何問題或需要進一步協助，請隨時讓我知道！

Execute

("Complete": "All commands executed successfully.")

感謝您的回覆，讓我知道所有指令都成功執行了！如果您有任何其他問題或需要進一步協助，請隨時告訴我。我隨時樂意協助您！

ericweng@AIAgentForRasberr

Active code page: 65001

C:\Users\EricWeng>ssh 192.168.1.104

ericweng@192.168.1.104's password:

Linux AIAgentForRaspberryPi 6.6.51+rpt-rpi-v8 #1 SMP PREEMPT Debian 1:6.6.51-1+rpt3 (2024-10-08) aarch64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

Last login: Tue Apr 8 04:42:46 2025 from 192.168.1.132

ericweng@AIAgentForRaspberryPi:~ \$ ls

Bookshelf Desktop Documents Downloads hello.py Music Pictures Public Templates Videos

ericweng@AIAgentForRaspberryPi:~ \$ |

```
{"Er  
"Exe  
"Ras  
"Exp  
}
```

ed

- 聊天室 1
- 聊天室 2

4. 儲存並關閉編輯器，系統將會每隔10分鐘重新啟動一次。

請注意，這個設定會讓系統每10分鐘就重新啟動一次，請確保這是您真正想要的操作，以免造成資料遺失或不便。

Execute

{"Error":{"ExecutedCommand":"crontab -e","RaspberryPiOutput":"無法啟動編輯器","ExpectedBehavior":"啟動 crontab 編輯器以便進行編輯"}}

啟動 crontab 編輯器進行設定

如果無法直接啟動 crontab 編輯器，您可以嘗試以下替代方法：

1. 使用 **VISUAL** 環境變數來指定編輯器，例如使用 **nano** 編輯器：

BASH

Copy

```
export VISUAL=nano
crontab -e
```

2. 接著您應該可以在 **nano** 編輯器中進行 crontab 的設定。請將以下設定加入到 crontab 中：

BASH

Copy

```
*/10 * * * * /sbin/shutdown -r now
```

3. 儲存並關閉編輯器，系統將會每隔10分鐘重新啟動一次。

這樣應該可以讓您成功設定每10分鐘重新啟動的工作排程。如果有任何問題或需要進一步協助，請隨時讓我

Execute

請輸入訊息...



Future Directions

- Bug fix
 - Shell
 - Initial prompt
- Chat setting
 - API key
 - Model
 - Device
- MCP
 - Code upload & download