

HAPI FHIR Server

LINUX

1. 準備一台Linux環境的電腦(或虛擬機)

```
ubuntu@vm1704854465831-4317628-iaas: ~  
D:\工作\twcc>ssh -i .\kpmopatalsppt.pem ubuntu@103.124.72.82  
The authenticity of host '103.124.72.82 (103.124.72.82)' can't be established.  
ECDSA key fingerprint is SHA256:+5waM1/rBsY+uZxE4P7WauDeLPOWHQprQJiuXDmsJrE.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? y  
Please type 'yes', 'no' or the fingerprint: yes  
Warning: Permanently added '103.124.72.82' (ECDSA) to the list of known hosts.  
Welcome to Ubuntu 22.04.3 LTS (GNU/Linux 5.15.0-91-generic x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:       https://ubuntu.com/advantage  
  
System information as of Wed Jan 10 02:47:35 PM CST 2024  
  
System load:  0.0          Processes:      97  
Usage of /:   3.3% of 93.73GB  Users logged in: 0  
Memory usage: 1%          IPv4 address for ens3: 192.168.211.18  
Swap usage:   0%  
  
Expanded Security Maintenance for Applications is not enabled.  
  
0 updates can be applied immediately.  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
ubuntu@vm1704854465831-4317628-iaas:~$
```

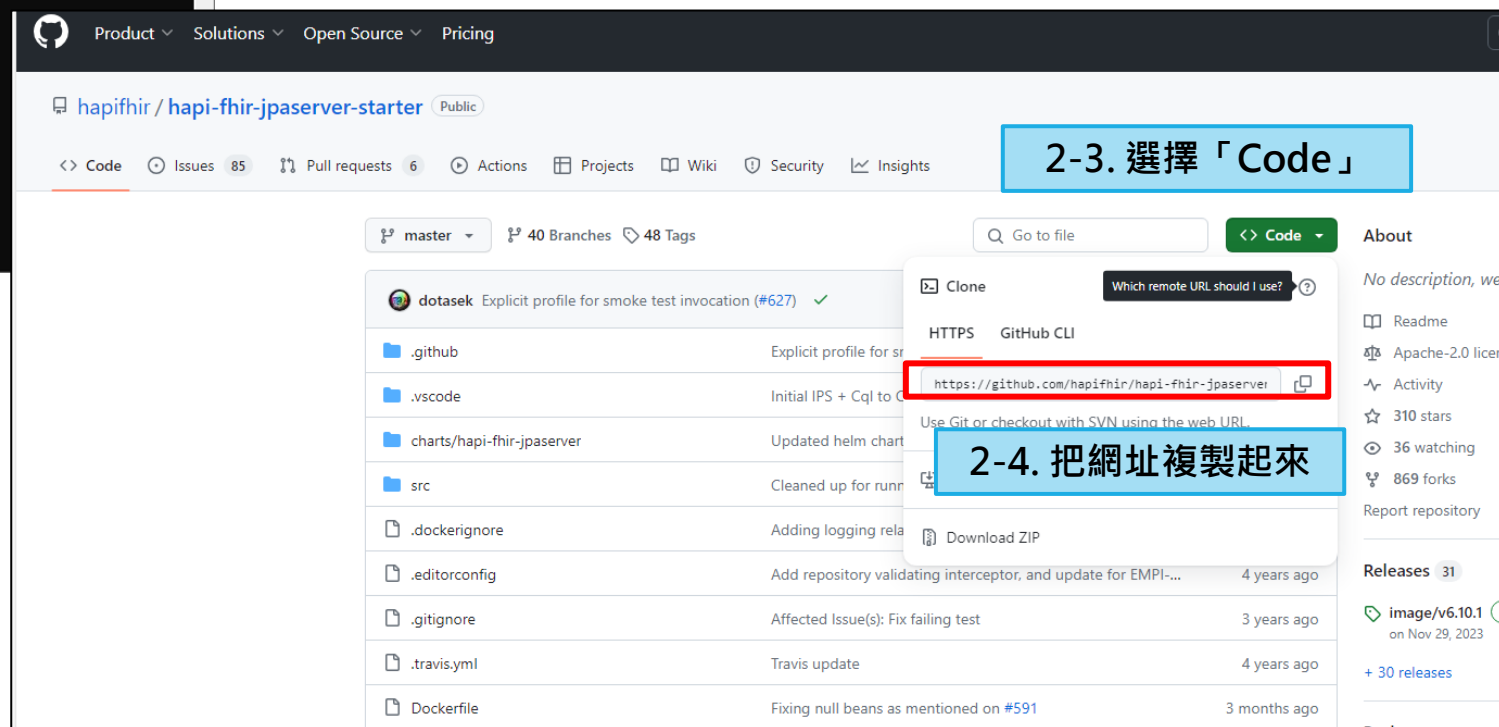
2-1. 連線至Linux電腦(或虛擬機)

2. 安裝HAPI Server

```
ubuntu@vm1704954932033-4320973-iaas: ~  
Expanded Security Maintenance for Applications is not enabled.  
0 updates can be applied immediately.  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The list of available updates is more than a week old.  
To check for new updates run: sudo apt update  
  
Last login: Thu Jan 11 14:59:00 2024 from 120.97.100.11  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
  
ubuntu@vm1704954932033-4320973-iaas:~$ sudo snap install docker  
Download snap "core22" (1033) from channel "stable"
```

2-1. 輸入 **sudo snap install docker**

2-2. 在搜尋引擎中搜尋「hapi fhir starter」
或網址「<https://github.com/hapifhir/hapi-fhir-jpaserver-starter>」



2. 安裝HAPI Server

```
ubuntu@vm1704954932033-4320973-iaas:~$ git clone https://github.com/hapifhir/hapi-fhir-jpaserver-starter.git
```

```
Cloning into 'hapi-fhir-jpaserver-starter'...
remote: Enumerating objects: 7598, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (9/9), done.
remote: Total 7598 (delta 2), reused 7 (delta 2), pack-reused 7587
Receiving objects: 100% (7598/7598), 3.22 MiB | 9.58 MiB/s, done.
Resolving deltas: 100% (2870/2870), done.
```

```
ubuntu@vm1704954932033-4320973-iaas:~$ ls
hapi-fhir-jpaserver-starter  persistence.json
```

```
ubuntu@vm1704954932033-4320973-iaas:~$ cd hapi-fhir-jpaserver-starter/
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$
```

2-5. 輸入 `git clone https://github.com/hapifhir/hapi-fhir-jpaserver-starter.git` (剛剛的網址) 並且可以輸入 `ls` 確認是否有下載成功

2-6. `cd hapi-fhir-jpaserver-starter/` 進入資料夾當中

```
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ ls
build-docker-image.bat  catalina.properties  docker-build.bat  Dockerfile  pom.xml
build-docker-image.sh   charts                docker-compose.yml  LICENSE     README.
```

```
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ rm docker-compose.yml
```

```
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ ls
build-docker-image.bat  catalina.properties  docker-build.bat  LICENSE  README.md  src
build-docker-image.sh   charts                Dockerfile         pom.xml  server.xml
```

```
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ nano docker-compose.yml
```

2-7. `rm docker-compose.yml` 先把原本的檔案刪除

2-8. `nano docker-compose.yml` 新增並編輯檔案

2. 安裝HAPI Server

```
ubuntu@vm1704954932033-4320973-iaas: ~/hapi-fhir-jpaserver-starter
GNU nano 6.2 docker-compose.yml *
services:
  fhir:
    container_name: fhir
    image: "hapiproject/hapi:latest"
    ports:
      - "8080:8080"
    configs:
      - source: hapi
        target: /app/config/application.yaml
    depends_on:
      - db
db:
  image: postgres
  restart: always
  environment:
    POSTGRES_PASSWORD: admin
    POSTGRES_USER: admin
    POSTGRES_DB: hapi
  volumes:
    - ./hapi.postgress.data

configs:
  hapi:
    file: ./hapi.application.yaml

^C Help      ^O Write Out
^X Exit      ^R Read File

2-10. 要關閉，要先ctrl+X

Save modified buffer?
^Y Yes      ^C Cancel

2-11. Y

File Name to Write: docker-compose.yml
^G Help      ^M-D DOS Format      ^M-A Append
^C Cancel    ^M-M Mac Format      ^M-P Prepend
```

2-12. Enter

2-9. 把以下全部貼上在檔案中
version: '3.7'

services:

fhir:

container_name: fhir

image: "hapiproject/hapi:latest"

ports:

- "8080:8080"

configs:

- source: hapi

target: /app/config/application.yaml

depends_on:

- db

db:

image: postgres

restart: always

environment:

POSTGRES_PASSWORD: admin

POSTGRES_USER: admin

POSTGRES_DB: hapi

volumes:

- ./hapi.postgress.data:/var/lib/postgresql/data

configs:

hapi:

file: ./hapi.application.yaml

2. 安裝HAPI Server

```
build-docker-image.sh charts docker-compose.yml LICENSE README.md src
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ ls
build-docker-image.bat catalina.properties docker-build.bat Dockerfile pom.xml server.xml
build-docker-image.sh charts docker-compose.yml LICENSE README.md src
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ nano ^C
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ ^C
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ nano hapi.application.yaml
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$
```

2-13. nano hapi.application.yaml 編輯檔案

```
ubuntu@vm1704954932033-4320973-iaas: ~/hapi-fhir-jpaserver-starter
GNU nano 6.2 hapi.application.yaml *
spring:
  datasource:
    url: 'jdbc:postgresql://db:5432/hapi'
    username: admin
    password: admin
    driverClassName: org.postgresql.Driver
  jpa:
    properties:
      hibernate.dialect: ca.uhn.fhir.jpa.model.dialect.HapiFhirPostgres94Dialect
      hibernate.search.enabled: false
^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location M-U Undo M-A Set Mark
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line M-E Redo M-6 Copy
```

2-14. 把以下全部貼上在檔案中，關閉方法如2-10.~2-12.

```
spring:
  datasource:
    url: 'jdbc:postgresql://db:5432/hapi'
    username: admin
    password: admin
    driverClassName: org.postgresql.Driver
  jpa:
    properties:
      hibernate.dialect:
ca.uhn.fhir.jpa.model.dialect.HapiFhirPostgres94Dial
ect
      hibernate.search.enabled: false
```

2. 安裝HAPI Server

```
build-docker-image.bat catalina.properties docker-build.bat Dockerfile pom.xml server.xml  
build-docker-image.sh charts docker-compose.yml LICENSE README.md src  
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ nano ^C  
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ ^C  
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ nano hapi-application.yml  
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$ sudo docker-compose up -d  
[+] Running 0/1  
? fhir Pulling
```

2-15. `sudo docker-compose up -d`

```
ubuntu@vm1704954932033-4320973-iaas: ~/hapi-fhir-jpaserver-starter  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 2/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
[+] Running 3/3  
✓ Network hapi-fhir-jpaserver-starter_default Created 0.1s  
✓ Container hapi-fhir-jpaserver-starter-db-1 Started 6.3s  
✓ Container fhir Started 4.9s  
ubuntu@vm1704954932033-4320973-iaas:~/hapi-fhir-jpaserver-starter$
```

3. 檢查是否安裝成功

可以去瀏覽器查看是否架設成功
<http://203.145.214.76:8080/>

Home Server: Local Tester Source Code About This Server

Options

Encoding (default) XML JSON

Pretty (default) On Off

Summary (none) true text data count

Server

Server Home/Actions

Resources

- Account
- ActivityDefinition
- AdverseEvent
- AllergyIntolerance
- Appointment
- AppointmentResponse
- AuditEvent
- Basic
- Binary
- BiologicallyDerivedProduct
- BodyStructure
- Bundle
- CapabilityStatement
- CarePlan
- CareTeam
- CatalogEntry
- ChargeItem

COMPANY NAME

YOUR SAMPLE TEXT HERE

This server provides a complete implementation of the FHIR Specification using a 100% open source software stack.

This server is built from a number of modules of the [HAPI FHIR](#) project, which is a 100% open-source (Apache 2.0 Licensed) Java based implementation of the FHIR specification.

Server	HAPI FHIR R4 Server
Software	HAPI FHIR Server - 6.10.0
FHIR Base	http://localhost:8080/fhir

Server Actions

Retrieve the server's **conformance** statement.

[Conformance](#)

Retrieve the update **history** across all resource types on the server.

[History](#) Since Limit # (opt)

Post a bundle containing multiple resources to the server and store all resources within a single atomic transaction.

[Transaction](#) Bundle*