

Microbiota amplicOn
CHaracterization Implement

MOCHI

User Guide

TABLE OF CONTENTS

Chapter 1: Installation of local service

MacOS.....	3
(A) Install Docker	3
(B) Start MOCHI service.....	5
Window 10.....	7
(A) Install Docker	7
(B) Start MOCHI service.....	9
Linux	11
(A) Install Docker	11
(B) Start MOCHI service	11
Additional Information	12

CHAPTER 1: INSTALLATION OF LOCAL SERVICE

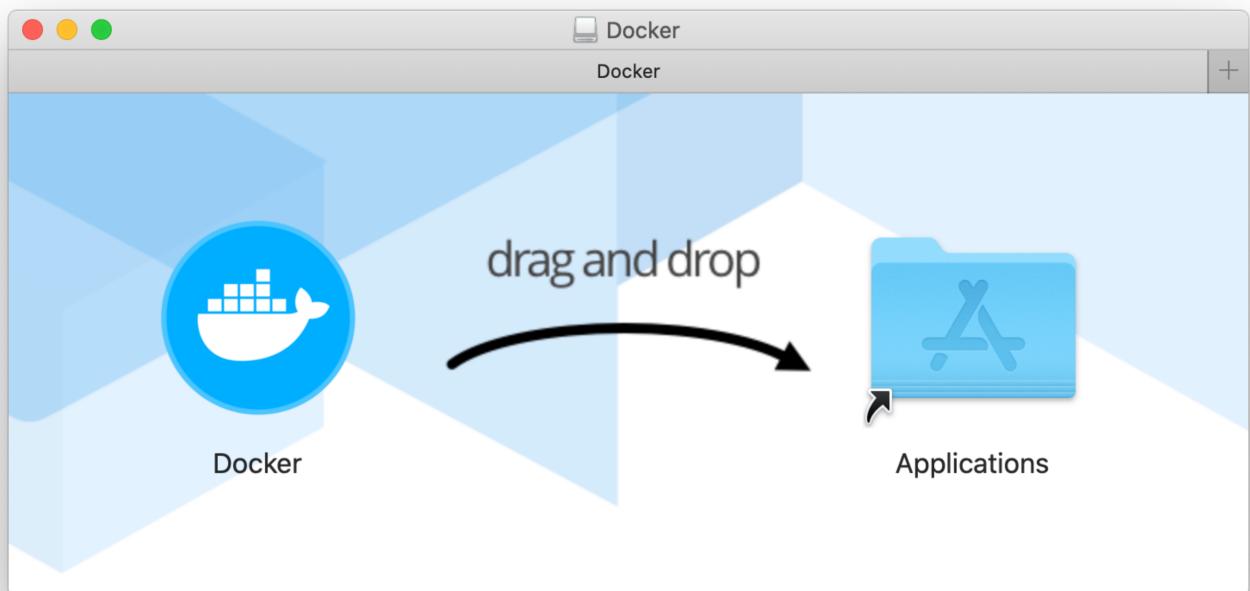
MacOS

(A) Install Docker

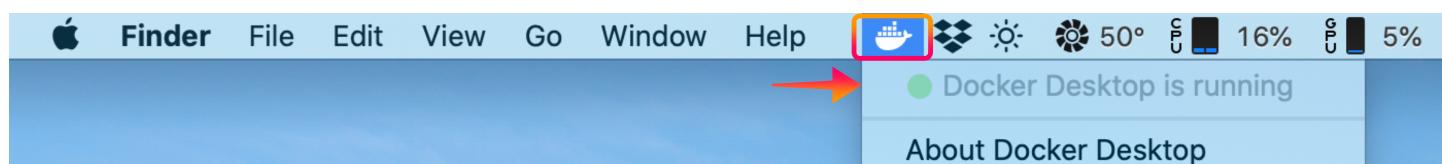
1. Download [Docker Desktop](#).



2. Open "**Docker.dmg**" file. Drag **Docker Desktop** app to the Applications folder.



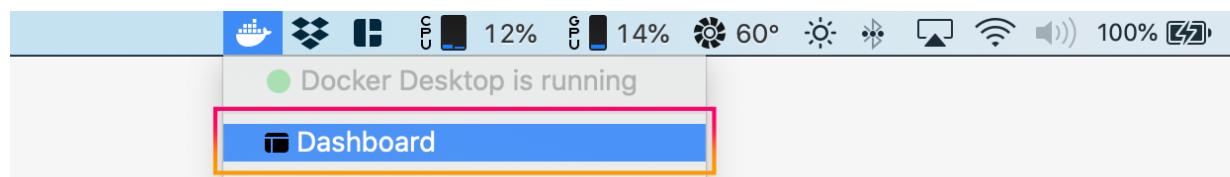
3. Start the Docker service by double clicking the Docker app. Wait for a few seconds to load, the docker icon should appear in the status bar.



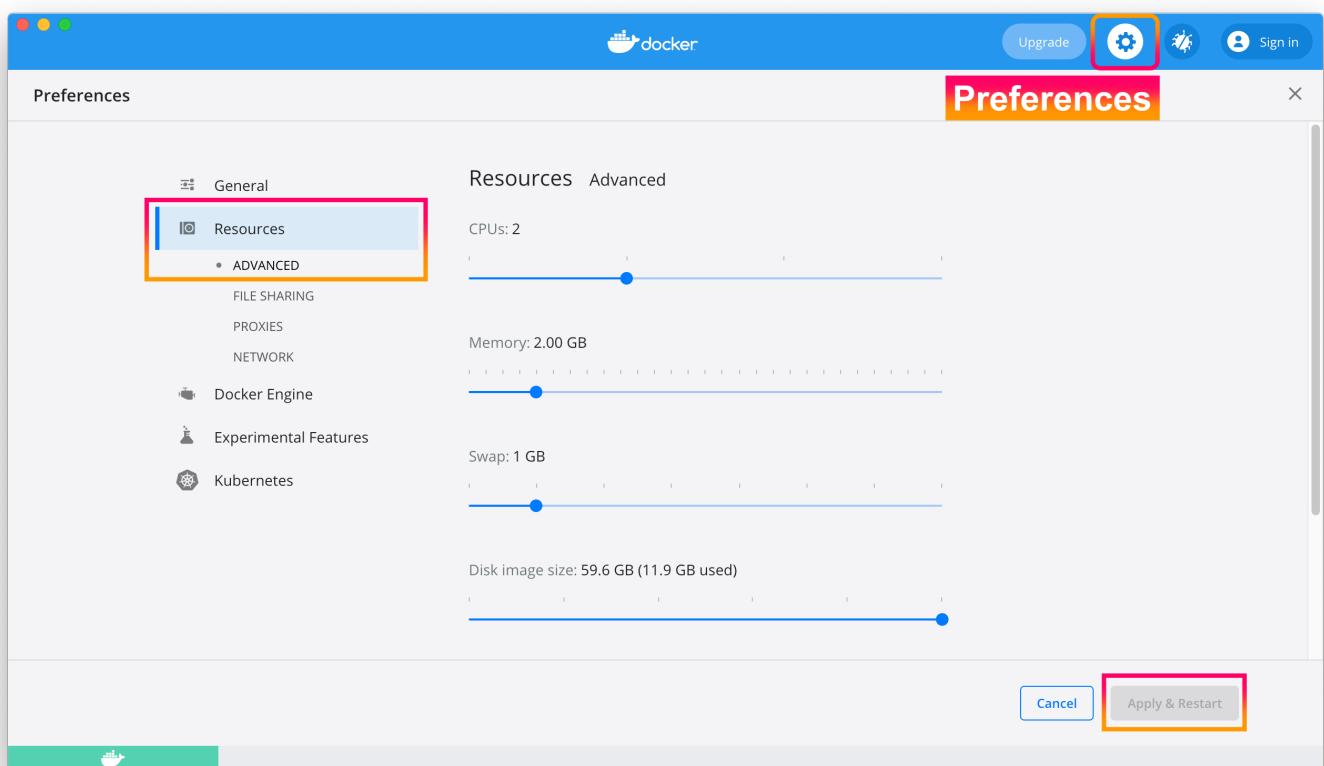
4. (Optional) In MacOS, the computational resources are preset in docker app. To optimize the efficiency of analysis, user can adjust the settings with following instruction. We recommend settings **above 4 CPUs** and **8-16 GB memory** (by default, MOCHI only use maximum 16GB memory).

⚠ Assign all of the resources to Docker may cause your system to delay or crash.

4.1. Open the Docker dashboard from the drop-down menu in the status bar.



4.2. Go to **Preference / Resources / Advanced**. Adjust the resources with rolling bar. Press "Apply & Restart". Waiting for the Docker to restart.



(B) Start MOCHI service

1. Download “**docker-compose.yml**” from [NCTU website](#), and place it under a folder named “**MOCHI**”.



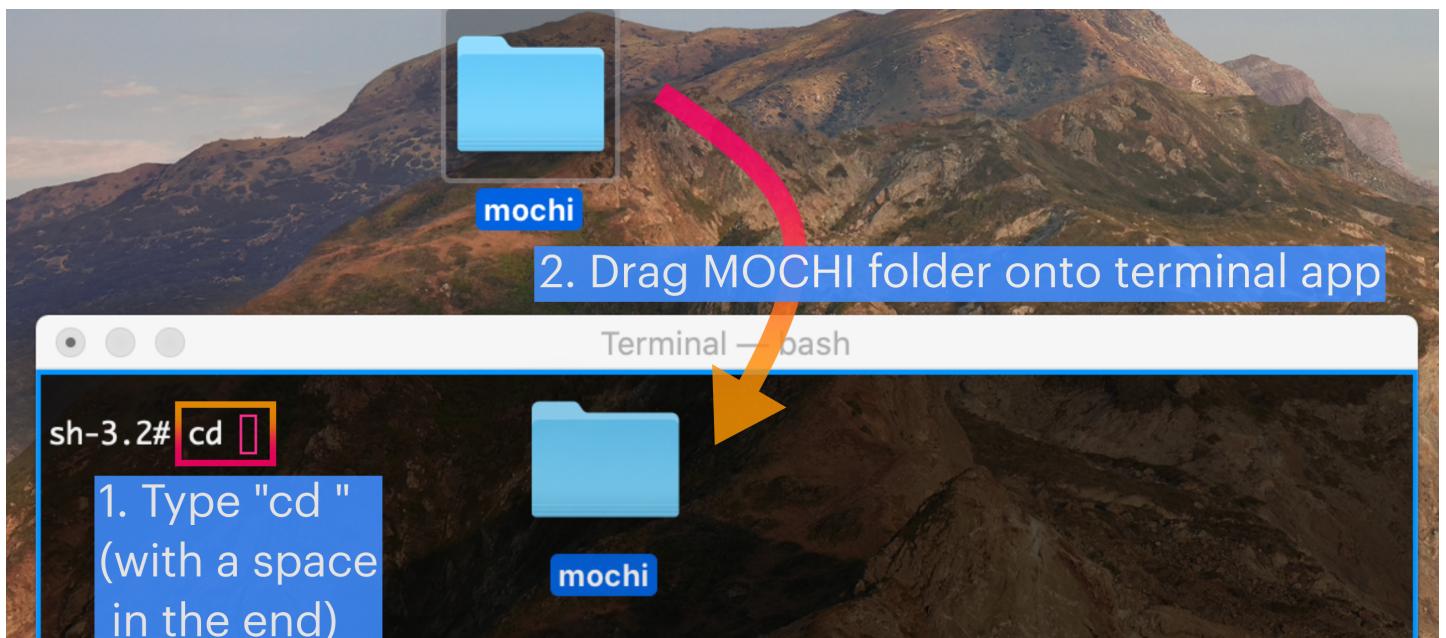
2. Open **Terminal** app. (Press “Command + Space” to open spotlight and type “terminal”.)



3. Run `cd /path/to/MOCHI` to navigate under MOCHI directory.

```
sh-3.2# cd /Users/Mac/Desktop/mochi
```

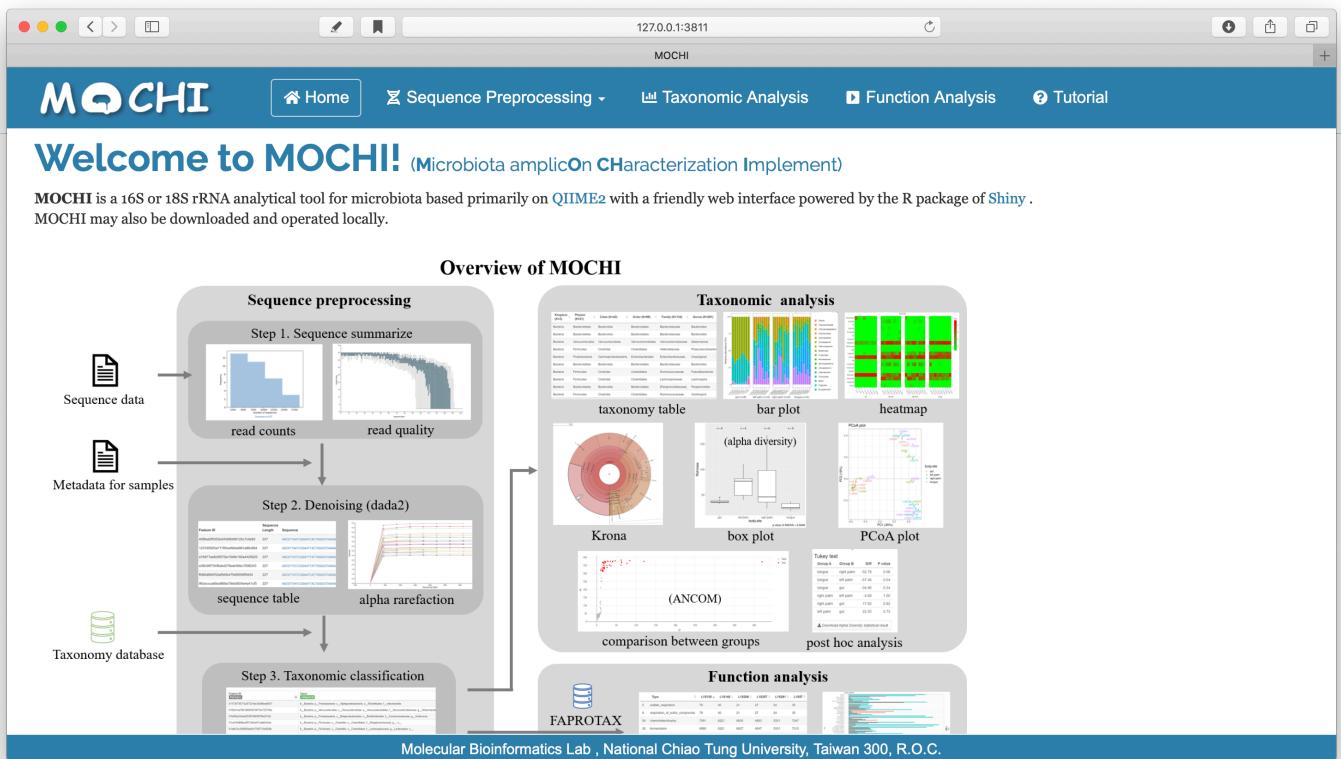
Hint: If you do not know the folder path of MOCHI, type `cd` with a space, and drag MOCHI folder onto the terminal window. The folder path should appear on the terminal automatically.



4. Run `docker-compose up -d` to download and start the MOCHI image from Docker Hub.
 (The download process will only run during the first setup. The size of MOCHI image is around 8 GB, and the running time depends on the download speed.)

```
[powang@MacBook-Pro--PoWang mochi % docker-compose up -d
WARNING: Some services (mochi_server) use the 'deploy' key, which will be ignored. Compose does not support 'deploy' configuration - use `docker stack deploy` to deploy to a swarm.
Creating network "mochi_default" with the default driver
Pulling mochi_server (dockerjjz/mochi_local:...)...
latest: Pulling from dockerjjz/mochi_local
f15005b0235f: Pull complete
1901fd813023: Pull complete
a92940affedf: Pull complete
dbebda29cb22: Pull complete
3c63b26b92fd: Pull complete
e0c15c0b4e0b: Pull complete
Digest: sha256:1501a145eb826f9f799239964eb064170fdc5be8abcd7b04fd9a61c888b9dee
Status: Downloaded newer image for dockerjjz/mochi_local:latest
Creating mochi_server ... done]
```

5. Open the browser, and type **127.0.0.1:3811** on the address bar. A MOCHI interactive webpage is now available to begin your analysis.



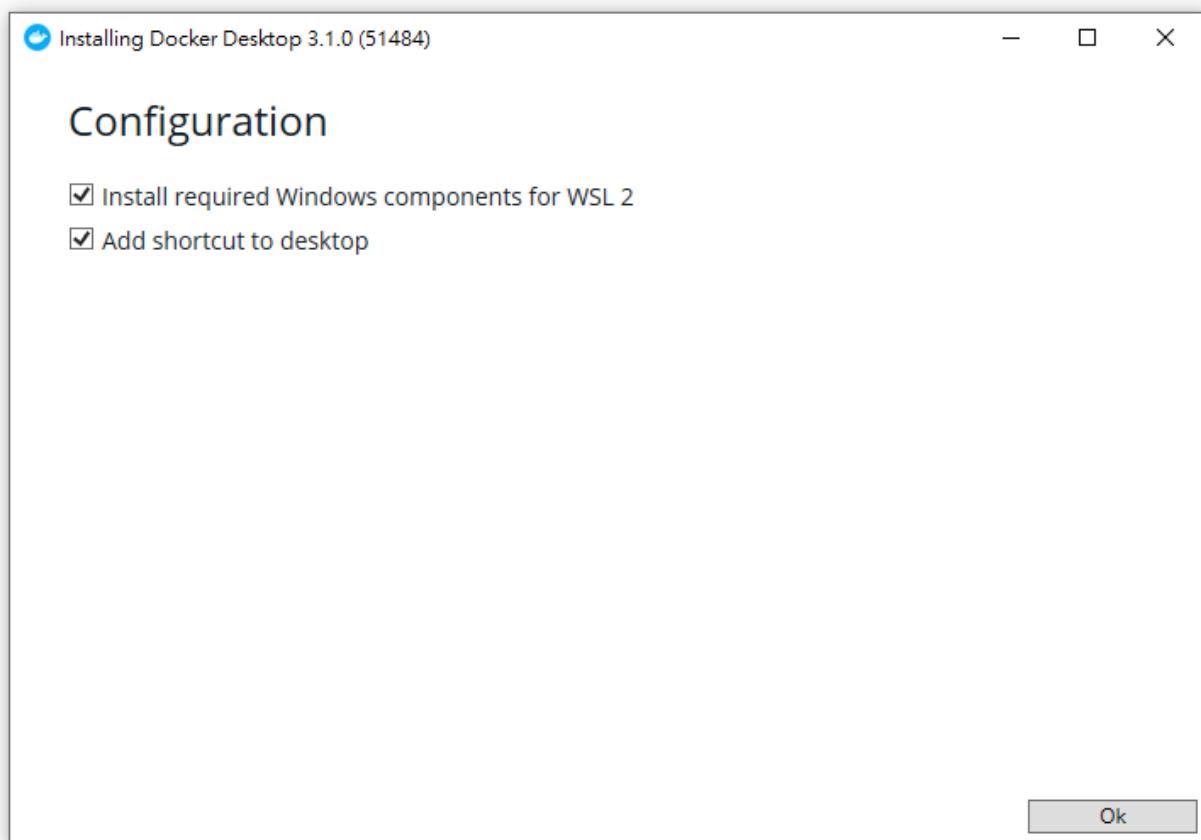
Window 10

(A) Install Docker

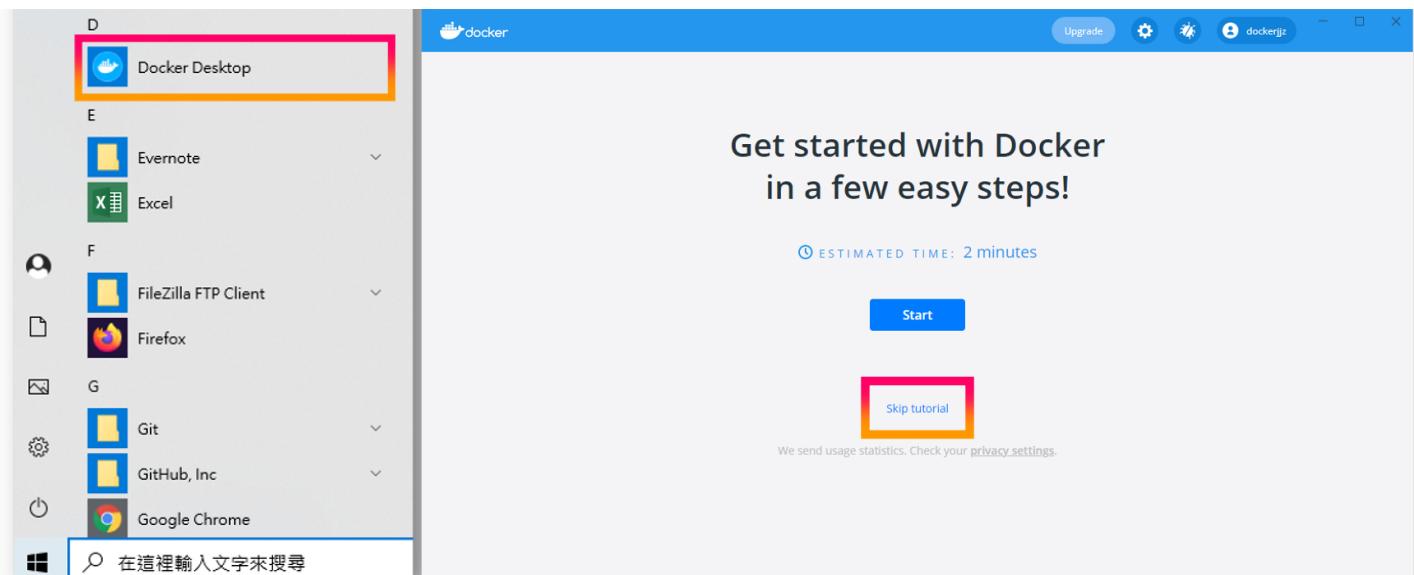
1. Download [Docker Desktop](#).



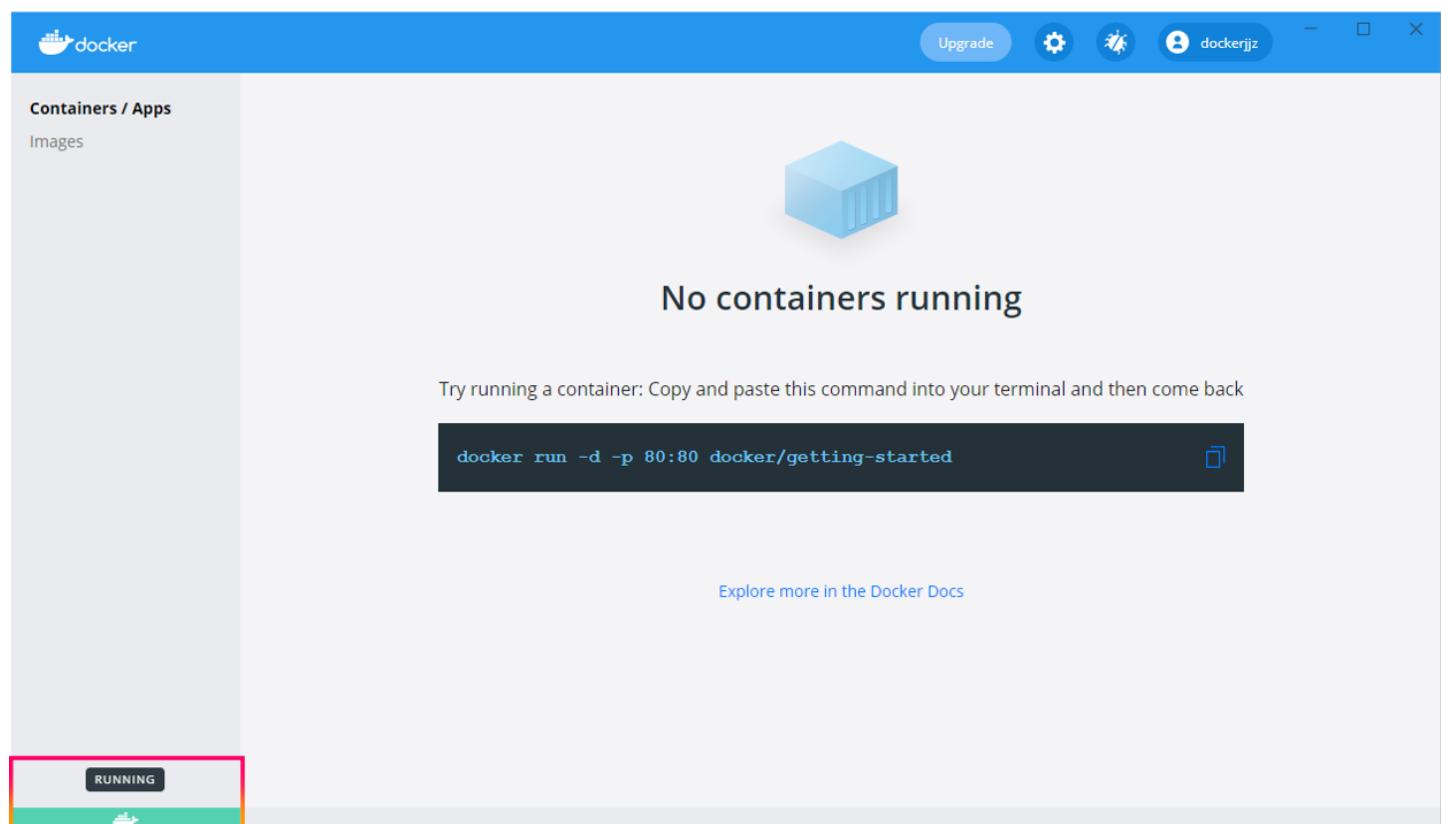
2. Open "**Docker Desktop Installer.exe**" and follow the instruction to setup. (You may be asked whether or not to install WSL2 engine. By default, it is installed along with the docker package to speed up the performance but not required for running MOCHI service. Please leave it to default setting if you don't know what to choose.)



3. Start docker service by **clicking the Docker icon**. Press “Skip tutorial”.



4. If the docker service has been successfully turned on, a green indicator will appear on the bottom-left in the docker app.

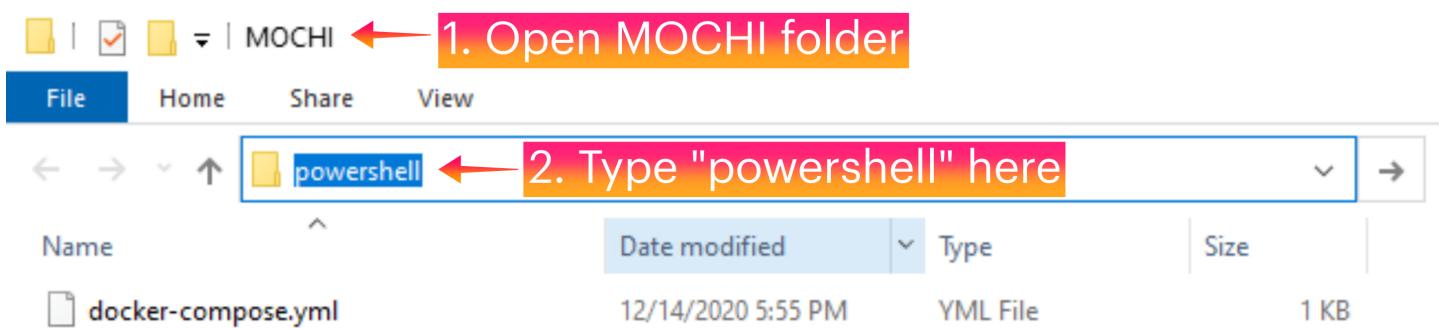


(B) Start MOCHI service

1. Download “**docker-compose.yml**” from [NCTU website](#), and place it under a folder named “**MOCHI**”.



2. Open **MOCHI folder**, type “**powershell**” in the address bar, and press enter. This will open a command-line shell under MOCHI directory.



3. In Powershell, run **docker-compose up -d** to download and start the MOCHI image from Docker Hub. (The download process will only run at the first setup. The size of MOCHI image is around 8 GB, and the running time depends on the download speed.)

```
PS C:\Users\dodolab\Desktop\MOCHI> docker-compose up -d
Creating network "mochi_default" with the default driver
Pulling mochi_server (dockerjjz/mochi_local_version:...)...
latest: Pulling from dockerjjz/mochi_local_version
f15005b0235f: Pull complete
1901fd813023: Pull complete
a92940affedf: Pull complete
dbebda29cb22: Pull complete
3c63b26b92fd: Pull complete
e4191a297544: Pull complete
Digest: sha256:794909d921dfe9cc55edba44a3fe66701e43845c5c5578bf1e194064e071cbe
Status: Downloaded newer image for dockerjjz/mochi_local_version:latest
Creating mochi_server_version ... done
```

4. Open the browser, and type **127.0.0.1:3811** on the address bar. A MOCHI interactive webpage is now available to begin your analysis.

The screenshot shows the MOCHI web application interface. At the top, there's a header with the MOCHI logo, a search bar containing "127.0.0.1:3811", and navigation links for Home, Sequence Preprocessing, Taxonomic Analysis, Function Analysis, and Tutorial. Below the header, a large blue banner says "Welcome to MOCHI! (Microbiota amplicOn CHaracterization Implement)". It describes MOCHI as a 16S or 18S rRNA analytical tool based on QIIME2, with a friendly web interface powered by the R package Shiny. It also mentions that MOCHI can be downloaded and operated locally.

Overview of MOCHI

The main content area is divided into three main sections:

- Sequence preprocessing:** This section shows the flow from "Sequence data" (read counts and read quality plots) through "Metadata for samples" (sequence table and alpha rarefaction plot) to "Taxonomy database" (FAPROTAX).
- Taxonomic analysis:** This section displays various taxonomic metrics and visualizations, including a taxonomy table, bar plot, heatmap, Krone plot, box plot, PCoA plot, (ANCOM), comparison between groups, and post hoc analysis.
- Function analysis:** This section shows a table of results and a corresponding bar chart.

At the bottom of the page, it says "Molecular Bioinformatics Lab, National Chiao Tung University, Taiwan 300, R.O.C."

Linux

(A) Install Docker

- ▶ Please follow the [official guidance](#) to setup and start docker engine on Ubuntu.

(B) Start MOCHI service

1. Download “**docker-compose.yml**” from [NCTU website](#), and place it under a folder named “**MOCHI**”.



2. Navigate to the MOCHI directory.

```
$ cd /path/to/place/MOCHI
```

3. Download and start the MOCHI image from Docker Hub. (The download process will only run during the first setup. The size of MOCHI image is around 8 GB, and the running time depends on the download speed.)

```
$ docker-compose up -d
```

4. Open the browser, and type **127.0.0.1:3811** on the address bar. A MOCHI interactive webpage is now available to begin your analysis.

Additional Information

* Stop, Restore or Remove MOCHI service

- ▶ To temporarily pause MOCHI service, please run **docker-compose stop**. This will create a paused MOCHI container, and can be restored with command **docker-compose start**.

NOTE: Please be cautious that interrupting the MOCHI service when an analysis is running will not save the running process. The MOCHI container is automatically paused once you restart your computer or docker service.

- ▶ To permanently close MOCHI service, please run **docker-compose down**. This will remove all the data generated by MOCHI. Please save the results in advance. If you wish to open the service again, please run **docker-compose up -d**.
- ▶ To uninstall MOCHI image from your computer, please run **docker images**:

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
dockerjjz/mochi_local	latest	441fe987dce8	2 weeks ago	9.05GB

and look for “IMAGE ID” of repository named “**dockerjjz/mochi_local**”. Then, run **docker rmi [IMAGE ID]** (e.g., **docker rmi 441**; type partial or complete image ID are both acceptable.)

* Remember to navigate under the MOCHI folder (where “**docker-compose.yml**” is located) before starting, stopping or restoring MOCHI service.

- ▶ For MacOS, Please see [MacOS / step B-3](#).
- ▶ For Window 10, Please see [Window 10 / step B-2](#).
- ▶ For Linux, Please see [Linux / step B-2](#).

* After installing MOCHI, two folders are created under the MOCHI folder:

1. **seqs_folder**

- ▶ This folder is used to store user’s sequence data which will be loaded in “Sequence preprocessing - Sequence summary”.
- ▶ The file type of sequence data need to be **fastq.gz** or **fq.gz**.
- ▶ The filename of sequence data need to satisfy Casava 1.8 demultiplexed format or the following example format **[SampleID]_[direction of reads]**:
 - ✓ Forward read: LS105_R1 or LS105_1
 - ✓ Reverse read: LS105_R2 or LS105_2

2. **taxa_database**

- ▶ This folder is used to store taxonomy database, such as Greengenes, Silva and PR2. Please see "Sequence preprocessing - Taxonomy classification".

- * If you wish to change the default path to the sequence data and the taxonomic database, open the "**docker-compose.yml**" file, replace the texts "**./seqs_folder**" and "**./taxa_database**" with new paths to the sequence data and the taxonomic database, respectively. Relative path is allowed.
- * The default maximum memory used by MOCHI is 16GB. To increase, please modified the resource limit in the "**docker-compose.yml**" file.

```
version: '3.7'

services:

  mochi_server:
    # build: .
    image: dockerjjz/mochi_local
    ports:
      - "3811:3838"
      - "8011:80"
    volumes:
      - ./seqs_folder:/home/imuser/raw_data/:rw
      - ./taxa_database/greengenes:/home/imuser/taxa_database/greengenes:rw
      - ./taxa_database/silva:/home/imuser/taxa_database/silva:rw
      - ./taxa_database/PR2/18S/seqs:/home/imuser/taxa_database/PR2/18S/seqs:rw
      - ./taxa_database/PR2/18S/taxonomy:/home/imuser/taxa_database/PR2/18S/taxonomy:rw
    container_name: mochi_server

    deploy:
      resources:
        limits:
          #cpus: '8'
          memory: 16G

      environment:
        - shiny_port=3811
        - nginx_port=8011
```