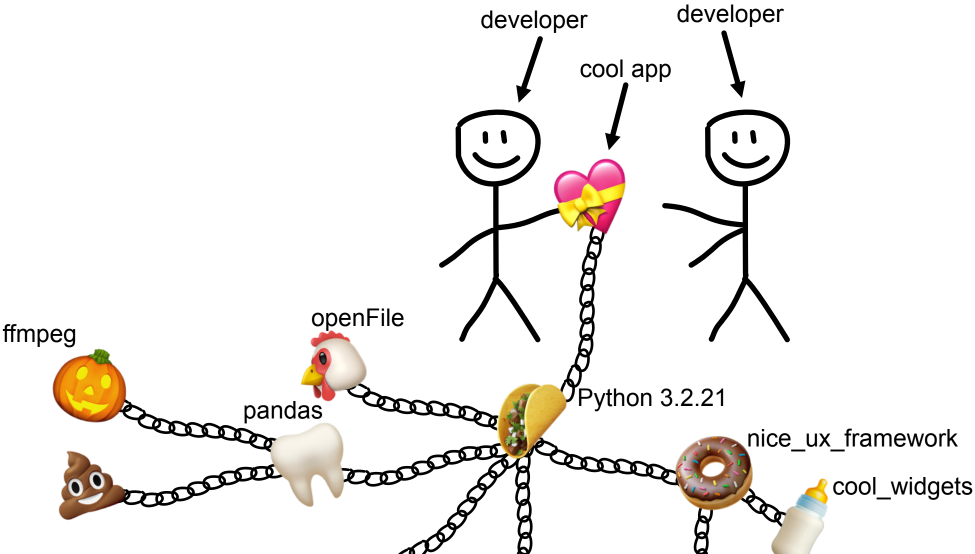
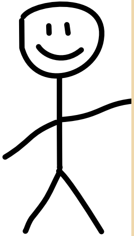
DEVELOPMENT BE LIKE



DEVELOPMENT BE LIKE



Build instructions

Install these modules:

```
> genjĳa         grrr
  skibidi        libgrass
  libyomama      libuuu
  arrr
```

To make it easy, just install

```
> pip
```

then do

```
> pip install -r requirements.txt
```

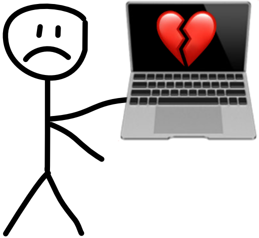
And to avoid messing up your pc, make sure to use

```
> python venv
```

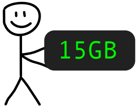
Oh and be sure to have

```
> python 3.2.21
```

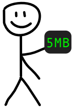
Your dev friend



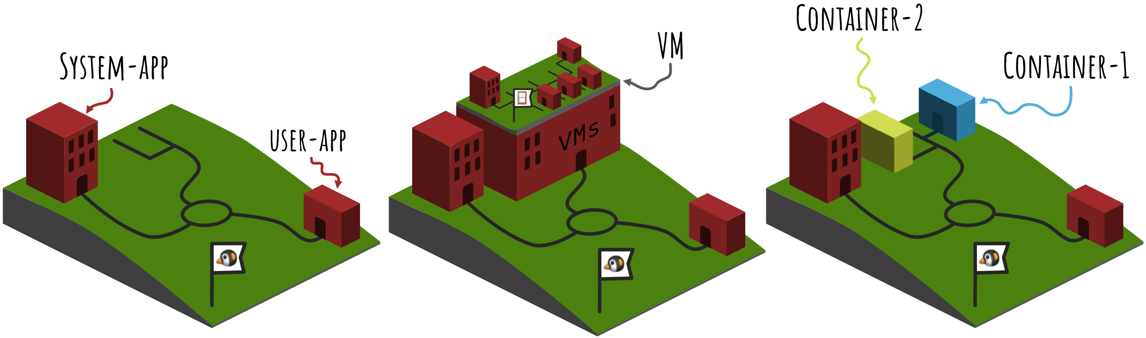
SOLUZIONI



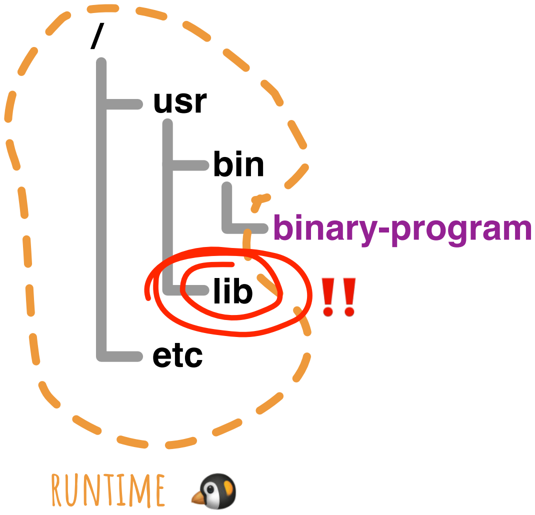
VMs	❌PESANTI	
PACKAGE MANAGERS	🤷‍♂️DIPENDE DAL LINGUAGGIO	
STATIC LINKING	🙅‍♂️DIFFICILMENTE PRATICABILE	
CONTAINERS	🙌 LA SCELTA MIGLIORE	REPLICABILI LEGGERI



CONTAINERS vs VMs

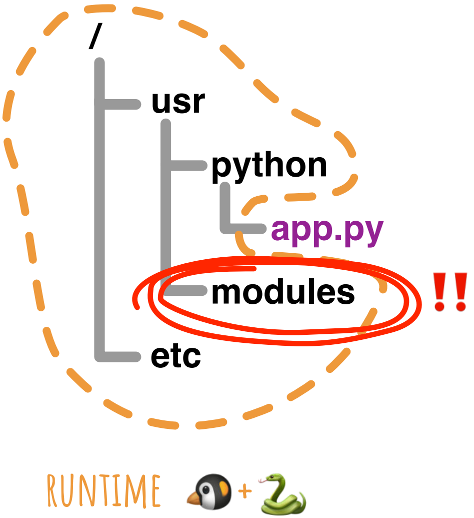


CONTAINERS



/
usr
bin
binary-program
lib
!!
etc

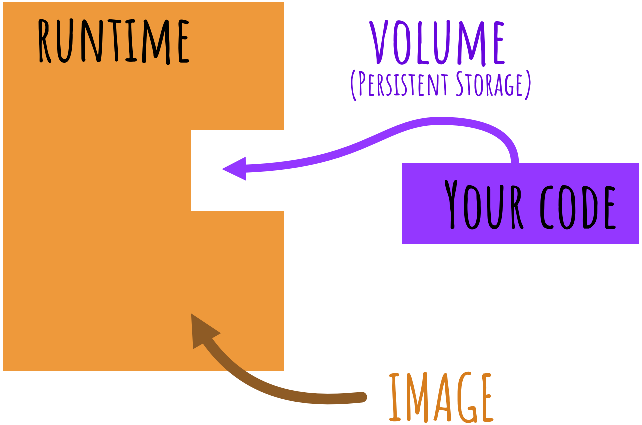
RUNTIME 🐧



/
usr
python
app.py
modules
!!
etc

RUNTIME 🐧 + 🐍

CONTAINERS



RUNTIME

VOLUME
(PERSISTENT STORAGE)

YOUR CODE

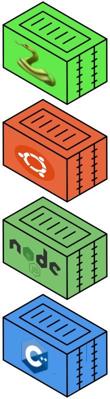
IMAGE

Docker

compose.yml

Dockerfile

Container hub

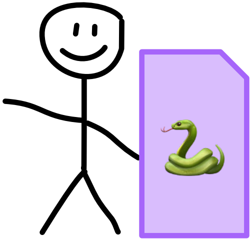


python

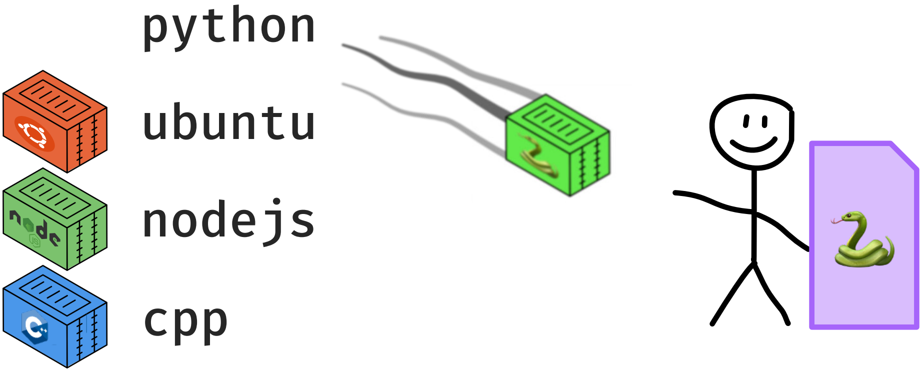
ubuntu

nodejs

cpp



Container hub



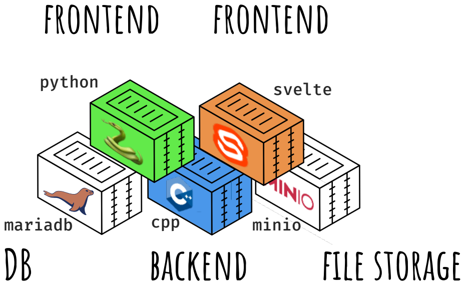
python

ubuntu

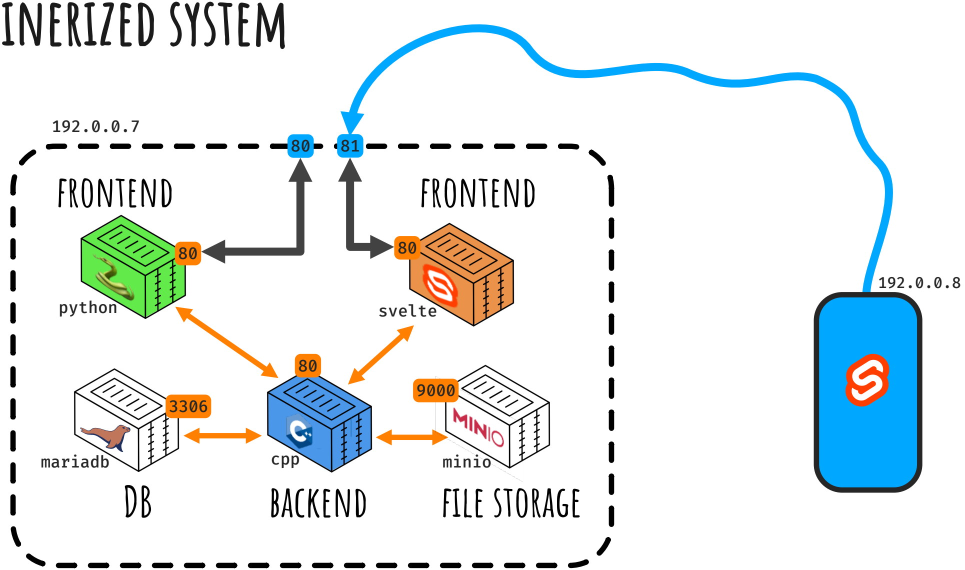
nodejs

cpp

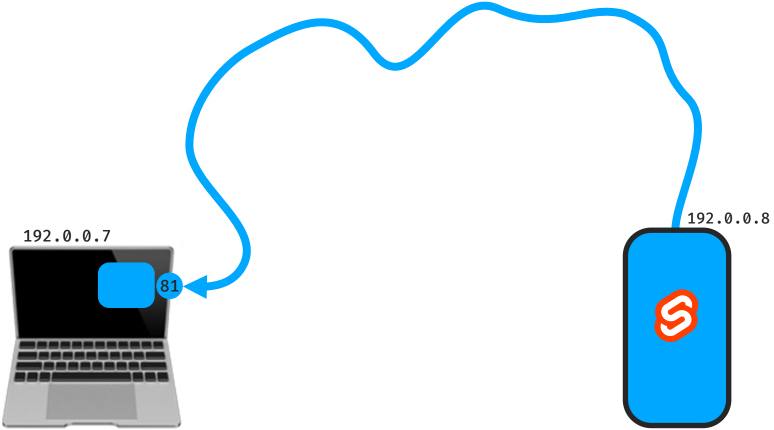
Containerized system



CONTAINERIZED SYSTEM

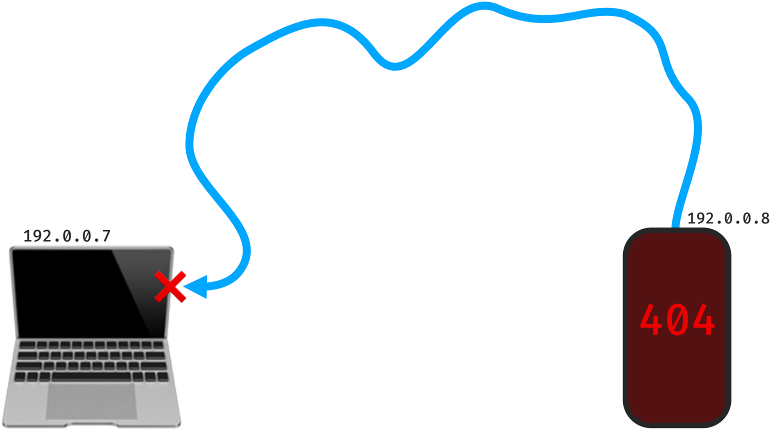


Containerized system



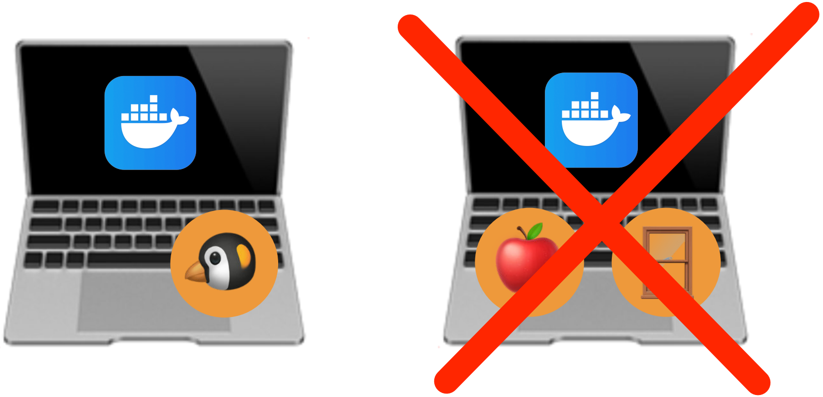
docker compose up

Containerized system



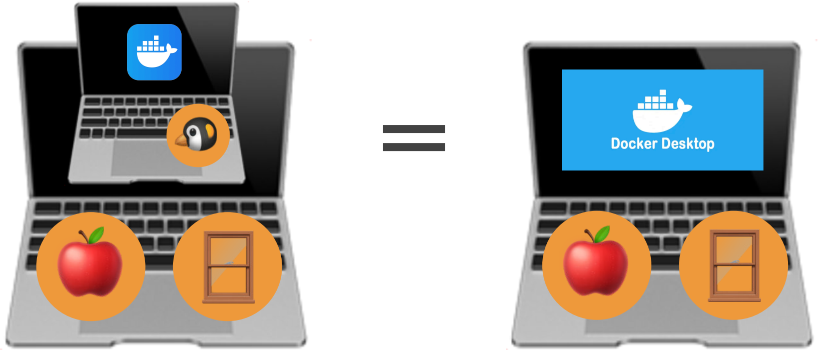
```
docker compose down
```

HANDS ON



"CONTAINERS ARE A LINUX TECHNOLOGY"

HANDS ON



USEFUL KNOWLEDGE

Dockerfile

```
FROM

RUN

COPY

ENTRYPOINT
```

compose.yml

```
services:
 container:
  image: image
  container_name: container
  ports:
   - hostport:contport
  volumes:
   - hostdir:contdir
  environment:
   - var=value
```

commands

```
docker compose up -d
docker compose down
docker attach container
docker exec -it container sh
```

USEFUL KNOWLEDGE - BUILDING IMAGES

Dockerfile

```
FROM image

RUN bashcommand

COPY hostfile contfile

ENTRYPOINT bashcommand
```

build.compose.yml

```
services
<name>:
  image: image
  build: .
```

```
docker compose -f build.compose.yml build
```

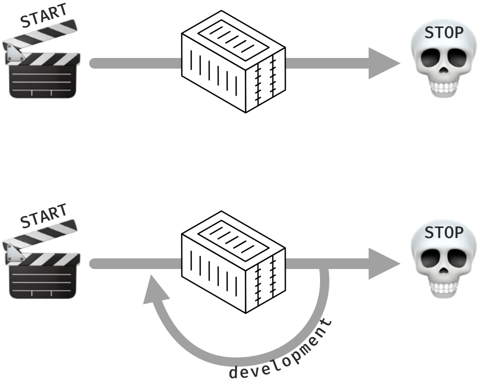
USEFUL KNOWLEDGE - RUNNING CONTAINERS

compose.yml

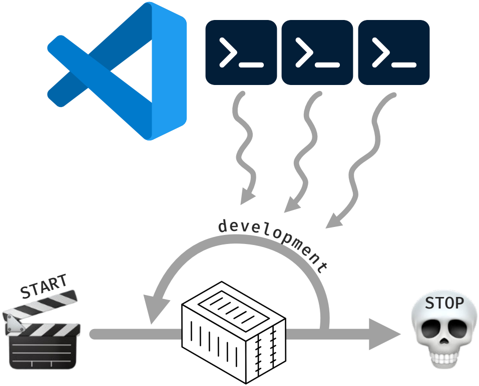
```
services:
 container:
  image: image
  container_name: container
  ports:
   - hostport:contport
  volumes:
   - hostdir:contdir
  environment:
   - var=value
```

```
docker compose up -d
```

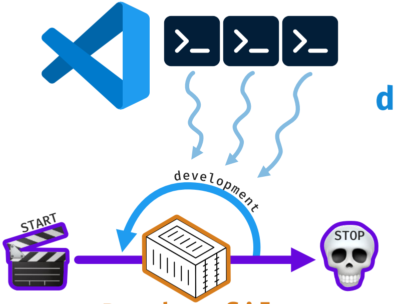
PROSEGUENDO...



VSCODE



COMPLESSIVAMENTE

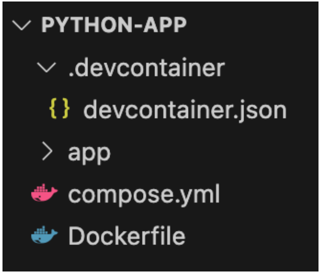


devcontainer.json

Dockerfile

compose.yml

NOW WE CODE!



```
PYTHON-APP
  .devcontainer
    devcontainer.json
  app
  compose.yml
  Dockerfile
```

run
debug
share
deploy