

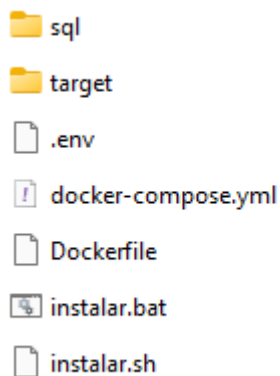
INDEX

INTRODUZIONE.....	4
PROCESSO DI INSTALLAZIONE.....	4
INSTALLAZIONE DELL'API, DEL DATABASE E DI phpMyAdmin.....	7
ERRORI.....	10

INTRODUZIONE

Questa guida descrive il processo necessario per poter utilizzare l'API Z8i. I file necessari per l'installazione sono allegati al presente manuale.

I file sopra menzionati sono i seguenti:

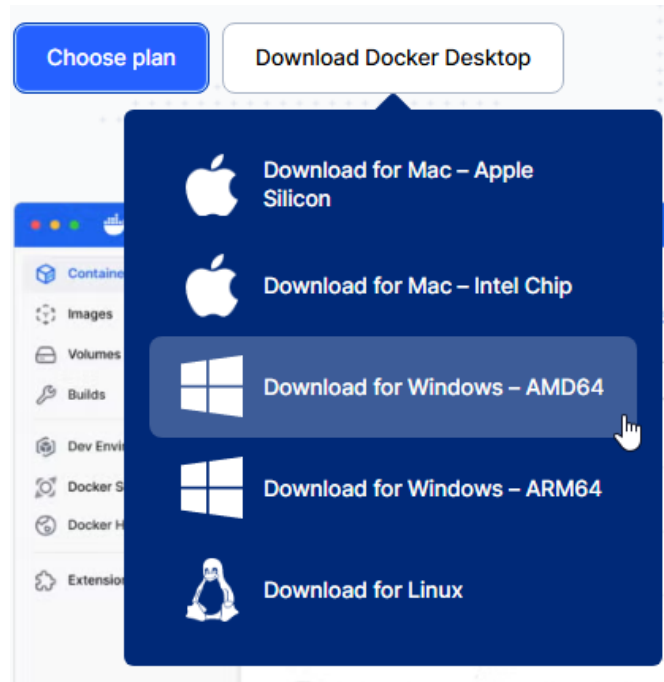


PROCESSO DI INSTALLAZIONE

Il primo passo consiste nell'installare Docker Desktop. È necessario utilizzare il seguente link per scaricare i file di installazione:

<https://www.docker.com/products/docker-desktop/>

Si prega di scegliere la versione di installazione appropriata per il proprio dispositivo; quella più comune è la versione AMD x86 per Windows.



Una volta scaricato il file indicato, proseguiremo con il processo di installazione fino al suo completamento.

Durante la prima esecuzione del software, verrà richiesto di accettare i termini del contratto di servizio.

Docker Subscription Service Agreement

By selecting **accept**, you agree to the [Subscription Service Agreement](#), the [Docker Data Processing Agreement](#), the [Data Privacy Policy](#) and the [Docker AI Supplemental Terms](#).

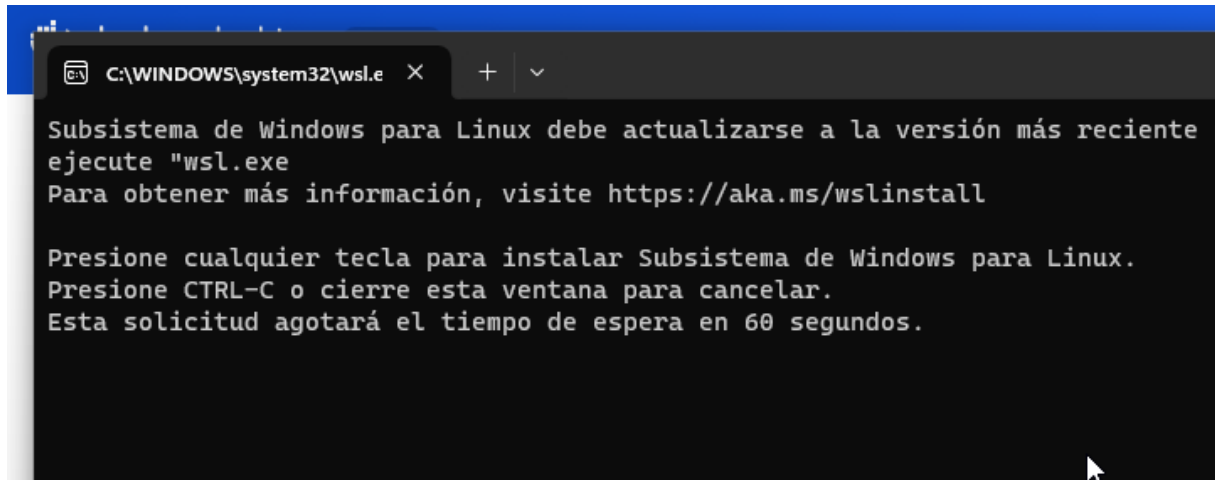
Commercial use of Docker Desktop at a company of more than 250 employees OR more than \$10 million in annual revenue requires a paid subscription (Pro, Team, or Business). [See subscription details](#)

[View Full Terms](#)

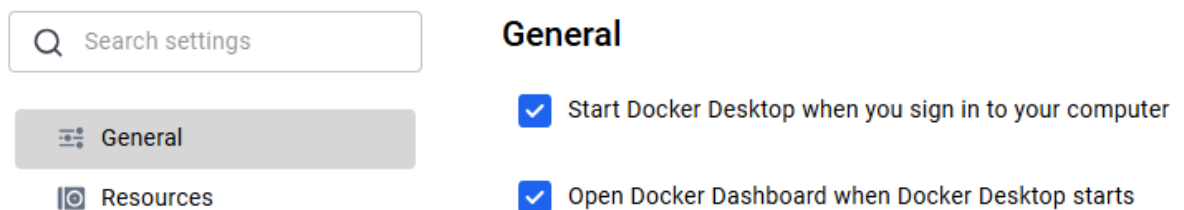
[Accept](#)

[Close](#)

Una volta accettati, l'applicazione richiederà di installare il Sottosistema Windows per Linux. Premere un tasto qualsiasi per continuare con l'installazione.



Una volta completata l'installazione, Docker Desktop si aprirà. Assicurarsi che sia in esecuzione all'avvio del terminale.



Assicurarsi che questa opzione sia selezionata in Configurazione → Generale.

INSTALLAZIONE DELL'API, DEL DATABASE E DI phpMyAdmin

Una volta completata la preparazione, lo strumento consentirà di installare l'API generando l'ambiente necessario al suo funzionamento.

Il passo successivo consiste nell'eseguire il file *instalar.bat*. Se si utilizza Linux, eseguire *instalar.sh*.

Docker genererà quindi un database secondo lo schema da noi fornito.

```
version: "3.8"

services:
  db:
    image: mysql:8.0
    container_name: open_z8_mysql
    profiles: ["localdb"]
    environment:
      MYSQL_ROOT_PASSWORD: Gram1231
      MYSQL_DATABASE: open_z8
      MYSQL_USER: apiuser
      MYSQL_PASSWORD: Gram1231
    volumes:
      - db_data:/var/lib/mysql
      - ./sql:/docker-entrypoint-initdb.d
    restart: unless-stopped
    ports:
      - "3306:3306"
  api:
    build: .
    container_name: open_z8_api
    env_file: .env
    depends_on:
      - db
    ports:
      - "8080:8080"
    restart: unless-stopped
  phpmyadmin:
    image: phpmyadmin/phpmyadmin:latest
    container_name: open_z8_phpmyadmin
    profiles: ["localdb"]
    depends_on:
      - db
    environment:
      PMA_HOST: db
      PMA_USER: root
      PMA_PASSWORD: Gram1231
      UPLOAD_LIMIT: 64M
    ports:
      - "8081:80"
    restart: unless-stopped

volumes:
  db_data:
```

Si tenga presente che il file *docker-compose.yml* può definire sia la porta del database, le relative credenziali, la porta su cui opera l'API, sia la porta per la connessione a phpMyAdmin.

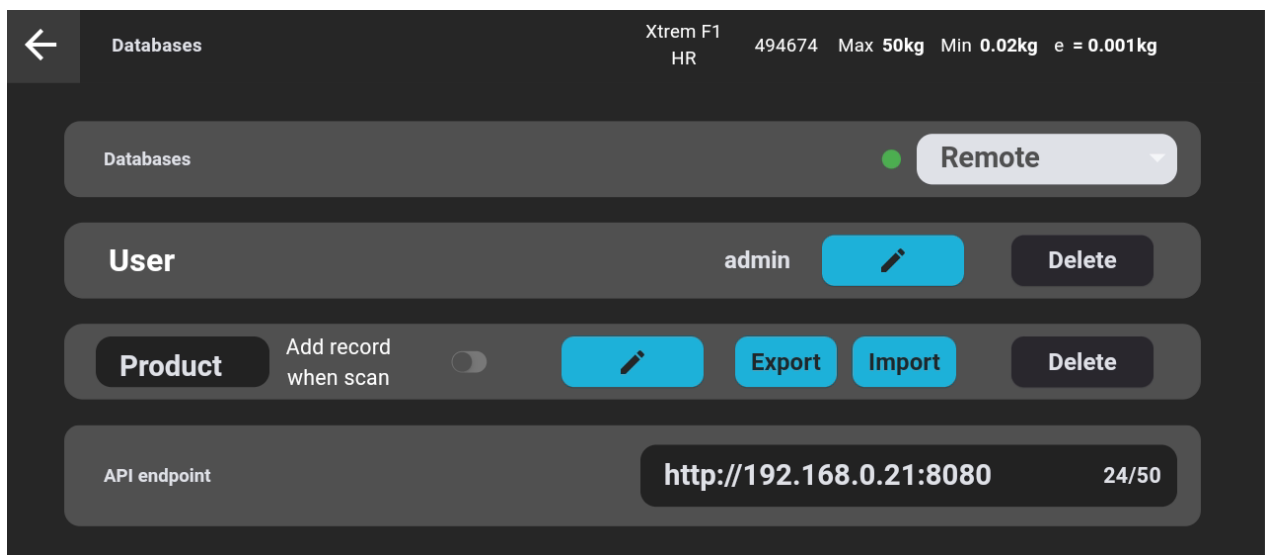
Nel servizio API è possibile verificare che le informazioni necessarie affinché l'API determini a quale database connettersi siano fornite nel file *.env*.

```
# =====  
# Configuracion de la base de datos  
# =====  
  
SPRING_DATASOURCE_URL=jdbc:mysql://db:3306/open_z8  
SPRING_DATASOURCE_USERNAME=apiuser  
SPRING_DATASOURCE_PASSWORD=Gram1231
```

Se si desidera aggiornare il file *app.jar* con l'API, è sufficiente sostituire il file ed eseguire nuovamente il file *instalar.bat*, oppure *instalar.sh* nel caso di Linux.

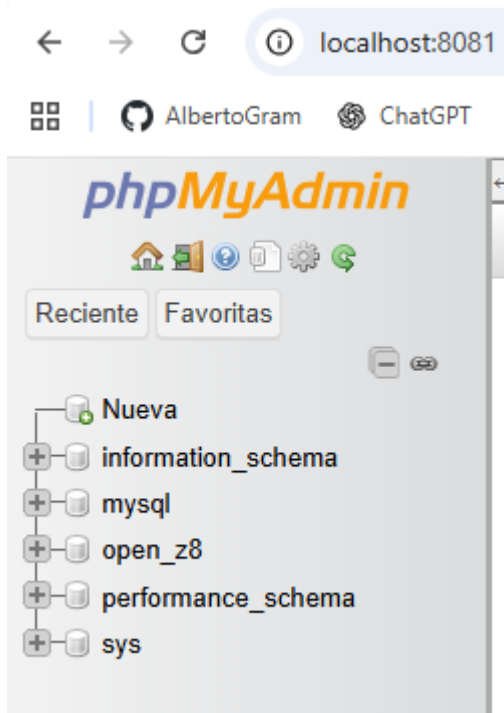
Infine, per effettuare le richieste, queste devono essere eseguite al seguente URL:

http://{indirizzo_server}:{porta}



Per connettersi a phpMyAdmin e interagire con il database del container MySQL, aprire una finestra del browser e accedere al seguente endpoint:

<http://localhost:8081>



ERRORI

Se, durante l'avvio di Docker, viene visualizzato il seguente messaggio:



Virtualization support not detected

Docker Desktop failed to start because virtualisation support wasn't detected. Contact your IT admin to enable virtualization or check system requirements.

[View system requirements](#) ↗

Ciò è dovuto al fatto che la virtualizzazione non è attivata sul dispositivo. Attivarla per risolvere il problema.

Se si utilizza il database Docker, assicurarsi che l'endpoint punti a *db* e non a *localhost*:

```
# =====  
# Configuracion de la base de datos  
# =====  
  
SPRING_DATASOURCE_URL=jdbc:mysql://db:3306/open_z8  
SPRING_DATASOURCE_USERNAME=apiuser  
SPRING_DATASOURCE_PASSWORD=Gram1231
```

D'altra parte, se si desidera utilizzare un database situato sullo stesso dispositivo di Docker, utilizzare *host.docker.internal*, poiché *localhost* (127.0.0.1) verrà interpretato come il container e non come l'host.

```
# =====  
# Configuracion de la base de datos  
# =====  
  
SPRING_DATASOURCE_URL=jdbc:mysql://host.docker.internal:3306/open_z8  
SPRING_DATASOURCE_USERNAME=root  
# SPRING DATASOURCE PASSWORD=Gram1231
```

Si ricorda che se un servizio, come WAMP o un altro, utilizza la porta 3306, non sarà più possibile utilizzare la stessa porta per il database Docker locale. Sarà quindi necessario arrestare tale servizio oppure utilizzare una porta diversa, modificando il file *docker-compose*.



Gram Precision S.L.

Travesía Industrial, 11 · 08907 Hospitalet de Llobregat · Barcelona (Spain)

Tel. +34 93 300 33 32

www.gram-group.com