Installing OpenMRS on Docker

Docker is a unique container/image based platform made to let people distribute applications easily.

Our official images are deployed to Docker Hub.

- reference application distro
- platform distro

Those images are automatically generated, and all our test environments are using them.

If you are interested in running the reference application with demo data (like https://demo.openmrs.org), check demo repository, as it contains the demo data as well.

Installing Docker and docker-compose

Running Docker locally

See Docker Installation Instructions for your environment. If running on Linux, check https://docs.docker.com/compose/install/ to install docker compose.

Running Docker on Digital Ocean

Digital Ocean provides low case and convenient cloud hosting for development and testing. There may be cheaper alternatives for permanently hosting a website, but Digital Ocean is one of the fastest ways to get a temporary server up and running.

1. Create an account (free) if needed, then sign into digitalocean.com.
2. Create a droplet (takes 1 minute or less)
   a. Name: "openmrs" (or whatever you like)
   b. Size: at least 2 GB of memory (choosing the default environment or any one with less than 2 GB of memory may fail to run OpenMRS)
   c. Image: select Applications > Docker on 14.04
3. Once the droplet is created, log into it:
   a. View the droplet info to get its IP address and SSH into it – e.g., `ssh root@xxx.xxx.xxx.xxx` – using the password for the droplet you received via email.
   b. Access the droplet's terminal directly through the Digital Ocean website, by selecting your droplet and selecting Access > Console Access, using the password for the droplet you received via email.

Do you have Docker running?

```
docker -v
```

Docker should report its version number, which should be 1.5.0 or higher. If you get an error message, then you haven't gotten Docker installed properly; go back through the installation process and/or Google for solutions.

Option 1 - Deploying OpenMRS within Docker command

> ⚠ This is not the recommended approach. We recommend using docker-compose (manually created or using the SDK) whenever possible.

The video tutorial referencing the original work done by Chaitya Shah can be seen here (warning: this is mostly not working):

Run MySQL Database within Docker

https://hub.docker.com/_/mysql/

Run OpenMRS within Docker

Running interactively

In this mode, OpenMRS will be running until you cancel (e.g., using `Ctrl+C`) or close your terminal.

```
$ docker run -it --rm -p 8080:8080 openmrs/openmrs-reference-application-distro:demo
```

> If you are running docker directly on a linux host or, for any other reason, don't want to use the default web port, you can change the port for OpenMRS by changing the `-p 80:8080` parameter. For example, to run OpenMRS on port 8888, use the parameter `-p 8888:8080`.

Once you see

```
INFO: Server startup in --------- ms
```

OpenMRS should be up and running.

Running in background

In this mode, OpenMRS will run in the background and continue running – even if you exit your terminal – until you stop it.

```
docker run -d --name openmrs -p 80:8080 openmrs/openmrs-reference-application-distro
```

> If you are running docker directly on a linux host or, for any other reason, don't want to use the default web port, you can change the port for OpenMRS by changing the `-p 80:8080` parameter. For example, to run OpenMRS on port 8888, use the parameter `-p 8888:8080`.

The above command will run a docker container with the name "openmrs" as a background daemon (`-d`), mapping it to the default web port (80), and will return you to a command line. The console output for OpenMRS will be logged into that containers log.

To view the tail of the background container's log:

```
docker logs openmrs
```

To follow the background container's log file (quit watching by pressing `Ctrl+C`):

```
docker logs -f openmrs
```

To delete the OpenMRS container (e.g., if you want to start over with a new container):

```
docker rm -f openmrs
```

Browse to OpenMRS

The location of OpenMRS will depend on how you ran it and which environment you are using for Docker.

- If you are running Docker locally, you can probably find OpenMRS at http://192.168.59.103/openmrs/ or http://0.0.0.0/openmrs
- If you are running Docker locally, but specified a different port (e.g., `-p 8888:8080`), then you can probably find OpenMRS at that port via http://192.168.59.103:8888/openmrs/
- If you are running Docker on Digital Ocean droplet, you will need to find the IP Address of your droplet through your digitalocean.com account.

Once you've found your OpenMRS installation, you should see the installation wizard and can follow the steps of the wizard to complete the installation.

> ⚠ When the installation wizard asks for the root password for the database, enter the password: **test**

When the installation wizard is finished, you should be redirected to the login screen.

Username: **admin**
Password: **Admin123**

Option 2 - Running MySQL and OpenMRS in Docker using the SDK

Check the `build-distro` command in our SDK.

You can use the SDK to generate the docker-compose files.

Option 3 - Running MySQL and OpenMRS using direct docker-compose

Check build-distro on how to generate the docker-compose files (or copy from some of our test OpenMRS instances).

An example of a working docker-compose file:

docker-compose.yml

```
version: '2.1'

services:
  openmrs-referenceapplication-mysql:
    restart: "always"
    image: mysql:5.6
    command: "mysqld --character-set-server=utf8 --collation-server=utf8_general_ci"
    environment:
      MYSQL_DATABASE: ${MYSQL_DB:-openmrs}
      MYSQL_ROOT_PASSWORD: ${MYSQL_ROOT_PASSWORD:-Admin123}
      MYSQL_USER: ${MYSQL_USER:-openmrs}
      MYSQL_PASSWORD: ${MYSQL_PASSWORD:-Admin123}
    healthcheck:
      test: "exit 0"
    volumes:
      - openmrs-referenceapplication-mysql-data:/var/lib/mysql

  openmrs-referenceapplication:
    restart: "always"
    image: openmrs/openmrs-reference-application-distro:demo
    depends_on:
      - openmrs-referenceapplication-mysql
    ports:
      - "8080:8080"
    environment:
      DB_DATABASE: ${MYSQL_DB:-openmrs}
      DB_HOST: openmrs-referenceapplication-mysql
      DB_USERNAME: ${MYSQL_USER:-openmrs}
      DB_PASSWORD: ${MYSQL_PASSWORD:-Admin123}
      DB_CREATE_TABLES: 'true'
      DB_AUTO_UPDATE: 'true'
      MODULE_WEB_ADMIN: 'true'
    healthcheck:
      test: ["CMD", "curl", "-f", "http://localhost:8080/openmrs/"]
      timeout: 20s
    volumes:
      - openmrs-referenceapplication-data:/usr/local/tomcat/.OpenMRS/
      - /usr/local/tomcat/.OpenMRS/modules/ # do not store modules in data
      - /usr/local/tomcat/.OpenMRS/owa/ # do not store owa in data

volumes:
   openmrs-referenceapplication-mysql-data:
   openmrs-referenceapplication-data:
```

Make sure you have a *docker-compose.yml* file on the current directory:

```
# To start database and OpenMRS with default credentials:
$ docker-compose up -d

# To override credentials, use environments variables
$ MYSQL_DB=my_database MYSQL_ROOT_PASSWORD=my_root_password MYSQL_USER=my_user MYSQL_PASSWORD=my_password docker-compose up -d

# To see logs
$ docker-compose logs -f

# Bring containers down
$ docker-compose down

# Bring containers down and delete all the data

$ docker-compose down -v
```

You should be able to access in your browser **http://localhost:8080/openmrs/**

When the installation wizard is finished, you should be redirected to the login screen.

Username: **admin**
Password: **Admin123**

When running on Linux, it should be trivial to create backups of the docker volumes (there's a folder named 'volumes' inside your docker folder installation).

If you don't want to run MySQL in docker (or on the same machine), delete the 'openmrs-referenceapplication-mysql' service and check docker image to configure the right environment variables for OpenMRS.