**Phần 1: Docker là gì?**

**A.1 Docker là gì?**

[Docker là một nền tảng](https://www.geeksforgeeks.org/introduction-to-docker/)[chứa ứng dụng](https://www.geeksforgeeks.org/containerization-using-docker/) nguồn mở mà qua đó bạn có thể đóng gói ứng dụng của mình và tất cả các phần phụ thuộc của nó vào một đơn vị tiêu chuẩn hóa được gọi là bộ chứa. Các thùng chứa có trọng lượng nhẹ nên có thể di chuyển được và chúng được cách ly với cơ sở hạ tầng bên dưới cũng như với các thùng chứa khác. Bạn có thể chạy [hình ảnh docker](https://www.geeksforgeeks.org/what-is-docker-image/) dưới dạng [bộ chứa docker](https://www.geeksforgeeks.org/virtualisation-with-docker-containers/) trong bất kỳ máy nào có cài đặt docker mà không phụ thuộc vào [hệ điều hành.](https://www.geeksforgeeks.org/what-is-an-operating-system/)

**A.2 Các thành phần chính của Docker**

Sau đây là một số thành phần chính của Docker:

**Docker Engine:** Đây là một phần cốt lõi của docker, xử lý việc tạo và quản lý các container.

**Docker Image:** Đây là mẫu chỉ đọc được sử dụng để tạo vùng chứa, chứa mã ứng dụng và các phần phụ thuộc.

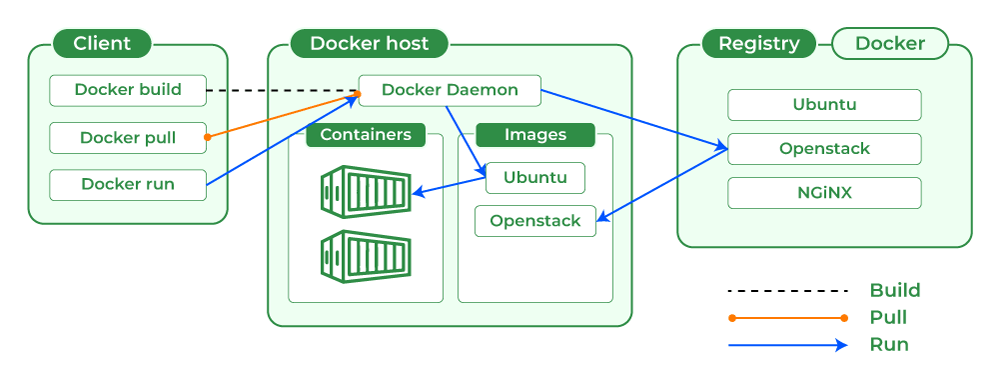
**Docker Hub:** Đây là kho lưu trữ dựa trên đám mây được sử dụng để tìm và chia sẻ hình ảnh vùng chứa.

**Dockerfile:** Đây là một tập lệnh chứa các hướng dẫn để xây dựng hình ảnh docker.

**Docker Registry**: Đây là một hệ thống phân phối lưu trữ cho hình ảnh docker, nơi bạn có thể lưu trữ hình ảnh ở cả chế độ công khai và riêng tư.

**A.3 Kiến trúc Docker là gì và Docker hoạt động như thế nào?**

Docker sử dụng **kiến ​​trúc client-server**. Máy khách Docker giao tiếp với trình nền docker giúp xây dựng, chạy và phân phối các vùng chứa docker. Máy khách Docker chạy với daemon trên cùng một hệ thống hoặc chúng ta có thể kết nối máy khách Docker với daemon Docker từ xa. Với sự trợ giúp của API REST qua ổ cắm UNIX hoặc mạng, máy khách docker và daemon tương tác với nhau.



**A.4 Lệnh Docker**

Thông qua việc giới thiệu các lệnh docker thiết yếu, docker đã trở thành một phần mềm mạnh mẽ trong việc đơn giản hóa quy trình quản lý container. Nó giúp đảm bảo quy trình phát triển và triển khai liền mạch. Sau đây là một số lệnh docker được sử dụng phổ biến:

* **Docker Run:** Nó được sử dụng để khởi chạy các container từ hình ảnh, với việc chỉ định các tùy chọn và lệnh thời gian chạy.
* **Docker Pull:** Nó tìm nạp các hình ảnh vùng chứa từ sổ đăng ký vùng chứa như Docker Hub về máy cục bộ.
* **Docker ps** : Nó giúp hiển thị các container đang chạy cùng với thông tin quan trọng của chúng như ID container, hình ảnh được sử dụng và trạng thái.
* **Docker Stop** : Nó giúp tạm dừng các container đang chạy và tắt các tiến trình bên trong chúng một cách nhẹ nhàng.
* **Docker Start:** Nó giúp khởi động lại các container đã dừng, tiếp tục hoạt động của chúng từ trạng thái trước đó.
* **Đăng nhập Docker:** Nó giúp đăng nhập vào sổ đăng ký docker cho phép truy cập vào các kho lưu trữ riêng tư.

**A.5 Công cụ Docker**

Phần mềm lưu trữ các container có tên là Docker Engine. Docker Engine là một ứng dụng dựa trên máy khách-máy chủ. Docker engine có 3 thành phần chính:

1. **Máy chủ:** Nó chịu trách nhiệm tạo và quản lý hình ảnh, vùng chứa, mạng và khối lượng Docker trên Docker. Nó được gọi là một quá trình daemon.
2. [**API REST**](https://www.geeksforgeeks.org/rest-api-introduction/)**:** Nó chỉ định cách các ứng dụng có thể tương tác với Máy chủ và hướng dẫn nó những việc cần làm.
3. **Máy khách:** Máy khách là giao diện dòng lệnh docker (CLI), cho phép chúng ta tương tác với Docker bằng các lệnh docker.

**A.6 Sự khác biệt giữa Docker Container và Máy ảo**

Sau đây là những khác biệt giữa vùng chứa docker và Máy ảo:

| **Docker container** | **Máy ảo** |
| --- | --- |
| Docker Container chứa các tệp nhị phân, thư viện và tệp cấu hình cùng với chính ứng dụng. | Máy ảo (VM) chạy trên Hypervisor, cho phép nhiều Máy ảo chạy trên một máy cùng với hệ điều hành của chính nó. |
| Chúng không chứa hệ điều hành khách cho mỗi vùng chứa và dựa vào nhân hệ điều hành cơ bản, điều này làm cho vùng chứa nhẹ. | Mỗi VM có bản sao hệ điều hành riêng cùng với ứng dụng và các tệp nhị phân cần thiết, điều này làm cho nó lớn hơn đáng kể và đòi hỏi nhiều tài nguyên hơn. |
| Các bộ chứa chia sẻ tài nguyên với các bộ chứa khác trong cùng một hệ điều hành máy chủ và cung cấp khả năng cách ly quy trình ở cấp hệ điều hành. | Chúng cung cấp khả năng cách ly quy trình ở cấp Phần cứng và khởi động chậm. |