Là 1 phần mềm open source cho phép nhiều sản phẩm chạy trong nó và giao tiếp vơi nhau.

* Triển khai ứng dụng nhanh, thống nhất trên nhiều môi trường khác nhau.
* Quản lý container không quan tâm đến kiến trúc hệ thống nền dưới.
* Tiêu tốn ít tài nguyên lưu trữ và chia sẻ dễ dàng **container image** qua **regisstry** với các docker khác nhau
* Dễ dàng cập nhật **container image** từ bản gốc với cấu trúc **layered file system** trùy biến cho triển khai container
* Docker được cài đặt và chia swer tài nguyên với HĐH như MACOS, WINDOW, LINUX
* Đóng gói ứng dụng cần triển khai cùng vs thư viện và cấu hình cần triển khai tạo thành 1 hệ thống file chia lớp (container image.) và chạy độc lập trong docker gọi là container **giống như 1 process trên OS.**
* Viết bằng ngôn ngữ **GO.** Sử dụng namespace(các khu vực được phân chia: có quy tắc về đặt tên, nảo mật) để tạo và quản lý container chạy trong nó.

Kiến trúc:

* Docker client: Chứa tập lệnh docker, Docker file gốm nhiều lệnh.
* Docker deamon (dockerd): Nhận lệnh và quản lý các đối tượng trong docker server.
* Images: Là một template mẫu sẵn chỉ đọc, chứa tạp lệnh để tạo một container.
* Container là một ứng dụng chạy độc lập từ một image.
* Registry: Lưu và chia sẻ images.

**Docker file** -> build-> **Images** ->docker run ->  **container** ->docker exec -> **Bash**

Pull

Registry

**Image và container:**

Image: Là mẫu container chỉ đọc, để tạo container.

* Dựa trên layer file system, hệ thống file chia nhiều lớp. Do vậy, một image có thể dượcd tạo ra từ **base image**  với các tùy biến bằng cách tạo một lớp khác bên trên của **base image** chứa lệnh **docker**  bổ dụng (**docker file**)
* Những Images mới chỉ chứa những lệnh khác so với lớp base. Vì thế dụng lượng của image nhẹ
* Khi chạy image, Docker sẽ hợp nhất base image với lớp image ở trên cùng để chạy.
* Images có thể được lưu và chia sẻ lên registry.
* Vd: Images dựa trên Ubuntu image nhưng kèm thei Tomcat-server, JDK

Container: (instance of images) là một bản chạy của một images đọc lập trong **docker server** như một process trong OS.

* Độc lập với nhau và với máy chủ.
* Có thể kết nối contianer với mạng, thêm bộ nhớ lưu trữ, hoặc tạo images mới tại thời điểm đó.
* Có thể được kết nối dùng lệnh Bash trên container.