**Containerisation of a Chatbot**

**ABSTRACT**

A container is a standard software unit that encapsulates code and all of its dependencies so that the program can be moved from one computing environment to another fast and reliably. A Docker container image is a small, standalone software package that contains everything needed to run a program, including code, runtime, system tools, system libraries, and settings. At runtime, container images become containers, and when Docker containers run on Docker Engine, images become containers. Containerized software, which is available for both Linux and Windows-based applications, will always run the same independent of the infrastructure. Containers separate software from its surroundings and ensure that it performs consistently despite variances in development and staging environments, for example.

**1 INTRODUCTION**

The Docker container technology was first released as an open source Docker Engine in 2013. It made use of existing computing concepts such as containers and, more particular, primitives known as cgroups and namespaces in the Linux environment. Docker's technology is unusual in that it focuses on developers' and systems administrators' needs to isolate application dependencies from infrastructure.

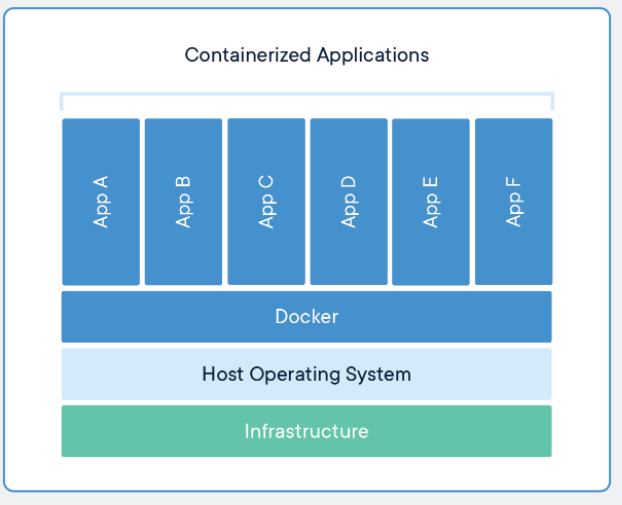
Docker containers that run on Docker Engine include the following:

* **Standard:** Docker established the industry standard for containers, allowing them to be transported anywhere.

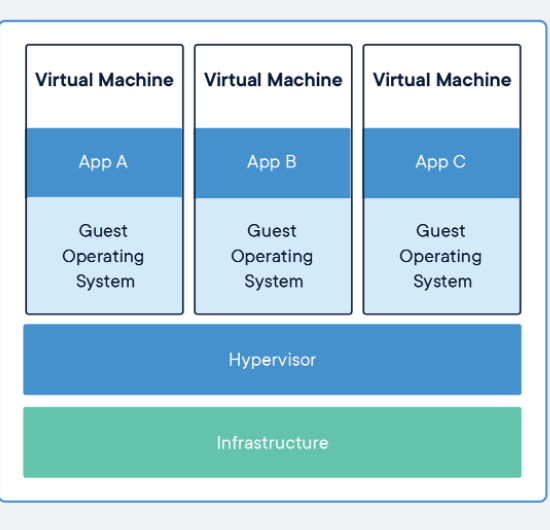
* **Lightweight:** Containers share the machine's OS system kernel, thus they don't need their own OS, resulting in improved server efficiency and lower server and licensing costs.
* **Secure:** Containerized applications are more secure, and Docker has the strongest default isolation features in the industry.

**Virtual Machines and Comparing Containers**

* **Containers:** Containers are an abstraction that groups code and dependencies together at the app layer. Multiple containers can run on the same machine and share the operating system kernel, each executing as a separate process in user space. Containers take up less space than virtual machines (container images are often in the tens of megabytes), can accommodate more applications, and require fewer virtual machines and Operating systems.

****

* **Virtual Machines:** Virtual machines (VMs) are abstractions of physical hardware that turn a single server into a collection of servers. Multiple virtual machines can run on a single system thanks to the hypervisor. Each VM contains a full copy of the operating system, the application, as well as the necessary binaries and libraries, which can take up tens of GBs. VMs can also take a long time to boot.

****

Pulling an anaconda docker image

Text

Description automatically generated

Creating a container from the image pulled above.

Graphical user interface, text, application

Description automatically generated

Installing ntlk on docker container Text

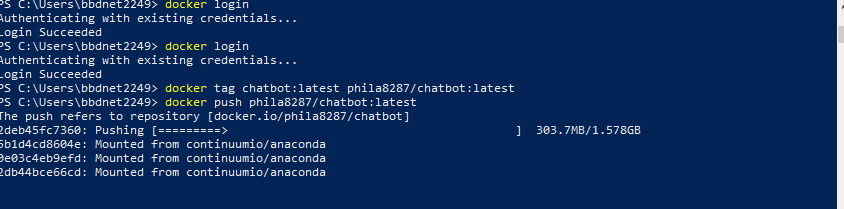
Description automatically generated

Running application from container

Graphical user interface, text

Description automatically generated

Publishing to Docker hub



**link to dockerhub:**[**https://hub.docker.com/repository/docker/phila8287/chatbot**](https://hub.docker.com/repository/docker/phila8287/chatbot)