





DevOps

I'm Not a Programmer

Infrastructure as Code

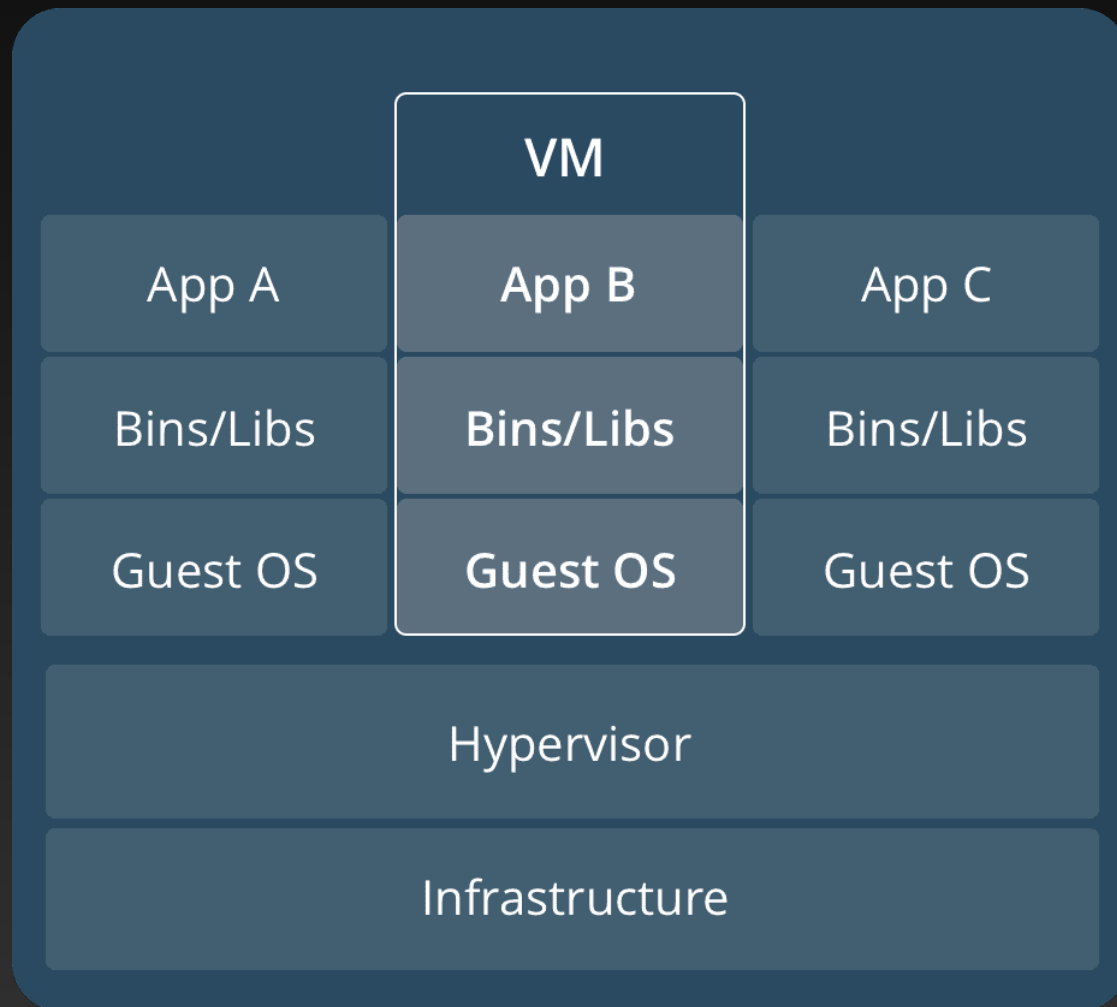
I'm Not a Programmer

```
function register()
{
    if (empty($_POST)) {
        $msg = '';
        if ($_POST['user_name']) {
            if ($_POST['user_password_new']) {
                if ($_POST['user_password_new'] == $_POST['user_password_repeat']) {
                    if (strlen($_POST['user_password_new']) > 5) {
                        if (strlen($_POST['user_name']) < 65 && strlen($_POST['user_name']) > 1) {
                            if (preg_match('/^[a-z\d]{2,64}$/i', $_POST['user_name'])) {
                                $user = read_user($_POST['user_name']);
                                if (!isset($user['user_name'])) {
                                    if ($_POST['user_email']) {
                                        if (strlen($_POST['user_email']) < 65) {
                                            if (filter_var($_POST['user_email'], FILTER_VALIDATE_EMAIL)) {
                                                create_user();
                                                $_SESSION['msg'] = 'You are now registered so please login';
                                                header('Location: ' . $_SERVER['PHP_SELF']);
                                                exit();
                                            } else $msg = 'You must provide a valid email address';
                                        } else $msg = 'Email must be less than 64 characters';
                                    } else $msg = 'Email cannot be empty';
                                } else $msg = 'Username already exists';
                            } else $msg = 'Username must be only a-z, A-Z, 0-9';
                        } else $msg = 'Username must be between 2 and 64 characters';
                    } else $msg = 'Password must be at least 5 characters';
                } else $msg = 'Passwords do not match';
            } else $msg = 'Empty Password';
        } else $msg = 'Empty Username';
        $_SESSION['msg'] = $msg;
    }
    return register_form();
}
```

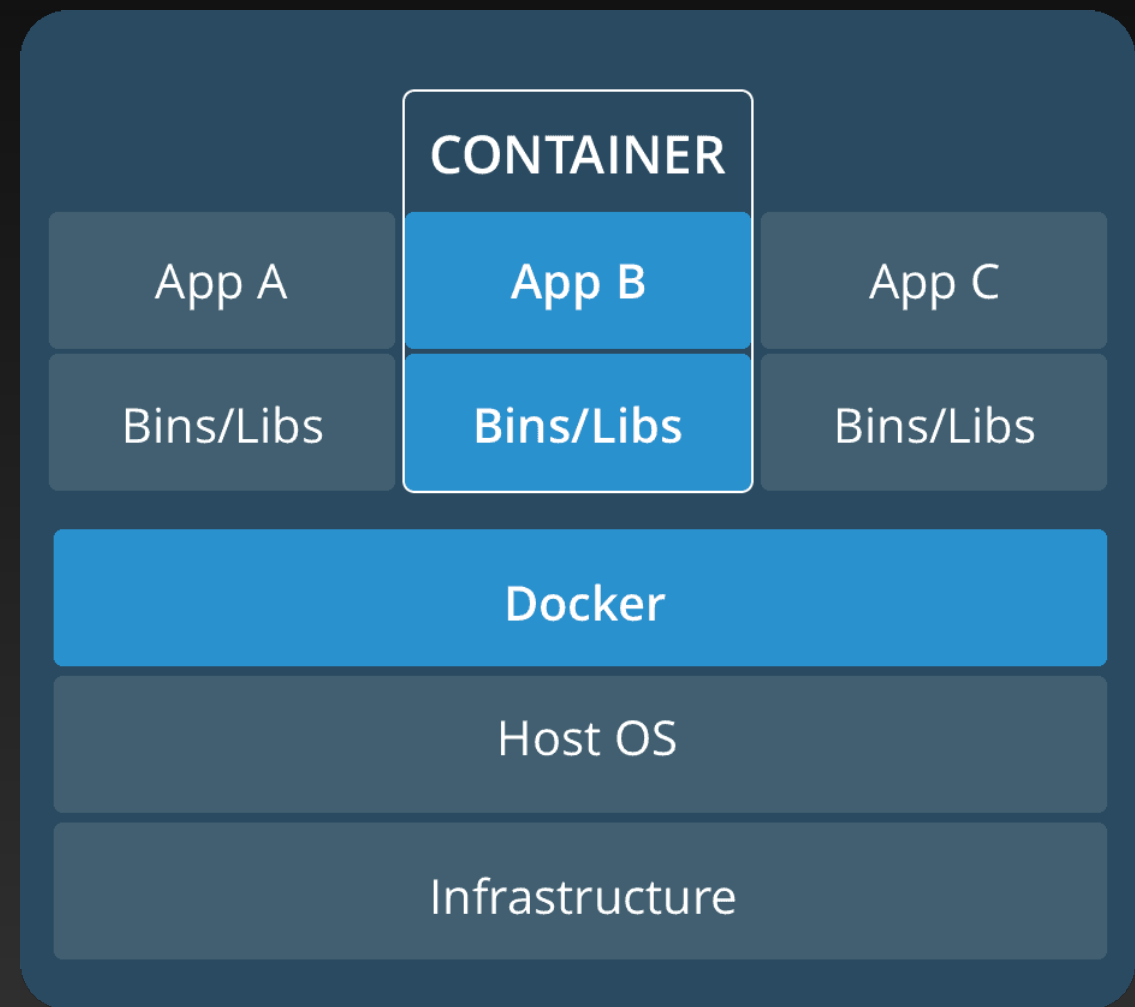


pikaburu

VM

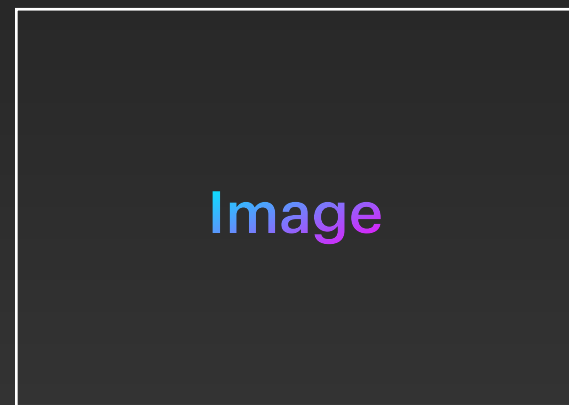
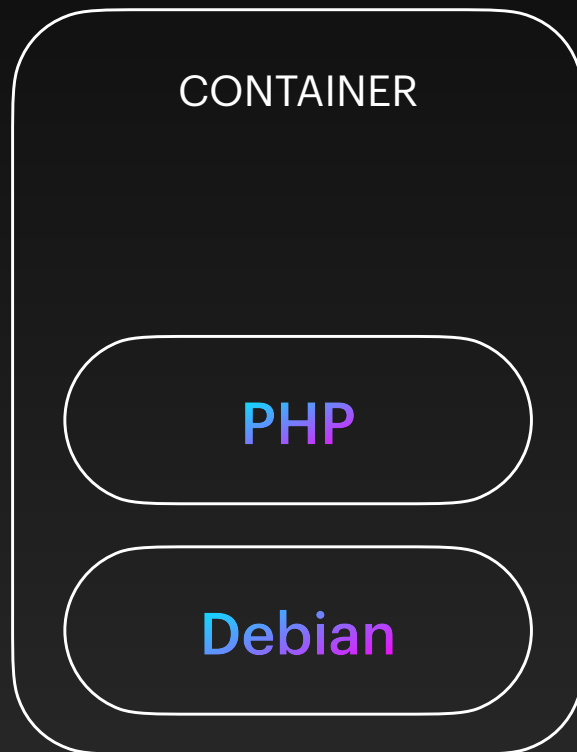


Docker

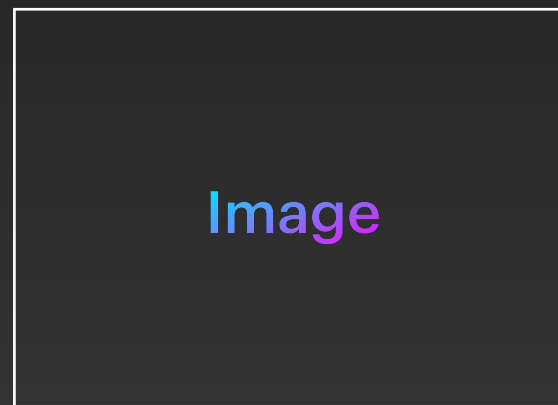
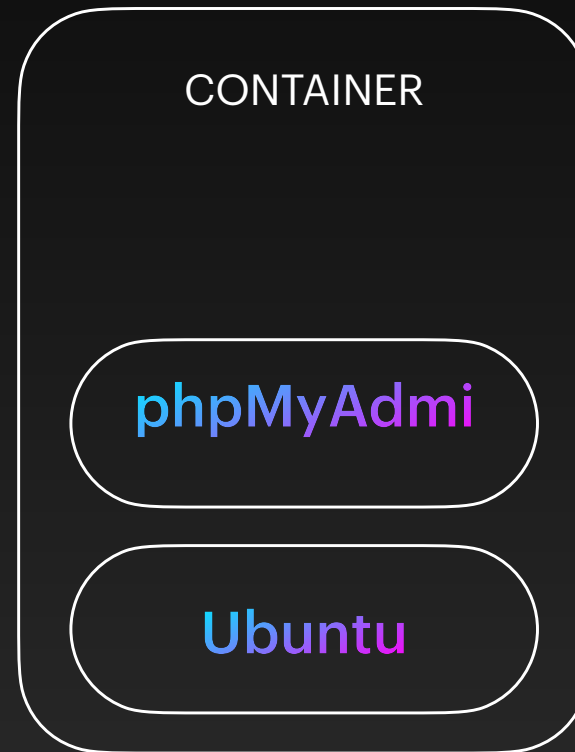


Containerization

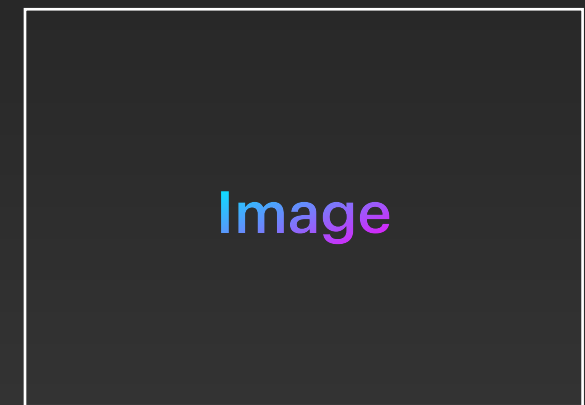
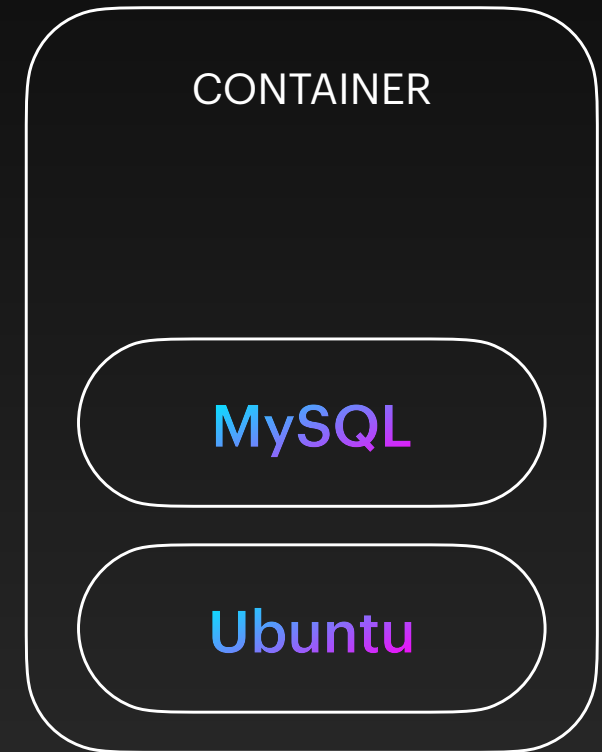
Shopping Web



phpMyAdmin

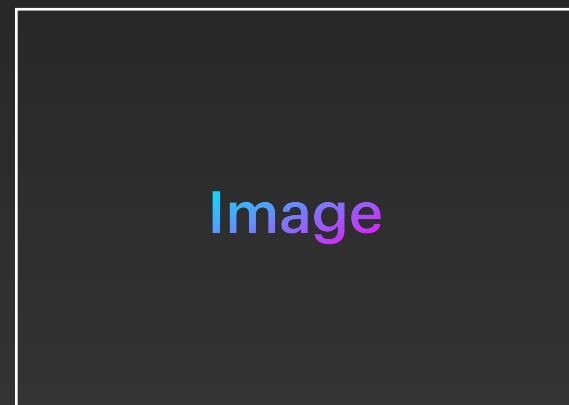
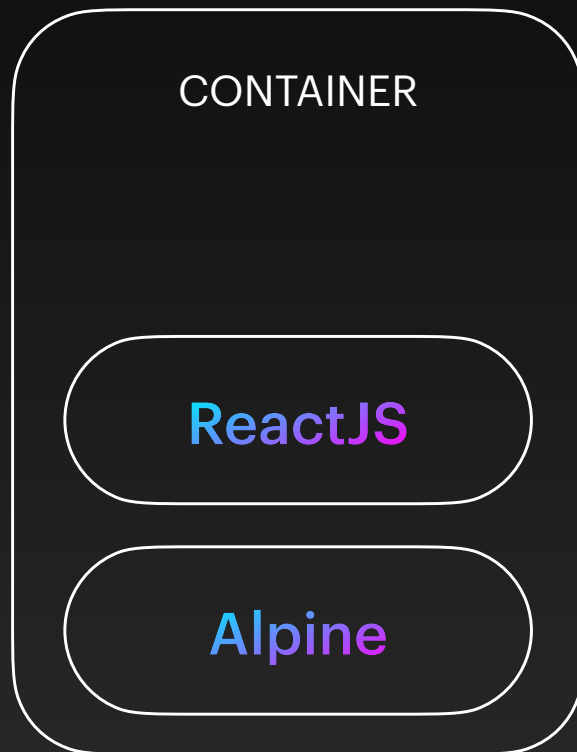


Database

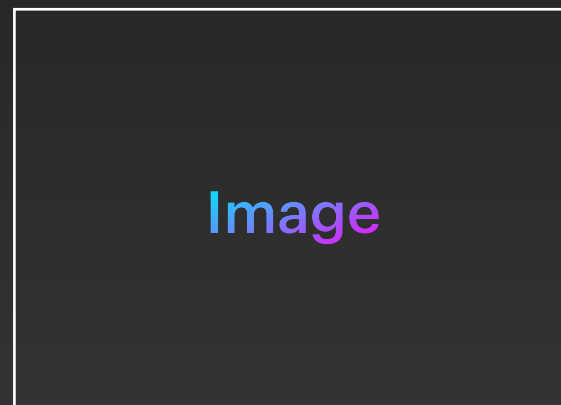
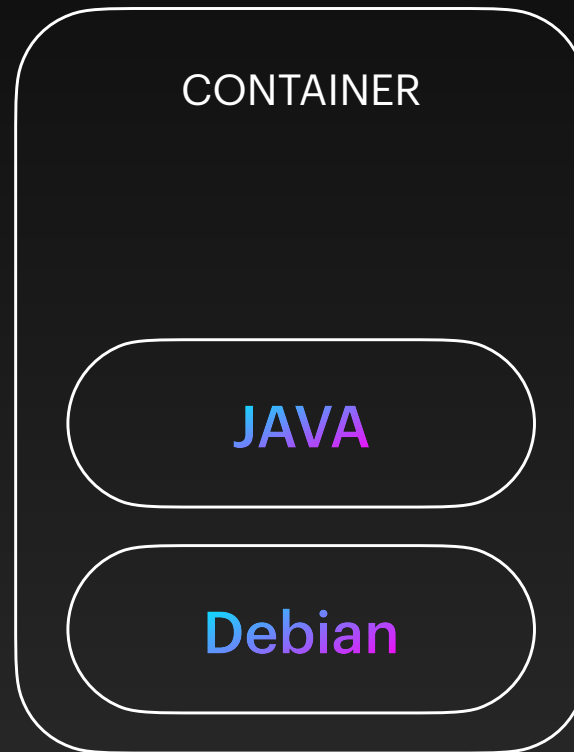


Containerization

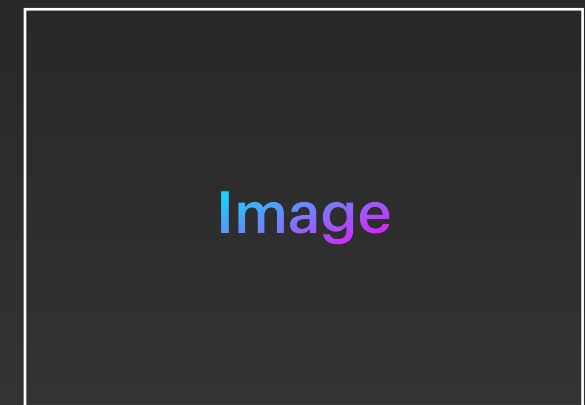
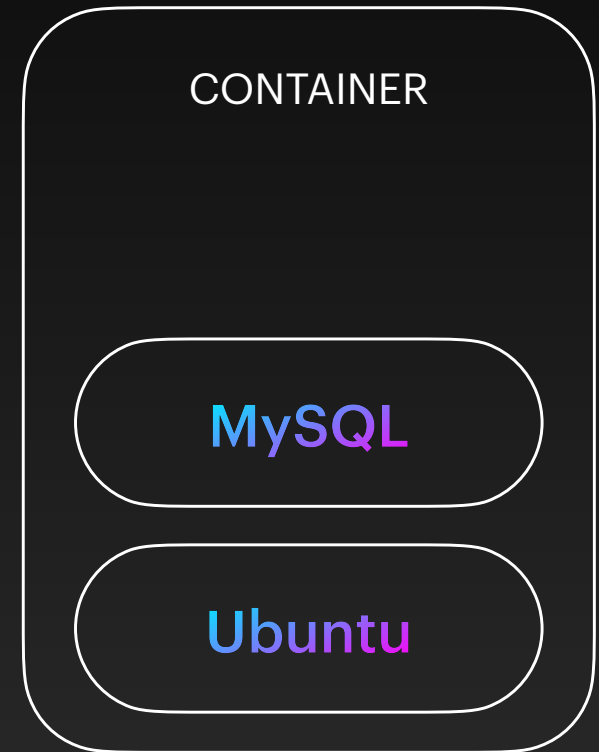
Shopping Web



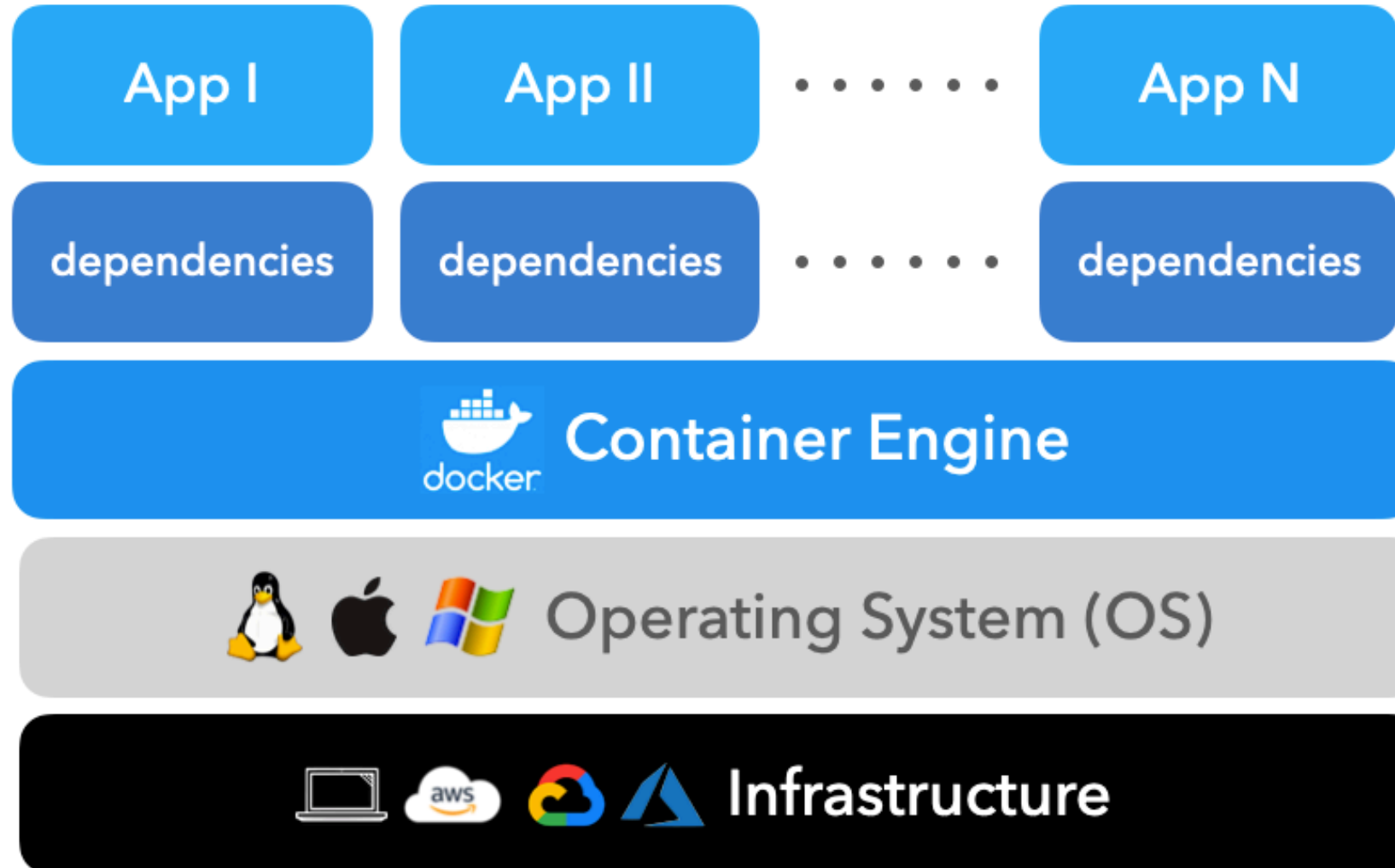
Shopping API



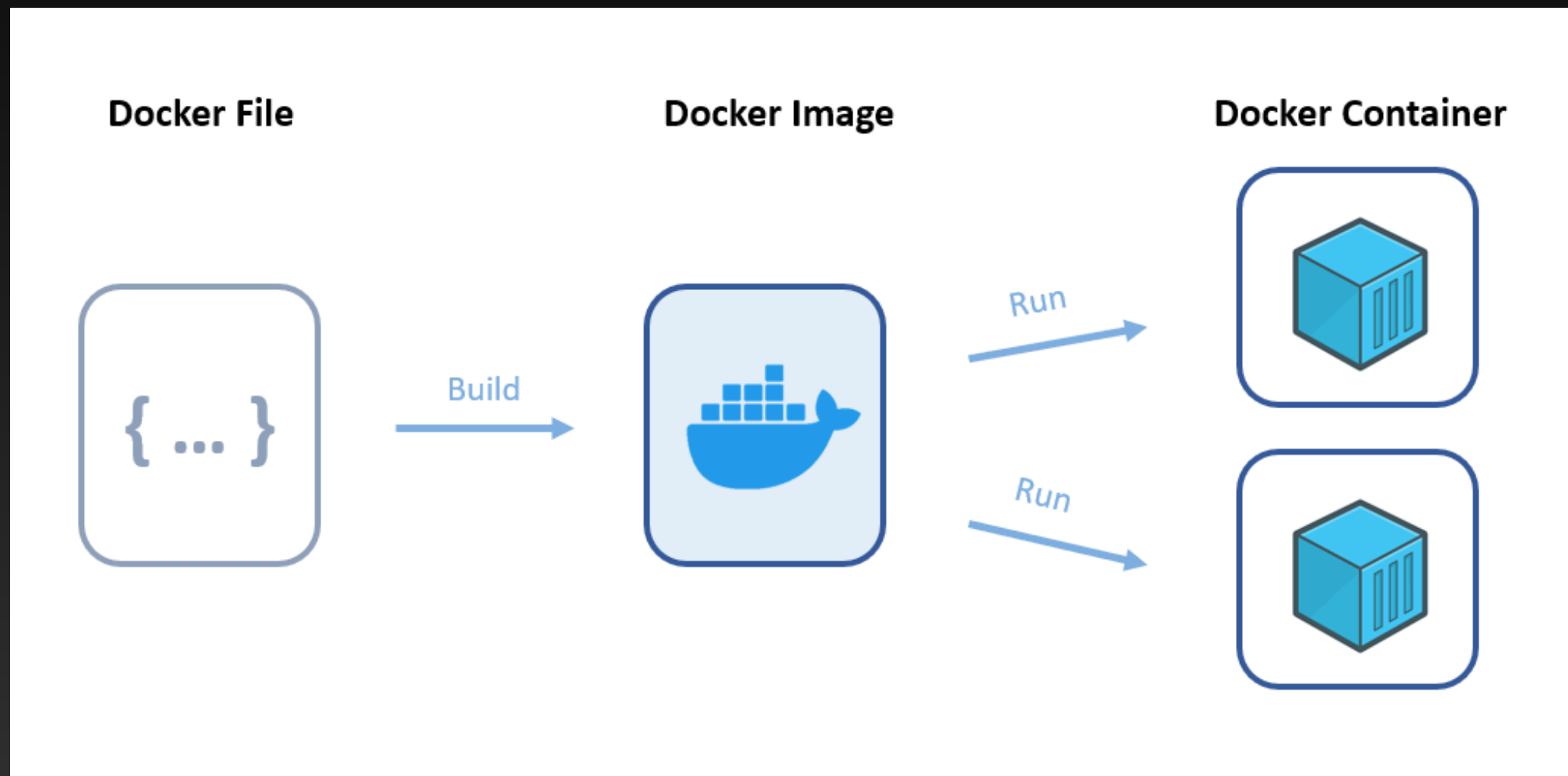
Database



Architecture



Infrastructure as Code



DevOps

Microservices Architecture

<https://www.blognone.com/node/95873>

Workshop

Docker

DevOps :: Infrastructure as Code

Hello World

Docker

DevOps :: Infrastructure as Code

Hello World

Docker

```
$ docker run hello-world
```

Create Project

Docker

DevOps :: Infrastructure as Code

Dockerfile

Docker

DevOps :: Infrastructure as Code

Dockerfile

./php/Dockerfile

```
FROM php:8.0-apache
```

```
RUN docker-php-ext-install mysqli && docker-php-ext-enable mysqli
```

```
RUN apt-get update && apt-get upgrade -y
```

```
EXPOSE 80
```

Docker Compose

Docker

DevOps :: Infrastructure as Code

Docker Compose

./docker-compose.yml

```
version: '3.8'
services:
```

```
  php:
    build: ./php
    container_name: php_server
    restart: always
    depends_on:
      - db
    volumes:
      - ./public_html:/var/www/html
    ports:
      - 80:80
```


Project

The image shows a code editor interface with a dark theme. The Explorer sidebar on the left shows a project structure with a 'TEST' folder containing a 'php' subfolder. Inside 'php', there are files for 'Dockerfile' and 'docker-compose.yml'. The main editor area is split into two panes. The left pane shows the 'Dockerfile' with the following content:

```
1 FROM php:8.0-apache
2 RUN docker-php-ext-install mysqli && docker-php-ext-en
3 RUN apt-get update && apt-get upgrade -y
4
5 EXPOSE 80
6
```

The right pane shows the 'docker-compose.yml' file with the following content:

```
1 version: '3.8'
2 services:
3
4   php:
5     build: ./php
6     container_name: php_server
7     restart: always
8     volumes:
9       - ./public_html:/var/www/html
10    ports:
11      - 80:80
12
```

The status bar at the bottom indicates the current position is 'Ln 6, Col 1', with settings for 'Spaces: 4', 'UTF-8', 'CRLF', and 'Docker'.

Test Run Docker

Docker

DevOps :: Infrastructure as Code

Test Run Docker

Docker

1. Build Image

```
$ docker compose build
```

2. Run

```
$ docker compose up -d
```

3. Down

```
$ docker compose down
```

Docker CLI

docker command-line interface

1. Show Image?

```
$ docker images
```

2. Show Running Container, All Container

```
$ docker ps
```

```
$ docker ps -a
```

3. Show Log Container

```
$ docker logs -f <container_name>
```

4. Delete Container

```
$ docker rm -f <container_name>
```

5. Delete Container

```
$ docker rmi <image_name>
```

6. etc,...

Deployment

Docker

DevOps :: Infrastructure as Code

LMP Stack

Linux + Mysql + Php

github.com/foptun/kvc-docker

Server Example

Docker

DevOps :: Infrastructure as Code

WSL

Install WSL2

- `wsl —version`
- `wsl —update`

Ref

- <https://9mza.net/post/docker-apache-php-mysql>