**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TR** **ƯỜNG ĐẠI HỌC SƯ PHẠM KĨ THUẬT TPHCM**

-----🙞🙜🕮🙞🙜-----



**BÁO CÁO CUỐI KỲ**

**ĐIỆN TOÁN ĐÁM MÂY**

**Đề tài:**

**SỬ DỤNG DOCKER ĐỂ ẢO HÓA**

**SERVER UBUNTU**

**GVHD: TS. HUỲNH XUÂN PHỤNG**

**MÃ HỌC PHẦN: 221CLCO332779**

**NHÓM SINH VIÊN THỰC HIỆN:**

**1. Vũ Trung Hiếu 20110482**

**2. Nguyễn Gia Huy 20110103**

Thành phố Hồ Chí Minh, Tháng 12 năm 2022

**NHẬN XÉT, ĐÁNH GIÁ ĐỒ ÁN**

HỌC KÌ I, NĂM HỌC 2022-2023

**Đề tài: Sử dụng Docker để ảo hoá server Ubuntu**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Tỷ lệ % hoàn thành** |
| 1 | Nguyễn Gia Huy | 20110103 | 100% |
| 2 | Vũ Trung Hiếu | 20110482 | 100% |

Ghi chú:

* Tỷ lệ % = 100 %
* Trưởng nhóm: Nguyễn Gia Huy

**Nhận xét của giảng viên**

...................................................................................................................................

...................................................................................................................................

...................................................................................................................................

...................................................................................................................................

...................................................................................................................................

...................................................................................................................................

...................................................................................................................................

...................................................................................................................................

Ngày … tháng 12 năm 2022 Giáo viên chấm điểm

**LỜI CẢM ƠN**

“Để có thể hoàn thành đồ án cuối kỳ môn học này, chúng em xin gửi lời cảm ơn chân thành đến: Ban giám hiệu trường Đại Học Sư phạm Kỹ thuật Thành phố Hồ Chí Minh vì đã tạo điều kiện về cơ sở vật chất với những phòng học và phòng thực hành hiện đại, hệ thống thư viện đa dạng các tài liệu giúp chúng em thuận lợi hơn trong việc học tập, tìm kiếm thông tin và kiến thức về đồ án.

Chúng em xin gửi lời cám ơn đặc biệt đến Tiến sĩ Huỳnh Xuân Phụng – giảng viên môn Điện toán đám mây, đã tận tâm giúp đỡ, hỗ trợ và đồng hành cùng chúng em trong khoảng thời gian vừa qua. Bằng những tiết học bổ ích và tài liệu đầy đủ, thầy đã trang bị cho chúng em có đầy đủ kiến thức để có thể vận dụng vào đồ án của nhóm mình. Chúng em cũng xin cám ơn thầy đã tạo cơ hội cho chúng em làm đồ án này, vừa giúp chúng em có những kiến thức và thực hành với cái nền tảng công nghệ hiện đại như: AWS, Docker,.. vừa có kỹ năng làm việc nhóm trên môi trường đại học từ đó tạo cho chúng em một nền tảng vững trãi cho sự nghiệm sau này.

Do chưa có nhiều kinh nghiệm và kiến thức vẫn còn hạn chế nên đồ án của nhóm em khó tránh khỏi những thiếu xót về giao diện cũng như chức năng. Chúng em rất mong nhận được sự nhận xét, đóng góp ý kiến và phê bình từ phía thầy để đồ án của nhóm em được hoàn thiện hơn. Lời cuối chúng em xin kính chúc thầy có nhiều sức khỏe, hạnh phúc và thành công.”

**MỤC LỤC**

[PHẦN 1. MỞ ĐẦU 1](#_Toc121848598)

[1. Lý do chọn đề tài 1](#_Toc121848599)

[2. Đối tượng và phạm vi nghiên cứu 1](#_Toc121848600)

[3. Mục đích nghiên cứu 2](#_Toc121848601)

[4. Phương pháp nghiên cứu 2](#_Toc121848602)

[PHẦN 2. NỘI DUNG 3](#_Toc121848603)

[CHƯƠNG 1. CƠ SỞ LÝ THUYẾT VỀ AWS VÀ AMAZON EC2 3](#_Toc121848604)

[1. Tổng quan về Amazon Web Services 3](#_Toc121848605)

[**1.1. Khái niệm về Amazon Web Services 3**](#_Toc121848606)

[**1.2. Cơ sở hạ tầng của AWS 3**](#_Toc121848607)

[**1.3. Lợi ích của AWS 3**](#_Toc121848608)

[**1.4. Các dịch vụ, tính năng của AWS 4**](#_Toc121848609)

[2. Tổng quan về Amazon EC2 5](#_Toc121848610)

[**2.1. Khái niệm 5**](#_Toc121848611)

[**2.2. Lợi ích của Amazon EC2 5**](#_Toc121848612)

[**2.3. Các tính năng của EC2 6**](#_Toc121848613)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT VỀ DOCKER 8](#_Toc121848614)

[1. Tổng quan về Docker 8](#_Toc121848615)

[**1.1. Khái niệm về Docker 8**](#_Toc121848616)

[**1.2. Thành phần của Docker 8**](#_Toc121848617)

[2. Cách thức hoạt động của Docker 10](#_Toc121848618)

[3. Lợi ích của Docker 10](#_Toc121848619)

[CHƯƠNG 3. CHƯƠNG 3: XÂY DỰNG ỨNG DỤNG 11](#_Toc121848620)

[1. Mô tả ứng dụng 11](#_Toc121848621)

[2. Chức năng của ứng dụng 11](#_Toc121848622)

[3. Các bước triển khai 11](#_Toc121848623)

[4. Kiểm tra các chức năng: 15](#_Toc121848624)

[**4.1. Đăng nhập, đăng ký 15**](#_Toc121848625)

[**4.2. View người dùng 15**](#_Toc121848626)

[**4.3. View Admin 20**](#_Toc121848627)

[PHẦN 3. KẾT LUẬN 23](#_Toc121848628)

[1. Kết quả đạt được 23](#_Toc121848629)

[1.1 Nền tảng kiến thức 23](#_Toc121848630)

[1.2 Vận dụng 23](#_Toc121848631)

[2. Ưu điểm 23](#_Toc121848632)

[3. Nhược điểm 23](#_Toc121848633)

[4. Hướng phát triển 23](#_Toc121848634)

[TÀI LIỆU THAM KHẢO 24](#_Toc121848635)

1. MỞ ĐẦU

## **1. Lý do chọn đề tài**

Hiện nay, trong một thời đại mà công nghệ đang hiện hữu ở gần như mọi lĩnh vực trong cuộc sống của chúng ta, từ những hoạt động sinh hoạt đời thường cho đến những thiết bị hiện đại đang được sử dụng. Việc sử dụng internet đang trở nên cực kì phổ biến và tiện lợi, mang lại những lợi ích khổng lồ cho con người. Vì thế, nhu cầu sử dụng các cơ sở hạ tầng để có thể phục vụ số lượng ứng dụng, website khổng lồ như hiện nay ngày càng nâng cao. Và với sự phát triển của công nghệ, việc thay thế các cơ sở hạ tầng vật lý bằng các dịch vụ đám mây đang trở nên cực kì phổ biến và tiện lợi. Điện toán đám mây mang lại nhiều lợi ích so với cơ sở hạ tầng vật lý khi nó mang lại sự nhanh chóng, dễ dàng để sử dụng; giảm số lượng về mặt con người để quản lý; có thể dễ dàng triển khai ở bất cứ đâu trên thế giới để tăng trải nghiệm người dùng.

Một trong những công nghệ đám mây được sử dụng phổ biến nhất là dịch vụ thuê máy ảo để chạy các ứng dụng. Và vì thế, nhóm chúng em đã chọn cho mình đề tài “Sử dụng Docker để ảo hoá server Ubuntu” để làm đồ án cho môn học Điện toán đám mây. Nhằm sử dụng các công nghệ hiện đại như Docker trên nền máy chủ Ubuntu để có thể ảo hoá ứng dụng thành các máy tính ảo cho người dùng.

Việc sử dụng các cơ sở hạ tầng trên đám mây giúp tiết kiệm một lượng nhân công, chi phí lớn cho việc duy trì hoạt động khi so với các biện pháp thông thường. Việc sử dụng như một phần mềm còn mang lại sự bảo mật đáng kể so với cơ sở hạ tầng vật lý và mang lại cơ số lợi ích khác

## **2. Đối tượng và phạm vi nghiên cứu**

Đối với đề tài được chọn, các thành viên trong nhóm sẽ tập trung nghiên cứu và tìm hiểu các công nghệ sau:

* Công nghệ ảo hoá Docker: tìm hiểu về khái niệm, cơ chế hoạt động và cách sử dụng, áp dụng vào thực tế. Từ đó ảo hoá thành các máy ảo cho người dung sử dụng
* Dịch vụ Amazon Elastic Compute Cloud (Amazon EC2): tìm hiểu về khái niệm, phương thức hoạt động và các tính năng đi kèm với dịch vụ

## **3. Mục đích nghiên cứu**

Việc lựa chọn các đối tượng trên để nghiên cứu để cung cấp một giao diện website để người dùng có thể sử dụng để tạo ra các Instance và cung cấp các tính năng mà một nền tảng đám mây thường có như tạo Snapshot, Images, Key và Network mà không cần quan tâm đến cơ sở hạ tầng

## **4. Phương pháp nghiên cứu**

Sử dụng các kiến thức đã học được trong môn học kết hợp với việc tìm hiểu các tài liệu, giáo trình về điện toán đám mây nói chung và dịch vụ EC2 nói riêng. Sử dụng các nguồn tài liệu về dịch vụ ảo hoá Docker để phục vụ cho cho đề tài. Bên cạnh đó, chúng em nghiên cứu các tính năng đi kèm của Docker để nhằm phục vụ cho mục đích mở rộng đề tài

1. NỘI DUNG
   1. CƠ SỞ LÝ THUYẾT VỀ AWS VÀ AMAZON EC2
      1. Tổng quan về Amazon Web Services
         1. Khái niệm về Amazon Web Services

Amazon Web Services (AWS) là nền tảng đám mây toàn diện và được sử dụng rộng rãi nhất, cung cấp trên 200 dịch vụ đầy đủ tính năng từ các trung tâm dữ liệu trên toàn thế giới. Hàng triệu khách hàng—bao gồm các công ty khởi nghiệp tăng trưởng nhanh nhất, các tập đoàn lớn nhất cũng như các cơ quan hàng đầu của chính phủ đều tin tưởng vào AWS để giảm chi phí, trở nên linh hoạt hơn và đổi mới nhanh hơn.

* + - 1. Cơ sở hạ tầng của AWS

Đám mây AWS trải rộng trên 96 Vùng sẵn sàng tại 30 khu vực địa lý trên khắp thế giới và đã công bố kế hoạch tăng thêm 15 Vùng sẵn sàng và 5 Khu vực AWS khác tại Úc, Canada, Israel, New Zealand và Thái Lan. Với cơ sở hạ tầng khổng lồ trên, AWS cung cấp: 25 Khu vực địa phương, 29 Vùng Wavelength dành cho các ứng dụng có độ trễ siêu thấp; Phục vụ 245 quốc gia và vùng lãnh thổ; 115 Vị trí Direct Connect

* + - 1. Lợi ích của AWS

**- Bảo mật**: Tại AWS, việc bảo mật bắt đầu từ cơ sở hạ tầng cốt lõi. Được xây dựng riêng cho nền tảng đám mây và có thiết kế nhằm đáp ứng các yêu cầu bảo mật nghiêm ngặt nhất trên thế giới, cơ sở hạ tầng được giám sát 24/7 nhằm giúp đảm bảo tính bảo mật, toàn vẹn và sẵn sàng cho dữ liệu

**- Mức độ sẵn sàng**: Mỗi khu vực đều được tách biệt hoàn toàn và gồm nhiều Vùng sẵn sàng, là những phân vùng được tách biệt hoàn toàn trong cơ sở hạ tầng. Để cô lập mọi sự cố và đạt được mức độ sẵn sàng cao hơn, có thể phân chia các ứng dụng trên nhiều Vùng sẵn sàng tại cùng khu vực. Ngoài ra, tầng điều khiển và bảng điều khiển quản lý của AWS được phân phối trên nhiều khu vực và bao gồm các điểm cuối API tại khu vực.

- **Hiệu năng:** Cơ sở hạ tầng toàn cầu AWS được xây dựng hướng đến hiệu năng. Khu vực AWS cung cấp độ trễ thấp, tỷ lệ mất gói thấp và chất lượng mạng tổng thể cao. Bất kể ứng dụng là gì, ta có thể nhanh chóng khởi chạy tài nguyên khi cần, triển khai hàng trăm hay thậm chí hàng nghìn máy chủ trong vài phút.

- **Khả năng điều chỉnh quy mô**: Cơ sở hạ tầng toàn cầu của AWS cho phép các công ty trở nên cực kỳ linh hoạt và có thể tận dụng khả năng mở rộng vô hạn về mặt lý thuyết của đám mây. Có thể nhanh chóng khởi chạy tài nguyên khi cần, triển khai hàng trăm thậm chí hàng nghìn máy chủ trong vài phút.

**- Sự linh hoạt**: Cơ sở hạ tầng toàn cầu AWS mang đến sự linh hoạt trong việc lựa chọn cách thức và nơi ta muốn chạy khối lượng công việc của mình, cũng như thời điểm sử dụng cùng một mạng, mặt phẳng điều khiển, API và các dịch vụ AWS. Nếu bạn muốn chạy các ứng dụng của mình trên toàn cầu, ta có thể chọn từ Khu vực AWS và AZ bất kỳ.

**- Phạm vi toàn cầu:** Khi triển khai các ứng dụng và khối lượng công việc lên đám mây, ta có thể linh hoạt trong việc lựa chọn cơ sở hạ tầng công nghệ gần nhất với người dùng mục tiêu chính.

* + - 1. Các dịch vụ, tính năng của AWS
* AWS EC2 – dịch vụ máy ảo
* AWS S3 – dịch vụ lưu trữ
* AWS RDS - dịch vụ SQL Database
* AWS DynamoDB - NoSQL database
* AWS VPC—Private network
* AWS IAM – quản lý quyền truy cập đến tài nguyên

Và rất nhiều những dịch vụ phổ biến khác

* + 1. Tổng quan về Amazon EC2
       1. Khái niệm

Amazon Elastic Compute Cloud (Amazon EC2) là một cơ sở hạ tầng điện toán đám mây được cung cấp bởi Amazon Web Services (AWS) giúp cung cấp tài nguyên máy tính ảo hoá theo yêu cầu.

Amazon EC2 cung cấp các ứng dụng máy tính ảo hoá có thể mở rộng về khả năng xử lý cùng các thành phần phần cứng ảo như bộ nhớ máy tính (ram), vi xử lý, linh hoạt trong việc lựa chọn các phân vùng lưu trữ dữ liệu ở các nền tảng khác nhau và sự an toàn trong quản lý dịch vụ bởi kiến trúc ảo hoá đám mây mạnh mẽ của AWS.

Amazon EC2 sẽ cung cấp một hoặc máy chủ ảo có thể kết hợp với nhau để dễ dàng triển khai ứng dụng nhanh nhất và đảm bảo tính sẵn sàng cao nhất. Thậm chí về mặt thanh toán bạn dễ dàng biết được các mức chi phí cần thanh toán dựa trên thông tin tài nguyên bạn sử dụng.

Amazon EC2 Instanc là một cloud server. Với một tài khoản bạn có thể tạo và sử dụng nhiều Amazon EC2 Instance. Các Amazon EC2 Instance được chạy trên cùng một server vật lý và chia sẻ memory, CPU, ổ cứng...

* + - 1. Lợi ích của Amazon EC2

**Đa địa điểm:** Amazon EC2 đem đến khả năng đặt phiên bản ở nhiều địa điểm. Các địa điểm Amazon EC2 bao gồm Khu vực và Vùng sẵn sàng. Vùng sẵn sàng là các địa điểm riêng biệt được thiết kế xây dựng chống lỗi trong các Vùng sẵn sàng và có khả năng kết nối ít tốn kém và độ trễ thấp đến các Vùng sẵn sàng khác trong cùng Khu vực

**Lựa chọn hệ điều hành và phần mềm**: Amazon Machine Images (AMIs) được cấu hình sẵn với danh sách hệ điều hành ngày càng lớn, bao gồm các bản phân phối Microsoft Windows và Linux như Amazon Linux 2, Ubuntu, Red Hat Enterprise Linux, CentOS, SUSE và Debian.

**Tối ưu hóa chi phí và dung lượng**: Với mô hình tính phí theo giây, ta chỉ phải trả tiền cho những gì bạn sử dụng

**Amazon EC2 Auto Scaling** cho phép ta tự động thay đổi quy mô dung lượng Amazon EC2 theo hướng tăng lên hay giảm xuống theo các điều kiện do chỉ định. Bạn có thể sử dụng các chính sách dynamic scaling (thay đổi quy mô linh động) và predictive scaling (thay đổi quy mô dự đoán) của EC2 Auto Scaling

**Lưu trữ:** Yêu cầu lưu trữ của các khối lượng công việc Amazon EC2 khác nhau có thể sẽ rất lớn. Không chỉ cung cấp dung lượng lưu trữ phiên bản tích hợp, AWS còn cung cấp Amazon Elastic Block Store (Amazon EBS) và Amazon Elastic File System (Amazon EFS) thích hợp với các yêu cầu khối lượng công việc lưu trữ đám mây khác.

**Kết nối mạng:** Hiệu suất gói cao trên giây và độ trễ thấp với Enhanced Networking. Enhanced Networking cho phép ta đạt hiệu năng gói mỗi giây (PPS) cao hơn đáng kể, giảm độ biến động mạng và độ trễ cũng thấp hơn.

**Bảo trì:** AWS thường xuyên bảo trì phần cứng, phần mềm, nguồn và mạng định kỳ với thời gian gián đoạn tối thiểu trên tất cả các loại phiên bản EC2. Các công nghệ bảo trì không can thiệp như cập nhật trực tiếp và di chuyển trực tiếp không đòi hỏi các phiên bản phải dừng hoạt động hoặc khởi động lại.

* + - 1. Các tính năng của EC2
* Môi trường máy chủ tính toán ảo hoá (virtual computing) có tên gọi là **Instance**
* Các mẫu được định nghĩa trước cho instances, được biết như Amazon Machine Images (AMIs), cung cấp các gói phần mềm cần thiết cho máy chủ (gồm hệ điều hành và các phần mềm)
* Cung cấp các cấu hình khác nhau dành cho máy chủ ảo như: CPU, Memory, ổ cứng lưu trữ và mạng internet dành cho instance
* Bảo mật thông tin đăng nhập vào máy chủ ảo instance bằng cách sử dụng cơ chế key pairs (AWS lưu trữ thông tin public key và ta chỉ cần giữ thông tin private key)
* Tách biệt nguồn tài nguyên như instances và Amazon EBS volumes trên các vùng lưu trữ vật lý ở nhiều vị trí khu vực, được biết đến như Regions và Availability Zones
* Các phân vùng ổ cứng (volume) dành cho lưu trữ dữ liệu tạm thời sẽ bị xoá khi mà bạn dừng hoặc huỷ các máy chủ ảo instance – instance store volume
* Các phần vùng ổ cứng dành cho lưu trữ dữ liệu mang tính lâu dài thì có dịch vụ Amazon Elastic Block Store (Amazon EBS)
* Dịch vụ tường lửa cho phép chỉ định loại giao thức, port, source ip range,… được phép truy xuất đến máy chủ ảo instance. Dịch vụ được biết đến với tên gọi là “security group”
* Địa chỉ IPv4 tĩnh cho hệ thống – Elastic IP Address
* Các thông tin metadata, được biết đến với khái niệm “tags”, ta có thể sử dụng để gán thông tin cho các Amazon EC2 Resources
* Hệ thống mạng ảo AWS cung cấp giúp ta có thể cô lập các máy chủ instance với nhau, kết nối riêng với nhau theo ý muốn. Được gọi là “Virtual Private Clouds (VPCs)”.
* Amazon CloudWatch: giúp quan sát các tài nguyên đang hoạt động trên hệ thống
  1. CƠ SỞ LÝ THUYẾT VỀ DOCKER
     1. Tổng quan về Docker
        1. Khái niệm về Docker

Docker là nền tảng phần mềm cho phép ta dựng, kiểm thử và triển khai ứng dụng một cách nhanh chóng. Docker đóng gói phần mềm vào các đơn vị tiêu chuẩn hóa được gọi là container có mọi thứ mà phần mềm cần để chạy, trong đó có thư viện, công cụ hệ thống, mã và thời gian chạy. Bằng cách sử dụng Docker, ta có thể nhanh chóng triển khai và thay đổi quy mô ứng dụng vào bất kỳ môi trường nào và biết chắc rằng mã sẽ chạy được.

Việc chạy Docker trên AWS đem đến cho các nhà phát triển và quản trị viên một phương thức dựng, vận chuyển và chạy ứng dụng phân phối ở quy mô bất kỳ có chi phí thấp và độ tin cậy cao.

* + - 1. Thành phần của Docker
         1. Docker Client

Docker client là cách mà người dùng tương tác với docker thông qua command line. Khi thực hiện thì docker sẽ gọi đến Docker Daemon thông qua các API.

* + - * 1. Docker Daemon

Docker daemon là server Docker cho yêu cầu từ Docker API. Nó quản lý images, containers, networks và volume.

* + - * 1. Docker Registry

Đây là nơi lưu trữ riêng của Docker Images. Images được push vào registry và client sẽ pull images từ registry. Có thể sử dụng registry của riêng bạn hoặc registry của nhà cung cấp như : AWS, Google Cloud, Microsoft Azure,…

* + - * 1. Docker Hub

Đây là Registry lớn nhất của Docker Images ( mặc định). Có thể tìm thấy images và lưu trữ images của riêng bạn trên Docker Hub ( miễn phí).

* + - * 1. Docker Networking

Tính năng cho phép kết nối các container lại với nhau. Kết nối này có thể trên 1 host hoặc nhiều host.

* + - * 1. Docker Compose:

Là công cụ cho phép run app với nhiều Docker containers 1 cách dễ dàng hơn. Docker Compose cho phép bạn config các command trong file docker-compose.yml để sử dụng lại. Có sẵn khi cài Docker.

* + - * 1. Docker Swarm:

Đây là công cụ giúp phối hợp và triển khai các container trên các host sử dụng Docker khác nhau

* + - * 1. Docker Services:

Là các containers trong production. 1 service chỉ run 1 image nhưng nó mã hoá cách thức để run image — sử dụng port nào, bao nhiêu bản sao container run để service có hiệu năng cần thiết và ngay lập tức.

* + - * 1. Docker Volume

Docker volumes giúp lưu trữ và chia sẽ dữ liệu. Khi sẻ dụng docker volumes thì dữ liệu trong docker sẽ k bị biến mất khi ta xóa container mà sẽ được lưu vào một nơi cụ thể.

* + - * 1. Docker Objects

Docker Image: Docker image là một file bất biến - không thay đổi, chứa các source code, libraries, dependencies, tools và các files khác cần thiết cho một ứng dụng để chạy.

Docker Container: Docker container là một run-time environment mà ở đó người dùng có thể chạy một ứng dụng độc lập. Những container này rất gọn nhẹ và cho phép bạn chạy ứng dụng trong đó rất nhanh chóng và dễ dàng.

* + 1. Diagram

       Description automatically generatedCách thức hoạt động của Docker

Docker hoạt động bằng cách cung cấp phương thức tiêu chuẩn để chạy code của người dùng. Docker là hệ điều hành dành cho container. Cũng tương tự như cách máy ảo ảo hóa (loại bỏ nhu cầu quản lý trực tiếp) phần cứng máy chủ, các container sẽ ảo hóa hệ điều hành của máy chủ. Docker được cài đặt trên từng máy chủ và cung cấp các lệnh đơn giản mà bạn có thể sử dụng để dựng, khởi động hoặc dừng container.

* + 1. Lợi ích của Docker
* Docker start và stop chỉ trong vài giây.
* Có thể khởi chạy container trên mỗi hệ thống mà ta muốn.
* Container có thể build và loại bỏ nhanh hơn máy ảo.
* Dễ dàng thiết lập môi trường làm việc. Chỉ cần config 1 lần duy nhất và không bao giờ phải cài đặt lại các dependencies. Nếu bạn thay đổi máy hoặc có người mới tham gia vào project thì bạn chỉ cần lấy config đó và đưa cho họ.
* Nó giữ cho word-space của bạn sạch sẽ hơn khi bạn xóa môi trường mà ảnh hưởng đến các phần khác.
  1. XÂY DỰNG ỨNG DỤNG
     1. Mô tả ứng dụng

Xây dựng một website giúp người dùng có thể quản lý các instance cùng các tính năng đi kèm như Snapshot, SSH Key, Network và Images. Giao diện giúp admin có thể quản lý toàn bộ người dùng.

Công nghệ được sử dụng trong ứng dụng

* Java Servlet chạy trên nền Tomcat Server 9.0
* Library: MySQL Connector, SQL2o, JSCH
* Database: MySQL
* Giao diện web: HTML, CSS, JS
* Docker dùng để ảo hoá server
  + 1. Chức năng của ứng dụng
* Người dùng có thể tạo, khởi động, dừng và xoá instance
* Khi tạo instance, người dùng có thể điều chỉnh các thông số: tên instance, CPUs, memory, hệ điều hành, SSH, Network, Server, protection và userdata
* Instance khi bật protection không thể bị xoá
* Người dùng có thể kết nối đến instance bằng SSH Key hoặc SSH Password
* Người dùng có thể tạo, xoá và restore snapshot từ instance
* Người dùng có thể tạo, xoá image từ instance. Có thể tạo một instance mới từ image đã được tạo
* Người dùng có thể thêm, xoá network phục vụ các instance
* Admin có thể quản lý mọi thứ của mọi người dùng. Admin có thể thêm, xoá, sửa người dùng
* Admin có thể thêm server mới vào hệ thống cho người dùng sử dụng
  + 1. Các bước triển khai
* Bước 1: Tạo các instance Ubuntu trên hệ thống AWS EC2. Chọn một máy để làm host chính cho website và database. Các máy còn lại sẽ dùng làm Worker

Graphical user interface, application

Description automatically generated

* Bước 2: Tinh chỉnh Security cho máy Host có thể kết nối SSH và kết nối với các máy Worker còn lại

A screenshot of a computer

Description automatically generated

* Bước 3: Cấp Elastic IP cho các máy (Có thể cấp hoặc không)

Graphical user interface, text, application, email

Description automatically generated

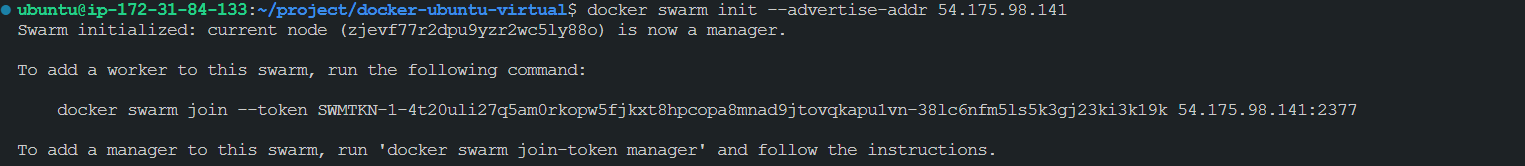
* Bước 4: Tiến hành SSH vào máy Host và các máy Worker để cài các ứng dụng cần thiết như Docker và các thành phần. Bật experimental để hỗ trợ tính năng snapshot.

Text

Description automatically generated Text

Description automatically generated

* Bước 5: Khởi động Docker Swarm trên máy host để các máy Worker có thể connect vào



* Bước 6: Chỉnh sửa file Config và tiến hành build file jar cho webserver

Text

Description automatically generatedText

Description automatically generated

* Bước 7: Tiến hành build với Dockerfile bên dưới để tạo image chạy website (Website chạy trên nền tomcat 9.0)

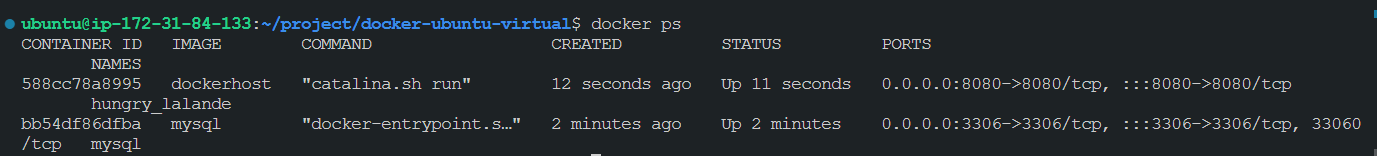
Text

Description automatically generated

Text

Description automatically generated

* Bước 8: Tiến hành chạy Database và thêm dữ liệu vào với lệnh
  + docker run --name mysql -e MYSQL\_ROOT\_PASSWORD=root -e MYSQL\_DATABASE=ubuntudockercloud -e MYSQL\_PASSWORD=root --publish 3306:3306 -d --network hostnetwork mysql
  + cat ubuntudockercloud.sql | docker exec -i mysql /usr/bin/mysql -u root --password=root ubuntudockercloud
* Bước 9: Khởi chạy website với lệnh:
  + docker run -p 8080:8080 --network hostnetwork dockerhost



* Bước 10: Tiến hành kiểm tra web trên đường dẫn: http://54.175.98.141:8080/DockerUbuntuVirtual/

Graphical user interface, website

Description automatically generated

* + 1. Kiểm tra các chức năng:
       1. Đăng nhập, đăng ký

Graphical user interface, application

Description automatically generatedGraphical user interface

Description automatically generated

* + - 1. View người dùng

Graphical user interface, text, application

Description automatically generated

* + - * 1. Tạo SSH Key

Graphical user interface, text, application, email

Description automatically generated

Trên máy Host:

A screenshot of a computer

Description automatically generated with medium confidence

* + - * 1. Tạo network

Graphical user interface, application

Description automatically generatedGraphical user interface, text, application

Description automatically generated

* Kiểm tra trên host

A screenshot of a computer

Description automatically generated with medium confidence

* + - * 1. Tạo Instance

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface

Description automatically generated with medium confidence

* Kiểm tra trên host

A computer screen capture

Description automatically generated with medium confidence

* + - * 1. SSH tới Instance vừa tạo

Text

Description automatically generated

* + - * 1. Tạo Snapshot cho instance vừa tạo

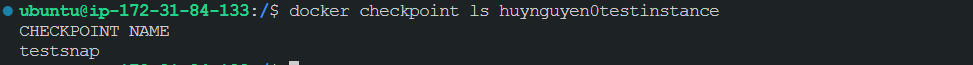
Graphical user interface, application, Word

Description automatically generated

Graphical user interface, application

Description automatically generated

* Kiểm tra trên host



* + - * 1. Tạo image cho instance vừa tạo

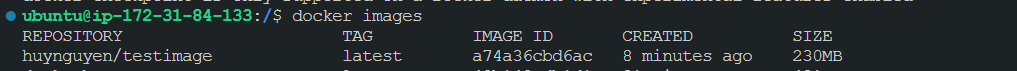
Graphical user interface, application

Description automatically generated

Graphical user interface, text, application

Description automatically generated

* Kiểm tra trên host:



* + - * 1. Tạo instance từ image vừa tạo được

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

* Kiểm tra trên host

A computer screen capture

Description automatically generated with medium confidence

* + - 1. View Admin

Graphical user interface, application

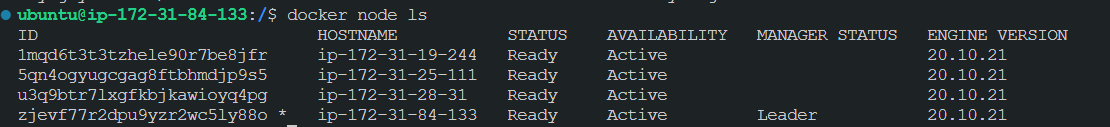
Description automatically generated

* + - * 1. Thêm các worker server vào hệ thống

Graphical user interface, application

Description automatically generated

* Kiểm tra trên host



* + - * 1. Quản lý User

Graphical user interface

Description automatically generated with low confidence

* Chỉnh sửa User

Graphical user interface, application

Description automatically generated

* Tạo user mới

Graphical user interface, application, Word

Description automatically generated

1. KẾT LUẬN

## **1. Kết quả đạt được**

Sau khi hoàn thành đề tài “Sử dụng Docker ảo hoá server Ubuntu” từ các kiến thức đã được học trong môn học cùng với đề tài của các anh chị khoá trước và tìm hiểu, nghiên cứu thêm từ các nguồn trên internet, nhóm đã đạt được những kết quả như sau:

### **1.1 Nền tảng kiến thức**

Nhìn chung, sau khi nghiên cứu, tìm hiểu, các thành viên trong nhóm đã hiểu hơn về AWS nói chung cũng như AWS EC2 và các tính năng đi kèm. Hiểu thêm về các lợi ích mà các nền tảng điện toán đám mây hiện nay mang lại và học cách vận dụng. Cùng với đó, việc sử dụng Docker giúp nhóm chúng em hiểu hơn về công nghệ ảo hoá hiện đại và phổ biến nhất hiện nay. Docker mang lại lợi ích rất lớn cho việc ảo hoá hiện nay

### **1.2 Vận dụng**

Về việc vận dụng, nhóm đã xây dựng được một website có thể truy cập từ người dùng và Admin đi kèm với một số tính năng cơ bản được cung cấp như: thao tác với instance, tạo key, tạo network,… Việc vận dụng này cũng giúp nhóm hiểu hơn và sử dụng các tính năng của EC2 thành thạo hơn trong tương lai cũng như tránh các sai sót đã gặp phải trong quá trình làm đồ án.

## **2. Ưu điểm**

Đề tài đã có thể mang lại những chức năng theo yêu cầu hoạt động khá ổn định từ phía người dùng và admin một cách cơ bản với giao diện khá bắt mắt, hiện đại so với các website hiện nay

## **3. Nhược điểm**

* Đôi khi, website hoạt động một cách thiếu ổn định
* Chức năng SSH Key hoạt động chưa ổn định
* Cần bổ sung một số tính năng cơ bản khác như Volume,…

## **4. Hướng phát triển**

Đề tài có thể dùng để phát triển thêm một số tính năng mà Docker cung cấp như tinh chỉnh ip cho network, thêm volume cho mỗi người dùng, tinh chỉnh firewall cho các instance trên hệ thống. Cùng với đó, nâng cấp công nghệ website mới hơn so với Java Servlet

# TÀI LIỆU THAM KHẢO

* <https://docs.docker.com/>
* <https://phoenixnap.com/kb/how-to-ssh-into-docker-container>
* <https://thenewstack.io/how-to-enable-docker-experimental-features-and-encrypt-your-login-credentials/>
* <https://github.com/trancongtruong1234/Remote-docker-controller>
* Và các nguồn khác từ internet