

|  |
| --- |
| **KHOA ĐÀO TẠO CHẤT LƯỢNG CAO**  **BỘ MÔN ĐIỆN TOÁN ĐÁM MÂY**  **----------** |
| **BÁO CÁO CUỐI KỲ**  **\*\*\***  **VIẾT ỨNG DỤNG KẾT HỢP NHIỀU DOCKER VỚI NHAU**    **MÃ MÔN HỌC: CLCO332779**  **THỰC HIỆN: NHÓM 42**  **LỚP: 05 CLC**  **GVHD: TS. Huỳnh Xuân Phụng**  **Tp. Hồ Chí Minh, tháng 12 năm 2022** |

**MỤC LỤC**

[**MỞ ĐẦU** 3](#_Toc122769988)

[**1.** **Đặt vấn đề** 3](#_Toc122769989)

[**2.** **Giải pháp và mục tiêu** 3](#_Toc122769990)

[**3.** **Phạm vi đề tài** 3](#_Toc122769991)

[**NỘI DUNG** 3](#_Toc122769992)

[**1.** **Cơ sở lý thuyết** 3](#_Toc122769993)

[a. Phân tích và thiết kế hệ thống thông tin 3](#_Toc122769994)

[b. Ngôn ngữ lập trình 3](#_Toc122769995)

[**2.** **Phân chia công việc** 5](#_Toc122769996)

[**3.** **Giới thiệu chương trình quản lý sinh viên** 6](#_Toc122769997)

[a. Trang chủ 6](#_Toc122769998)

[b. Sinh viên 6](#_Toc122769999)

[c. Chương trình đào tạo 7](#_Toc122770000)

[d. Môn học 9](#_Toc122770001)

[e. Lớp học phần 10](#_Toc122770002)

[f. Tham gia học 11](#_Toc122770003)

[**4.** **Triển khai chương trình trên dịch vụ EC2 của AWS** 12](#_Toc122770004)

[**5.** **Kết luận** 20](#_Toc122770005)

[a. Ưu điểm của chương trình: 20](#_Toc122770006)

[b. Hạn chế của chương trình: 20](#_Toc122770007)

[c. Ý tưởng phát triển chương trình: 21](#_Toc122770008)

**DANH MỤC HÌNH ẢNH**

[Hình 1 - Giao diện trang chủ 8](https://d.docs.live.net/4951f0d6c4070b2d/Desktop/Nhom42%20(1)%20(1).docx#_Toc122770015)

[Hình 2 - Giao diện trang danh sách sinh viên (1) 8](https://d.docs.live.net/4951f0d6c4070b2d/Desktop/Nhom42%20(1)%20(1).docx#_Toc122770016)

[Hình 3 - Giao diện trang danh sách sinh viên (2) 9](#_Toc122770017)

[Hình 4 - Giao diện trang danh sách sinh viên (3) 9](#_Toc122770018)

[Hình 5 - Giao diện chương trình đào tạo (1) 10](#_Toc122770019)

[Hình 6 - Giao diện chương trình đào tạo (2) 10](#_Toc122770020)

[Hình 7 - Giao diện chương trình đào tạo (3) 10](#_Toc122770021)

[Hình 8 - Giao diện trang môn học (1) 11](#_Toc122770022)

[Hình 9 - Giao diện trang môn học (2) 11](#_Toc122770023)

[Hình 10 - Giao diện trang môn học (3) 12](#_Toc122770024)

[Hình 11 - Giao diện trang lớp học phần (1) 12](#_Toc122770025)

[Hình 12 - Giao diện trang lớp học phần (2) 13](#_Toc122770026)

[Hình 13 - Giao diện trang lớp học phần (3) 13](#_Toc122770027)

[Hình 14 - Giao diện trang tham gia học (1) 14](#_Toc122770028)

[Hình 15 - Giao diện trang tham gia học (2) 14](#_Toc122770029)

[Hình 16 - Giao diện trang tham giam học (3) 14](#_Toc122770030)

[Hình 17 - Nút Start Lab 15](#_Toc122770031)

[Hình 18 - Giao diện sau khi Start Lab 15](#_Toc122770032)

[Hình 19 - Giao diện EC2 15](#_Toc122770033)

[Hình 20 - Giao diện cấu hình EC2 16](#_Toc122770034)

[Hình 21 - Tạo Keypair (1) 16](#_Toc122770035)

[Hình 22 - Tạo Keypair (2) 17](#_Toc122770036)

[Hình 23 - Chọn Subnet và đặt tên Security Group 17](#_Toc122770037)

[Hình 24 - Cấu hình các Rule cho Inbound Security Group 18](#_Toc122770038)

[Hình 25 - Tạo EC2 thành công 18](#_Toc122770039)

[Hình 26 - Connect đến EC2 (1) 18](https://d.docs.live.net/4951f0d6c4070b2d/Desktop/Nhom42%20(1)%20(1).docx#_Toc122770040)

[Hình 27 - Connect đến EC2 (2) 19](#_Toc122770041)

[Hình 28 - Cài đặt git (1) 19](#_Toc122770042)

[Hình 29 - Cài đặt git (2) 20](#_Toc122770043)

[Hình 30 - Docker Info (1) 20](#_Toc122770044)

[Hình 31 - Docker Info (2) 21](#_Toc122770045)

[Hình 32 - Cài đặt Docker Compose 21](#_Toc122770046)

[Hình 33 - Tạo các container - chạy chương trình 22](#_Toc122770047)

**DANH SÁCH SINH VIÊN THỰC HIỆN**

*HỌC KỲ I NĂM HỌC 2022-2023*

*Nhóm: 42* ***( Lớp 05 CLC)***

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **HỌ VÀ TÊN SINH VIÊN** | **MÃ SỐ SINH VIÊN** | **TỈ LỆ %**  **ĐÓNG GÓP** |
| 1 | Huỳnh Thanh Tuấn | 20110120 | 100% |
| 2 | Phan Hồng Sơn | 20110560 | 100% |
| 3 | KHAMSAVATH Bounmy | 20110L02 | 100% |

***Nhận xét của giáo viên:***

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

…………………………………………………………………………………….

*Tp. Hồ Chí Minh, ngày ….., tháng 12, năm 2022*

*Giáo viên chấm điểm (Ký và ghi họ tên)*

*Huỳnh Xuân Phụng*

# **MỞ ĐẦU**

1. **Đặt vấn đề**

Khi thực hiện chạy một chương trình mới trên một máy tính khác cần phải thực hiện cài đặt các gói thư viện, ngôn ngữ lập trình, … cũng như là phiên bản phù hợp đúng với cấu hình đề nghị, để chương trình đó có thể chạy và hoạt động ổn định trên máy tính đó. Các bước cấu hình và cài đặt này quá rắc rối và tốn nhiều thời gian.

1. **Giải pháp và mục tiêu**

Sử dụng docker để có thể tiết kiệm thời gian cấu hình và cài đặt.

Thực hiện chia chương trình chạy trên các container khác nhau và liên kết chúng qua một mạng. Để khi có cập nhật thay đổi phần nào thì chỉ cẩn build lại container của phần đó. Giúp tiết kiệm thời gian và tránh được việc build lại những phần không thay đổi.

1. **Phạm vi đề tài**

Xây dựng chương trình quản lý sinh viên chạy trên nhiều container và triển khai trên dịch vụ EC2 của Amazon Web Services.

# **NỘI DUNG**

1. **Cơ sở lý thuyết**
   1. **Phân tích và thiết kế hệ thống thông tin**

Hệ thống là tập hợp các phân tử có những mội quan hệ ràng buộc lẫn nhau cùng hoạt động chung cho một số mực tiêu nào đó.

Hệ thống quản lý là một hệ thống có một mịc đích mang lại lợi nhuận hoặc lợi ích nào đó. Đặc điểm của hệ thống là có sự tham gia của con người và có trao đổi thông tin.

Hệ thống thông tin (information system) là một hệ thống sử dụng công nghệ thông tin để thu thập, truyền, lưu trữ, xủ lý và biểu diễn thông tin trong một hay nhiều qúa trình kinh doanh .

* 1. **Ngôn ngữ lập trình**

**Ngôn ngữ C#:**

C# (hay C sharp) là một ngôn ngữ lập trình đơn giản, được phát triển bởi đội ngũ kỹ sư của Microsoft vào năm 2000. C# là ngôn ngữ lập trình hiện đại, hướng đối tượng và được xây dựng trên nền tảng của hai ngôn ngữ mạnh nhất là C++ và Java.

Trong các ứng dụng Windows truyền thống, mã nguồn chương trình được biên dịch trực tiếp thành mã thực thi của hệ điều hành. Trong các ứng dụng sử dụng .NET Framework, mã nguồn chương trình (C#, VB.NET) được biên dịch thành mã ngôn ngữ trung gian MSIL (Microsoft intermediate language).

Sau đó mã này được biên dịch bởi Common Language Runtime (CLR) để trở thành mã thực thi của hệ điều hành. Hình bên dưới thể hiện quá trình chuyển đổi MSIL code thành native code.

**Java Spring Boot:**

Spring Boot là một dự án phát triển bởi JAV (ngôn ngữ java) trong hệ sinh thái Spring framework. Nó giúp cho các lập trình viên chúng ta đơn giản hóa quá trình lập trình một ứng dụng với Spring, chỉ tập trung vào việc phát triển business cho ứng dụng

Để phát triển một ứng dụng web cơ bản HelloWorld sử dụng Spring framework bạn sẽ cần ít nhất 5 công đoạn sau:

Tạo một project sử dụng Maven với các dependency cần thiết của Spring MVC và Servlet API.

Một tập tin web.xml để khai báo DispatcherServlet của Spring MVC.

Một tập tin cấu hình của Spring MVC.

Một class Controller trả về một trang “Hello World” khi có request đến

Cuối cùng là phải có một web server dùng để triển khai ứng dụng lên chạy.

Trong các công đoạn này, chỉ có công đoạn tạo một class Controller thì có thể khác cho các ứng dụng khác nhau vì mỗi ứng dụng có một yêu cầu khác nhau. Còn các công đoạn khác thì như nhau.

Giờ đây với Spring Boot, chúng ta có thể tạo dự án Spring một cách nhanh chóng và cấu hình cũng đơn giản dùng Sublime Text để phát triển luôn khỏi cần cài đặt eclipse hay netbean nặng bỏ bà. Dưới đây là một số tính năng nổi bật của Spring Boot:

Tạo các ứng dụng Spring độc lập

Nhúng trực tiếp Tomcat, Jetty hoặc Undertow (không cần phải deploy ra file WAR)

Các starter dependency giúp việc cấu hình Maven đơn giản hơn

Tự động cấu hình Spring khi cần thiết

Không sinh code cấu hình và không yêu cầu phải cấu hình bằng XML

1. **Phân chia công việc**

|  |  |  |
| --- | --- | --- |
| **Họ và Tên** | **MSSV** | **Nhiệm Vụ** |
| Huỳnh Thanh Tuấn | 20110120 | Làm API bảng sinhvien và chuongtrinhdaotao  Tạo docker compose  Tạo dockerfile backend  Tạo Database |
| Phan Hồng Sơn | 20110560 | Làm API bảng monhoc, lophoc, thamgiahoc |
| KHAMSAVATH Bounmy | 20110L02 | -Làm Phần frontend  - Tạo dockerfile frontend |

1. **Giới thiệu chương trình quản lý sinh viên**
   1. **Trang chủ**

A screenshot of a computer

Description automatically generated with low confidenceChương trình sẽ có một trang chủ để hiển thị khi người dùng vào ứng dụng website

Hình 1 - Giao diện trang chủ

* 1. **Sinh viên**

Trang danh sách sinh viên gồm có chức năng thêm, sửa và xoá. Dùng để quản lý thông tinh các sinh viên như Mã sinh viên, Họ và tên, Năm học, Ngày sinh, Năm nhập học, Giới tính, Chương trình đào tạo của sinh viên.

Table

Description automatically generated

Hình 2 - Giao diện trang danh sách sinh viên (1)

Graphical user interface, text, application, email

Description automatically generated

Hình 3 - Giao diện trang danh sách sinh viên (2)

Graphical user interface, text, application, email

Description automatically generated

Hình 4 - Giao diện trang danh sách sinh viên (3)

### **Chương trình đào tạo**

Trang chương trình đào tạo gồm có chức năng thêm, sửa và xoá, dùng để quản lý các chương trình đào tạo trong một trường đại học, bao gồm các thông tin như Mã chương trình đào tạo, Tên chương trình đào tạo, Số tín chỉ, Năm bắt đầu học.

**Table

Description automatically generated**

Hình 5 - Giao diện chương trình đào tạo (1)

**Graphical user interface, text, application

Description automatically generated**

Hình 6 - Giao diện chương trình đào tạo (2)

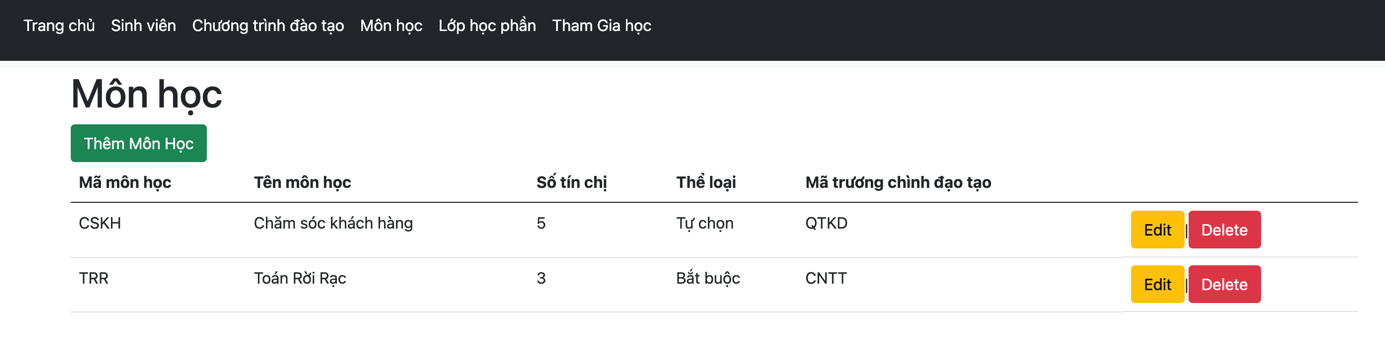
**Graphical user interface, application

Description automatically generated**

Hình 7 - Giao diện chương trình đào tạo (3)

### **Môn học**

Trang mộn học gồm có chức năng thêm, sửa và xoá, dùng để quản lý các môn học trong các hoc phần, bao gồm các thông tin như Mã môn học, Tên môn học, Số tín chỉ, Thể loại, Mã chương trình đào tạo.

****

Hình 8 - Giao diện trang môn học (1)

**Graphical user interface, text, application

Description automatically generated**

Hình 9 - Giao diện trang môn học (2)

**Graphical user interface, text, application

Description automatically generated**

Hình 10 - Giao diện trang môn học (3)

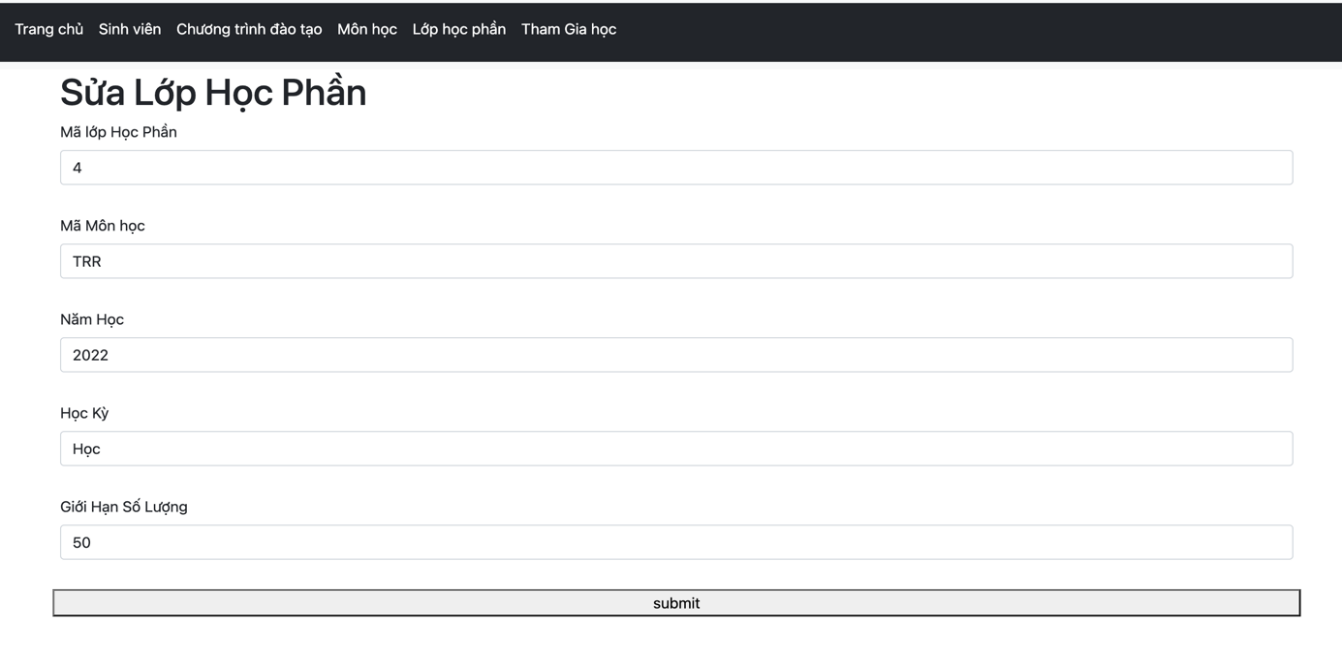
### **Lớp học phần**

Trang lớp học phần gồm có chức năng thêm, sửa và xoá, dùng để quản lý các hợp học phần trong một trường đại học. Bao gồm các các thông tin như Mã Lớp học phần, Mã môn học, Năm học, Học kỳ, Giới hạn về số lượng sinh viên

**Table

Description automatically generated**

Hình 11 - Giao diện trang lớp học phần (1)

****

Hình 12 - Giao diện trang lớp học phần (2)

**Graphical user interface, text, application, email

Description automatically generated**

Hình 13 - Giao diện trang lớp học phần (3)

### **Tham gia học**

Trang tham gia học gồm có chức năng thêm, sửa và xoá, dùng để quản lý các sinh viên trong các lớp học phần.

**Table

Description automatically generated**

Hình 14 - Giao diện trang tham gia học (1)

**Graphical user interface, application

Description automatically generated**

Hình 15 - Giao diện trang tham gia học (2)

**Graphical user interface, text, application

Description automatically generated**

Hình 16 - Giao diện trang tham giam học (3)

1. **Triển khai chương trình trên dịch vụ EC2 của AWS**

**Thực hiện các bước:**

Bước 1: Tạo EC2 Linux

1. Vào trang awsacamy và vào lớp Learner Lab sau đó chọn Start Lab



Hình 17 - Nút Start Lab

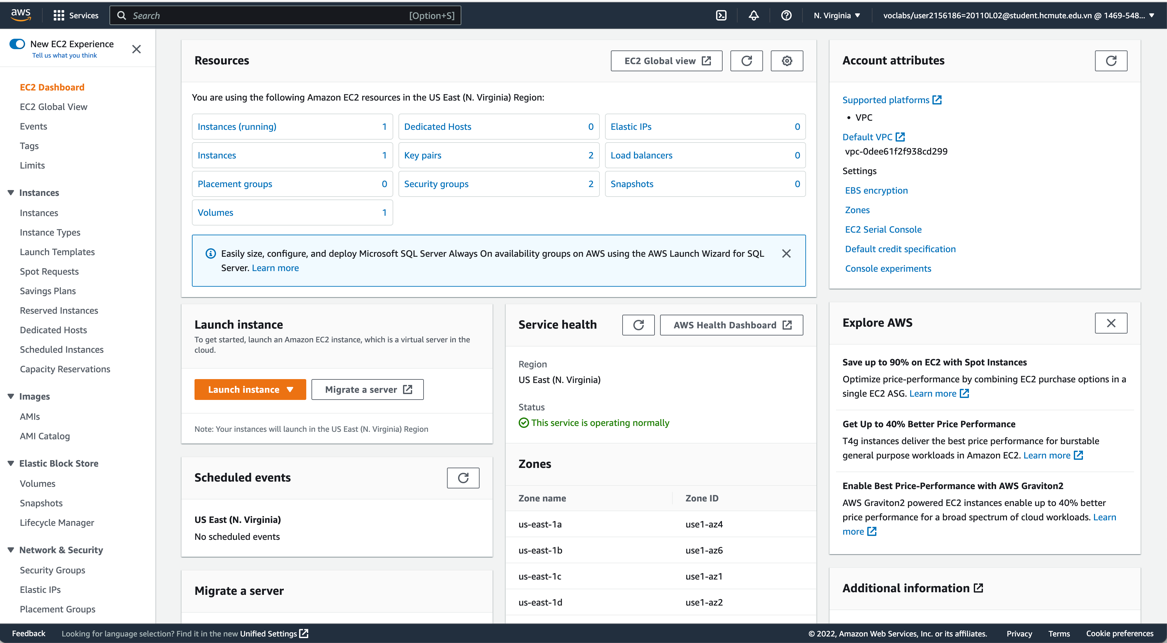
1. Sau khi bấm nút Start Lab nó sữ hiện lên như sau và chọn EC2

Graphical user interface, text, application, email

Description automatically generated

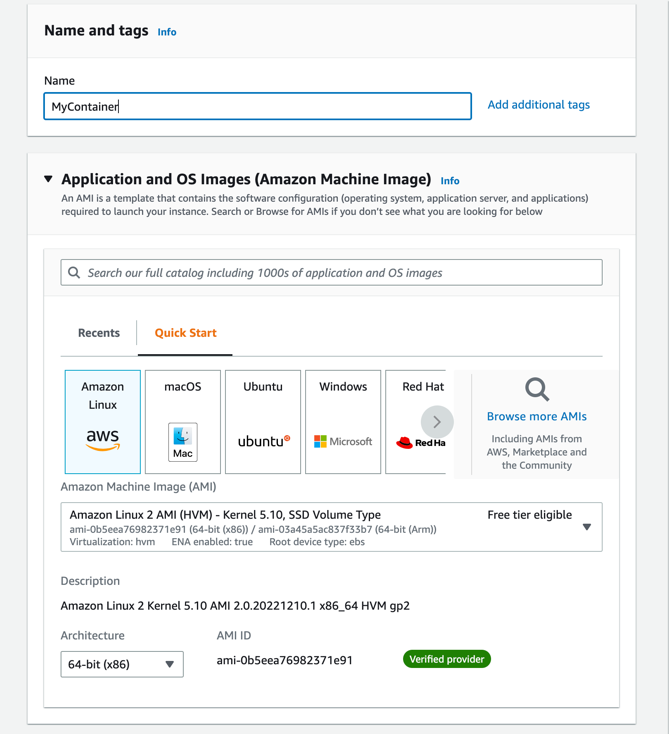
Hình 18 - Giao diện sau khi Start Lab

1. Nhấn vào nút Lauch instace



Hình 19 - Giao diện EC2

1. Sau khi Nhấn vào nút Lauch instace, trang cấu hình EC2 sẽ hiện ra. Ta đặt tên và sau đó chọn Amazon Linux



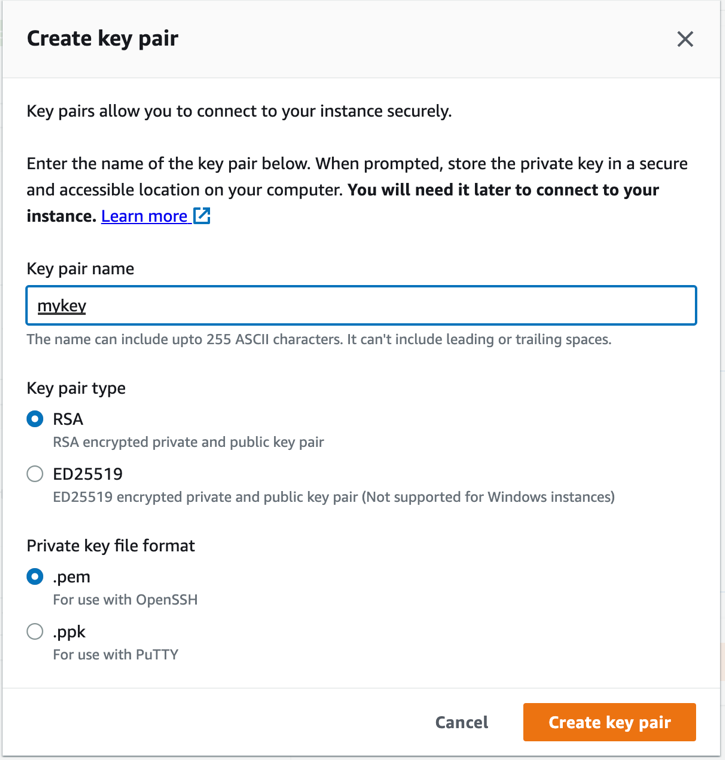
Hình 20 - Giao diện cấu hình EC2

1. Nhấn vào nút Create new key pair



Hình 21 - Tạo Keypair (1)

1. Đặt tên là mykey nhấn vào Create key pair



Hình 22 - Tạo Keypair (2)

1. Sau khi bấm nút Create key pair, Keypair sẽ được Download về máy
2. Chọn Subnet và đặt tên Security group name thành docker

Graphical user interface, application

Description automatically generated

Hình 23 - Chọn Subnet và đặt tên Security Group

1. Chọn Type thành All TCP và bấm nút Lauch instance

Graphical user interface, application

Description automatically generated

Hình 24 - Cấu hình các Rule cho Inbound Security Group

1. Thành công

Graphical user interface, text, application, email

Description automatically generated

Hình 25 - Tạo EC2 thành công

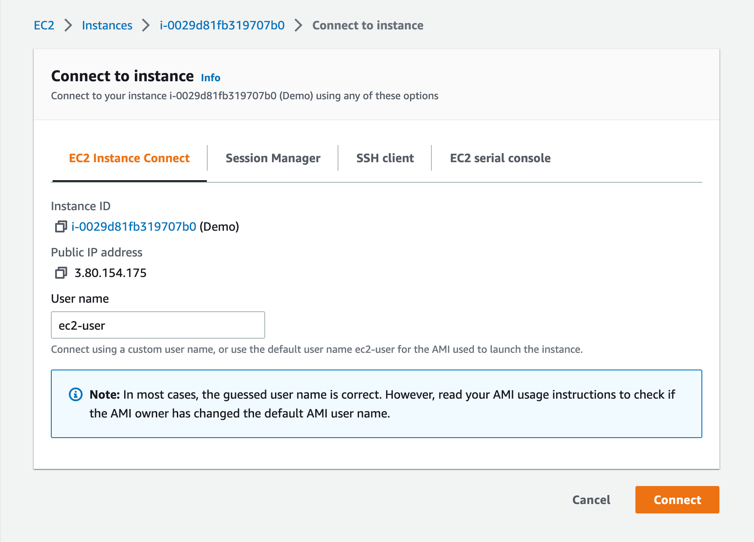
Bước 2: Kết nối đến EC2 vừa tạo

1. Graphical user interface, application

   Description automatically generated Chọn Connect

Hình - Connect đến EC2 (1)

1. Chọn Connect



Hình 27 - Connect đến EC2 (2)

Bước 3: Thực hiện cài đặt git

1. Mở terminal chạy sudo yum update -y
2. sudo yum install git -y
3. git clone <https://github.com/HT-Tuan/Project-Cloud-Computing.git>

Text

Description automatically generated

Hình 28 - Cài đặt git (1)

Text

Description automatically generated

Hình 29 - Cài đặt git (2)

Bước 4: Thực hiện cài đặt docker

1. sudo yum install docker -y
2. sudo systemctl enable docker
3. sudo service docker start
4. sudo usermod -a -G docker ec2-user
5. Docker info

Text

Description automatically generated

Hình 30 - Docker Info (1)

Shape

Description automatically generated with medium confidence

Hình 31 - Docker Info (2)

Bước 5: Thực hiện cài đặt docker compose

1. wget <https://github.com/docker/compose/releases/latest/download/docker-compose-linux-x86_64>
2. sudo mv docker-compose-linux-x86\_64 /usr/local/bin/docker-compose
3. cd /
4. cd ./usr/local/bin
5. sudo chmod +x /usr/local/bin/docker-compose
6. docker-compose version

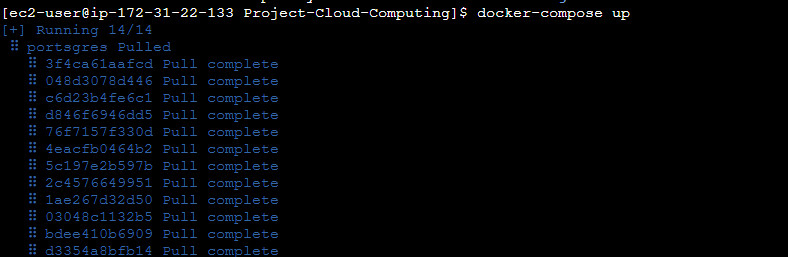
Text

Description automatically generated

Hình 32 - Cài đặt Docker Compose

Bước 6: Tạo các container – chạy chương trình.

Thực hiện lệnh docker-compose up

Hình 33 - Tạo các container - chạy chương trình

1. **Kết luận**
   1. **Ưu điểm của chương trình:**

Có một số lợi thế khi sử dụng chương trình quản lý sinh viên trong một cơ sở giáo dục. Một số trong những lợi thế này bao gồm:

Cải thiện hiệu quả: Một chương trình quản lý học sinh có thể giúp hợp lý hóa và tự động hóa nhiều tác vụ quản trị, chẳng hạn như chấm điểm, theo dõi điểm danh và lên lịch, giúp tiết kiệm thời gian cho giáo viên và nhân viên.

Giao tiếp nâng cao: Một chương trình quản lý học sinh có thể cung cấp một nền tảng tập trung để giao tiếp giữa giáo viên, học sinh và phụ huynh. Điều này có thể giúp cải thiện giao tiếp và phối hợp trong cộng đồng trường học.

Theo dõi dữ liệu tốt hơn: Một chương trình quản lý học sinh có thể giúp theo dõi và phân tích dữ liệu của học sinh, chẳng hạn như điểm số, chuyên cần và hành vi. Điều này có thể giúp giáo viên và quản trị viên xác định các xu hướng và lĩnh vực cần cải thiện.

* 1. **Hạn chế của chương trình:**

Có một số hạn chế tiềm năng của chương trình quản lý sinh viên có thể phụ thuộc vào các tính năng và khả năng cụ thể của chương trình. Một số hạn chế có thể bao gồm:

Tính linh hoạt hạn chế: Một số chương trình quản lý sinh viên có thể không linh hoạt và có thể không cho phép người dùng tùy chỉnh một số tính năng hoặc quy trình công việc để đáp ứng nhu cầu cụ thể của họ.

Phụ thuộc vào công nghệ: Các chương trình quản lý học sinh dựa vào công nghệ và nếu chương trình gặp sự cố kỹ thuật hoặc nếu người dùng không có quyền truy cập vào công nghệ cần thiết, thì việc quản lý dữ liệu học sinh có thể bị gián đoạn.

Bảo mật dữ liệu: Các chương trình quản lý sinh viên thường lưu trữ dữ liệu nhạy cảm về sinh viên, chẳng hạn như thông tin cá nhân và hồ sơ học tập. Nếu chương trình không có các biện pháp bảo mật mạnh mẽ, nó có khả năng dẫn đến vi phạm dữ liệu hoặc truy cập trái phép vào dữ liệu của học sinh.

Lỗi người dùng: Như với bất kỳ phần mềm nào, lỗi người dùng có thể dẫn đến lỗi nhập dữ liệu hoặc sử dụng chương trình không chính xác, điều này có thể dẫn đến việc ghi lại dữ liệu học sinh không chính xác hoặc không đầy đủ.

* 1. **Ý tưởng phát triển chương trình:**

Chương trình quản lý học sinh là một chương trình máy tính hoặc ứng dụng phần mềm được sử dụng để quản lý các khía cạnh khác nhau của trường học hoặc cơ sở giáo dục, chẳng hạn như hồ sơ học sinh, ghi danh, điểm số và điểm danh. Nó có thể được quản trị viên, giáo viên và các nhân viên khác sử dụng để theo dõi và sắp xếp dữ liệu học sinh, liên lạc với học sinh và phụ huynh, đồng thời hợp lý hóa các nhiệm vụ quản trị khác nhau.

Một số tính năng phổ biến của một chương trình quản lý sinh viên bao gồm:

* Nhân khẩu học và thông tin liên lạc của sinh viên
* Tuyển sinh và đăng ký
* Lập kế hoạch và quản lý lớp học
* Chấm điểm và bảng điểm
* Theo dõi chuyên cần
* Hồ sơ hành vi và kỷ luật của học sinh
* Công cụ giao tiếp (ví dụ: email, nhắn tin, cảnh báo)
* Báo cáo và phân tích

Khi phát triển chương trình quản lý sinh viên, bạn cần xem xét các nhu cầu và yêu cầu cụ thể của trường học hoặc tổ chức của mình, cũng như các nguồn lực và ngân sách hiện có. Bạn có thể muốn nghiên cứu phần mềm quản lý sinh viên hiện có và xem xét cách tùy chỉnh phần mềm này để đáp ứng nhu cầu của bạn hoặc bạn có thể chọn xây dựng một chương trình từ đầu. Điều quan trọng nữa là phải xem xét trải nghiệm người dùng và khả năng sử dụng, vì chương trình sẽ cần phải dễ sử dụng và điều hướng cho nhân viên, học sinh và phụ huynh.

**TÀI LIỆU THAM KHẢO**

1. [*Multi container apps*](https://docs.docker.com/get-started/07_multi_container/)
2. [*Installing docker-compose-plugin on amazon linux 2*](https://stackoverflow.com/questions/72187612/installing-docker-compose-plugin-on-amazon-linux-2)
3. [*Build REST API with Spring Boot and JPA*](https://www.youtube.com/watch?v=BCx8e9RDjB0)
4. [*Using Docker with Postgres*](https://earthly.dev/blog/postgres-docker/)