

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP HCM
KHOA ĐÀO TẠO CHẤT LƯỢNG CAO



ĐỒ ÁN ĐIỆN TOÁN ĐÁM MÂY
XÂY DỰNG TRANG VIẾT CODE
ONLINE CHO JAVA, C#, PYTHON, C++

Giáo viên hướng dẫn: Huỳnh Xuân Phụng

Lớp: Dien toan dam may_ Nhom 04CLC

Nhóm sinh viên thực hiện: Nhóm 9

MSSV	Họ và tên	Mức độ đóng góp
20142498	Nguyễn Đình Hiếu	100%
20110555	Vũ Thanh Sang	100%
20100591	Nguyễn Văn Tú	100%

TPHCM, tháng 12 năm 2022

LỜI CẢM ƠN

Để hoàn thành tốt đề tài và bài báo cáo này, chúng em xin gửi lời cảm ơn chân thành đến giảng viên, thầy Huỳnh Xuân Phụng, người đã trực tiếp hỗ trợ chúng em trong suốt quá trình làm đề tài. Chúng em cảm thầy đã đưa ra những lời khuyên từ kinh nghiệm thực tiễn của mình để định hướng cho chúng em đi đúng với yêu cầu của đề tài đã chọn, luôn giải đáp thắc mắc và đưa ra những góp ý, chỉnh sửa kịp thời giúp chúng em khắc phục nhược điểm và hoàn thành tốt cũng như đúng thời hạn đã đề ra.

Chúng em cũng xin gửi lời cảm ơn chân thành các quý thầy cô trong khoa Đào tạo Chất Lượng Cao nói chung và ngành Công Nghệ Thông Tin nói riêng đã tận tình truyền đạt những kiến thức cần thiết giúp chúng em có nền tảng để làm nên đề tài này, đã tạo điều kiện để chúng em có thể tìm hiểu và thực hiện tốt đề tài. Cùng với đó, chúng em xin được gửi cảm ơn đến các bạn cùng khóa đã cung cấp nhiều thông tin và kiến thức hữu ích giúp chúng em có thể hoàn thiện hơn đề tài của mình.

Đề tài và bài báo cáo được chúng em thực hiện trong khoảng thời gian ngắn, với những kiến thức còn hạn chế cùng nhiều hạn chế khác về mặt kỹ thuật và kinh nghiệm trong việc thực hiện một dự án web. Do đó, trong quá trình làm nên đề tài có những thiếu sót là điều không thể tránh khỏi nên chúng em rất mong nhận được những ý kiến đóng góp quý báu của các quý thầy cô để kiến thức của chúng em được hoàn thiện hơn và chúng em có thể làm tốt hơn nữa trong những lần sau. Chúng em xin chân thành cảm ơn.

Cuối lời, chúng em kính chúc quý thầy, quý cô luôn dồi dào sức khỏe và thành công hơn nữa trong sự nghiệp trồng người. Một lần nữa chúng em xin chân thành cảm ơn.

MỤC LỤC

CHƯƠNG 1. MỞ ĐẦU	4
1.1. Phát biểu về bài toán:	4
1.2. Mục đích, yêu cầu thực hiện:	4
1.3. Đối tượng và phạm vi nghiên cứu:	5
1.3.1. Đối tượng nghiên cứu:	5
1.3.2. Phạm vi nghiên cứu:	5
1.4. Giao diện dự kiến:	5
1.5. Công cụ và môi trường:	5
CHƯƠNG 2. PHÂN CÔNG CÔNG VIỆC	6
Bảng 1: Bảng phân công nhiệm vụ	6
CHƯƠNG 3. PHÂN TÍCH, THIẾT KẾ GIẢI PHÁP	7
3.1. Khái quát về điện toán đám mây:	7
3.1.1. Điện toán đám mây là gì?	7
3.1.2. Ai sử dụng điện toán đám mây?	7
3.1.3. Lợi ích của điện toán đám mây	7
3.2. ReactJs:	8
3.3. NodeJs:	9
3.4. EC2:	10
3.5. Docker:	11
3.6. DynamoDB:	12
CHƯƠNG 4. PHÂN TÍCH TRANG WEB VÀ CÁC DỊCH VỤ ĐIỆN TOÁN Đám Mây	13
4.1. Phân tích trang web:	13
4.1.1. Trang home:	13
4.1.2. Trang đăng nhập, đăng ký:	13
4.1.3. Trang code:	15

4.1.4. Trang luyện tập:	16
4.2. Xây dựng ứng dụng:	18
4.2.1. Tạo VPC:	18
4.2.2. Tạo Security groups:	20
4.2.3. Tạo EC2:	22
4.2.4. Tạo ra DynamoDB:	28
CHƯƠNG 5. KẾT LUẬN	31
5.1. Kết quả đạt được:	31
5.2. Định hướng phát triển:	31
5.3. Ưu điểm, hạn chế:	31
5.3.1. Ưu điểm:	31
5.3.2. Hạn chế:	32
TÀI LIỆU THAM KHẢO	33

CHƯƠNG 1. MỞ ĐẦU

1.1. Phát biểu về bài toán:

Trong thời đại ngày nay, công nghệ thông tin ngày càng phát triển, số lượng sinh viên theo ngành này ngày càng nhiều. Hiện nay có nhiều trình biên dịch hỗ trợ cho việc code của sinh viên, tuy nhiên muốn code được nhiều ngôn ngữ phải có nhiều trình biên dịch khác nhau. Hiểu được sự khó khăn đó nhóm đã xây dựng nên một trang web hỗ trợ cho việc code online cho các ngôn ngữ như java, c#, python, c++. Điều này mang lại sự thuận tiện cho người sử dụng, chỉ cần truy cập vào trang web lựa chọn ngôn ngữ và code. Kết hợp cùng với các kiến thức và các dịch vụ mà môn điện toán đám mây đã trang bị. Nhóm đã quyết định chọn chủ đề **“Xây dựng trang web viết code online cho java, c#, c++, python”** có sử dụng dịch vụ của aws.

1.2. Mục đích, yêu cầu thực hiện:

Đề tài **Xây dựng trang web viết code online cho java, c#, c++, python** đặt ra các vấn đề trọng tâm về mục tiêu mà sinh viên nghiên cứu cần thực hiện được bao gồm:

Vấn đề đầu tiên được đưa ra về yếu tố lý thuyết, sinh viên áp dụng những kiến thức, nghiên cứu hiểu được tổng quan và khả năng ứng dụng vào trong bài làm. Song song với việc áp dụng công nghệ được lựa chọn để sử dụng, nhóm sinh viên cần tìm hiểu thêm về cách xây dựng giao diện đẹp, thân thiện với người dùng. Xây dựng trang web viết code online cho java, c#, python, c++, xây dựng trang luyện tập và sử dụng docker để compiler code từ trình duyệt

Vấn đề thứ hai là nâng cao khả năng làm việc nhóm, tìm hiểu và thảo luận về đề tài, các vấn đề đưa ra từ đó tìm ra câu trả lời. Xây dựng thái tích cực trong việc học, làm đồ án.

1.3. Đối tượng và phạm vi nghiên cứu:

1.3.1. Đối tượng nghiên cứu:

Xây dựng trang web bằng ReacJS và redux cho fontend và sử dụng nodejs kết hợp dynamoDB cho phần backend.

1.3.2. Phạm vi nghiên cứu:

Xây dựng trang web viết code online cho java, c#, c++, python

1.4. Giao diện dự kiến:

Để thích hợp với một trang web như vậy thì nhóm dự kiến sẽ xây dựng các trang bao gồm trang chủ là nơi giới thiệu thông tin website, trang login để đăng nhập, trang đăng ký để tạo tài khoản, trang compile để thực hiện compile code theo ngôn ngữ được chọn và trang để luyện tập code.

1.5. Công cụ và môi trường:

- IDE: Visual Studio Code.
- Ngôn ngữ: ReactJs, Redux cho font-end và NodeJs, DynamoDB cho backend.

CHƯƠNG 2. PHÂN CÔNG CÔNG VIỆC

STT	Danh sách thành viên	Công việc thực hiện được	Tỉ lệ hoàn thành
1	Nguyễn Đình Hiếu	Xây dựng giao diện, viết báo cáo, xây dựng các chức năng trang home, đăng ký, trang compile code, trang luyện tập.	100%
2	Vũ Thanh Sang	Làm việc với backend, kết nối dyanmoDB, viết báo cáo.	100%
3	Nguyễn Văn Tú	Xây dựng giao diện, xây dựng các chức năng trang đăng nhập, đăng ký, xây dựng file docker, viết báo cáo.	100%

Bảng 1: Bảng phân công nhiệm vụ

CHƯƠNG 3. PHÂN TÍCH, THIẾT KẾ GIẢI PHÁP

3.1. Khái quát về điện toán đám mây:

3.1.1. Điện toán đám mây là gì?

Điện toán đám mây là việc phân phối các tài nguyên công nghệ thông tin (CNTT) theo nhu cầu qua Internet với chính sách thanh toán theo mức sử dụng. Thay vì mua, sở hữu và bảo trì các trung tâm dữ liệu và máy chủ vật lý, bạn có thể tiếp cận các dịch vụ công nghệ, như năng lượng điện toán, lưu trữ và cơ sở dữ liệu, khi cần thiết, từ nhà cung cấp dịch vụ đám mây như Amazon Web Services (AWS).

3.1.2. Ai sử dụng điện toán đám mây?

Các tổ chức thuộc mọi loại hình, quy mô và ngành hoạt động đang dùng dịch vụ đám mây cho nhiều trường hợp sử dụng, như sao lưu dữ liệu, khôi phục sau thảm họa, email, máy tính để bàn ảo, phát triển và kiểm thử phần mềm, phân tích dữ liệu lớn và ứng dụng web tương tác với khách hàng. Ví dụ: các công ty chăm sóc sức khỏe đang sử dụng dịch vụ đám mây để phát triển các phương pháp điều trị phù hợp hơn cho bệnh nhân. Các công ty dịch vụ tài chính đang sử dụng dịch vụ đám mây để tăng cường phát hiện và ngăn chặn gian lận theo thời gian thực. Và các nhà sản xuất trò chơi điện tử đang sử dụng dịch vụ đám mây để cung cấp các trò chơi trực tuyến cho hàng triệu người chơi trên toàn thế giới.

3.1.3. Lợi ích của điện toán đám mây

Nhanh chóng: Đám mây cho phép bạn dễ dàng tiếp cận nhiều công nghệ để bạn có thể đổi mới nhanh hơn và phát triển gần như mọi thứ mà bạn có thể tưởng tượng. Bạn có thể nhanh chóng thu thập tài nguyên khi cần từ các dịch vụ cơ sở hạ tầng, như điện toán, lưu trữ, và cơ sở dữ liệu, đến Internet of Things, machine learning, kho dữ liệu và phân tích, v.v.

Bạn có thể triển khai các dịch vụ công nghệ một cách nhanh chóng và tiến hành từ khâu ý tưởng đến khâu hoàn thiện nhanh hơn một vài cấp bậc cường độ so với trước đây. Điều này cho phép bạn tự do thử nghiệm, kiểm thử những ý tưởng mới để phân biệt trải nghiệm của khách hàng và chuyển đổi doanh nghiệp của bạn.

Quy mô linh hoạt: Với điện toán đám mây, bạn không phải cung cấp tài nguyên quá mức để xử lý các hoạt động kinh doanh ở mức cao nhất trong tương lai. Thay vào đó, bạn cung cấp lượng tài nguyên mà bạn thực sự cần. Bạn có thể tăng hoặc giảm quy mô của các tài nguyên này ngay lập tức để tăng và giảm dung lượng khi nhu cầu kinh doanh của bạn thay đổi.

Tiết kiệm chi phí: Nền tảng đám mây cho phép bạn thay thế các khoản chi phí cố định (như trung tâm dữ liệu và máy chủ vật lý) bằng các khoản chi phí biến đổi, đồng thời chỉ phải trả tiền cho tài nguyên CNTT mà bạn sử dụng. Bên cạnh đó, chi phí biến đổi cũng sẽ thấp hơn nhiều so với chi phí bạn tự trang trải do tính kinh tế theo quy mô.

Triển khai trên toàn cầu chỉ trong vài phút: Với đám mây, bạn có thể mở rộng sang các khu vực địa lý mới và triển khai trên toàn cầu trong vài phút. Ví dụ: AWS có cơ sở hạ tầng trên toàn thế giới, vì vậy, bạn có thể triển khai ứng dụng của mình ở nhiều địa điểm thực tế chỉ bằng vài cú nhấp chuột. Đặt các ứng dụng gần hơn với người dùng cuối giúp giảm độ trễ và cải thiện trải nghiệm của họ.

3.2. ReactJs:

ReactJs là một opensource được phát triển bởi Facebook, ra mắt vào năm 2013, bản thân nó là một thư viện Javascript được dùng để để xây dựng các tương tác với các thành phần trên website. Một trong những điểm nổi bật nhất của ReactJS đó là việc render dữ liệu không chỉ thực hiện được trên tầng Server mà còn ở dưới Client nữa.

Ngoài việc hỗ trợ xây dựng giao diện nhanh, hạn chế lỗi trong quá trình code, cải thiện hiệu website thì chúng cũng đem lại những ưu, nhược điểm khác nhau cho người sử dụng. Sau đây ta tìm hiểu về một vài ưu nhược của chúng.

✓ Ưu điểm:

+ Hỗ trợ chúng ta rất nhiều trong việc chia cấu trúc file để có thể code dễ dàng hơn. Cung cấp rất nhiều loại hook thay thế cho JavaScript một cách tiện lợi nhanh chóng.

+ Phù hợp với đa dạng thể loại website: ReactJS khiến cho việc khởi tạo website dễ dàng hơn bởi vì ta không cần phải code nhiều như khi tạo trang web thuần chỉ dùng JavaScript, HTML.

+Tái sử dụng các Component: nếu xây dựng một component tốt, ta có thể sử dụng lại chúng vào nhiều trang web hoặc layout khác nhau thay cho việc làm lại từ đầu.

+ Ngoài ra còn được sử dụng cho cả Mobile application.

+ Debug nhanh chóng và dễ dàng: Facebook đã phát hành 1 Chrome extension hỗ trợ việc debug khi gặp lỗi. Phần nào giúp ta tăng hiệu suất khi xây dựng trang web.

✓ Nhược điểm:

+ Tích hợp Reactjs vào các framework MVC truyền thống nên yêu cầu phải cấu hình lại toàn bộ file.

+ React khá nặng so với một vài framework khác (Khoảng 35kb so với 39kb của Angular).

+ Khó tiếp cận cho người mới học Web.

3.3. NodeJs:

NodeJS là một môi trường runtime chạy JavaScript đa nền tảng và có mã nguồn mở, được sử dụng để chạy các ứng dụng web bên ngoài trình duyệt của client. Nền

tầng này được phát triển bởi Ryan Dahl vào năm 2009, được xem là một giải pháp hoàn hảo cho các ứng dụng sử dụng nhiều dữ liệu nhờ vào mô hình hướng sự kiện (event-driven) không đồng bộ.

✓ Ưu điểm:

- + IO hướng sự kiện không đồng bộ, cho phép xử lý nhiều yêu cầu đồng thời.
- + Sử dụng JavaScript – một ngôn ngữ lập trình dễ học.
- + Chia sẻ cùng code ở cả phía client và server.
- + NPM(Node Package Manager) và module Node đang ngày càng phát triển mạnh mẽ.
- + Cộng đồng hỗ trợ tích cực.
- + Cho phép stream các file có kích thước lớn.

✓ Nhược điểm:

- + Không có khả năng mở rộng, vì vậy không thể tận dụng lợi thế mô hình đa lõi trong các phần cứng cấp server hiện nay.
- + Khó thao tác với cơ sở dữ liệu quan hệ.
- + Mỗi callback sẽ đi kèm với rất nhiều callback lồng nhau khác.
- + Cần có kiến thức tốt về JavaScript.
- + Không phù hợp với các tác vụ đòi hỏi nhiều CPU.

3.4. EC2:

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.

Những trường hợp sử dụng chẳng hạn như chạy các ứng dụng hoạt động trên đám mây và tại doanh nghiệp, thay đổi quy mô cho các ứng dụng HPC, phát triển các nền tảng Apple, đào tạo và triển khai các ứng dụng ML.

3.5. Docker:

Docker là nền tảng phần mềm cho phép bạn 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, bạn có thể nhanh chóng triển khai và thay đổi quy mô ứng dụng vào bất kỳ môi trường nào và biết chắc rằng mã của bạn 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.

Docker hoạt động bằng cách cung cấp phương thức tiêu chuẩn để chạy mã của bạn. 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.

Việc sử dụng Docker cho phép bạn vận chuyển mã nhanh hơn, tiêu chuẩn hóa hoạt động của ứng dụng, di chuyển mã một cách trơn tru và tiết kiệm tiền bằng cách cải thiện khả năng tận dụng tài nguyên. Với Docker, bạn sẽ được nhận một đối tượng duy nhất có khả năng chạy ổn định ở bất kỳ đâu. Cú pháp đơn giản và không phức tạp của Docker sẽ cho bạn quyền kiểm soát hoàn toàn. Việc đưa vào áp dụng rộng rãi nền

tầng này đã tạo ra một hệ sinh thái bền vững các công cụ và ứng dụng có thể sử dụng ngay đã sẵn sàng sử dụng với Docker.

3.6. DynamoDB:

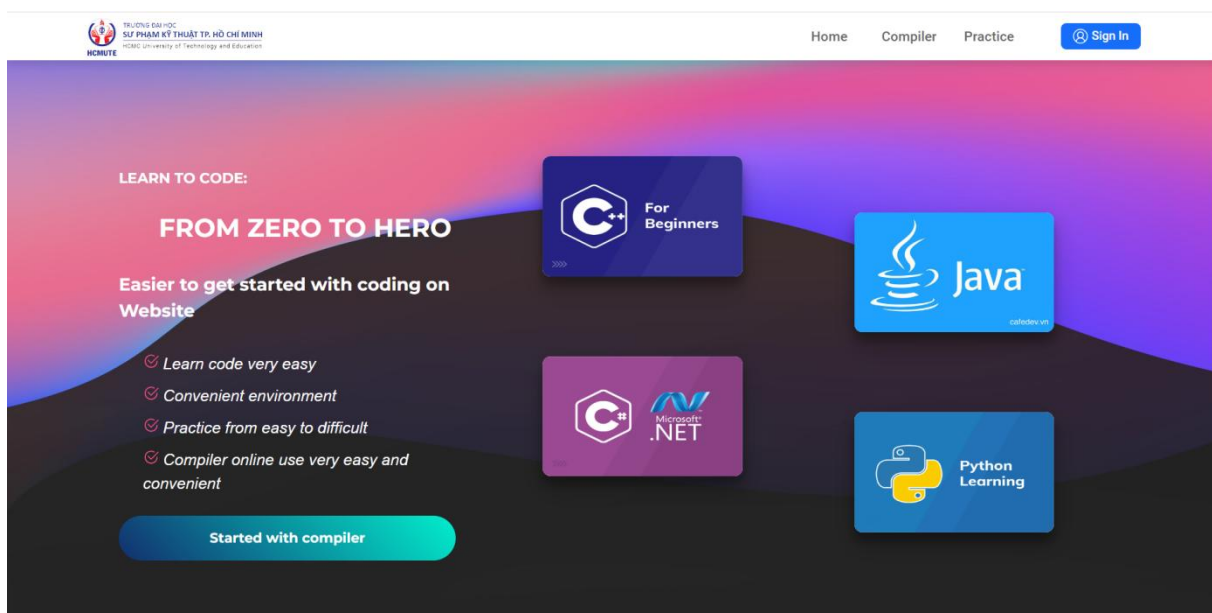
DynamoDB là một dịch vụ quản lý NoSQL có khả năng đáp ứng hiệu suất cao và nhanh kèm theo khả năng mở rộng. Nếu bạn là một nhà phát triển, bạn có thể sử dụng DynamoDB để tạo ra một bảng có khả năng lưu trữ và truy xuất bất kỳ số lượng dữ liệu, mà vẫn có thể phục vụ cho bất kỳ mức độ request traffic. DynamoDB tự động phân tán dữ liệu và traffic của một bảng ra một số lượng server vừa đủ để có thể xử lý request capacity đặt ra bởi khách hàng và lượng dữ liệu lưu trữ, và đồng thời đảm bảo hiệu suất nhanh và đồng nhất. Tất cả dữ liệu được lưu trữ trên SSD và tự động được sao chép ra các vùng sẵn sàng (Availability Zones) trong một khu vực (Region) để cung cấp độ sẵn sàng cao và độ bền của dữ liệu (high availability and data durability).

CHƯƠNG 4. PHÂN TÍCH TRANG WEB VÀ CÁC DỊCH VỤ ĐIỆN TOÁN Đám Mây

4.1. Phân tích trang web:

4.1.1. Trang home:

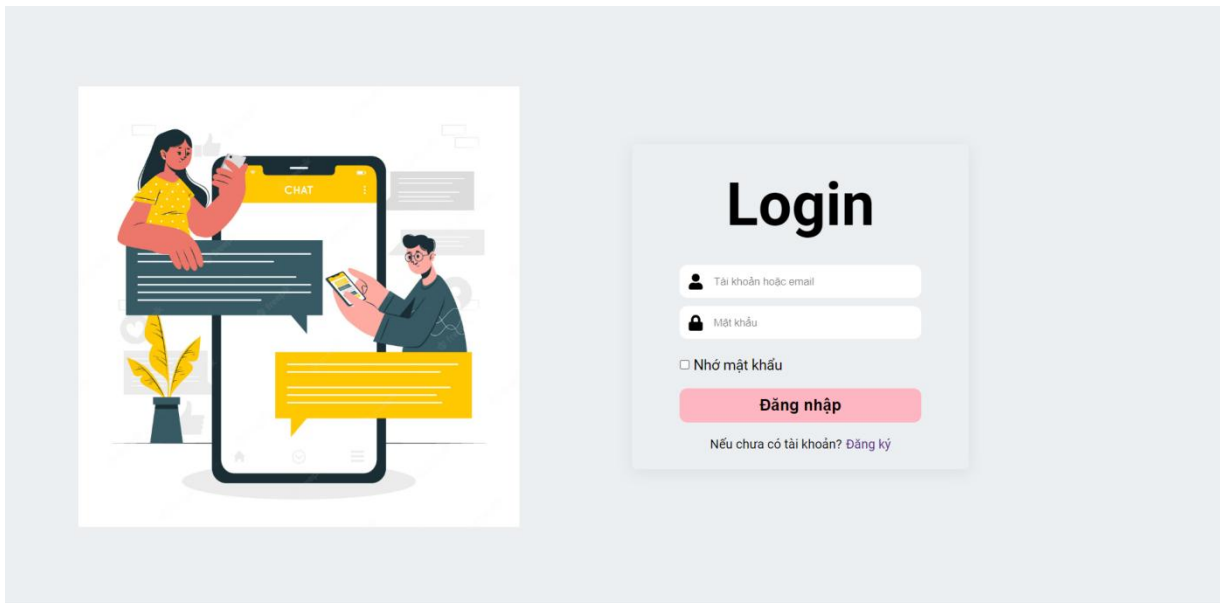
Mở đầu khi bắt đầu vào trang web sẽ là trang home, tại trang này sẽ hiển thị một vài thông tin giới thiệu sơ bộ về trang web. Phần header bao gồm menu dẫn người dùng đến trang home, trang compiler, trang practice và login. Phần content giới thiệu sơ về ngôn ngữ và một button dẫn người dùng đến trang compiler.



Hình 1: Trang home

4.1.2. Trang đăng nhập, đăng ký:

Khi nhấn vào nút sign in ở phần header sẽ dẫn người dùng đến trang đăng nhập. Tại đây nhập email và password để login vào trong trang web. Nếu chưa có tài khoản thì người dùng có thể đăng ký tài khoản.



Hình 2: Trang login

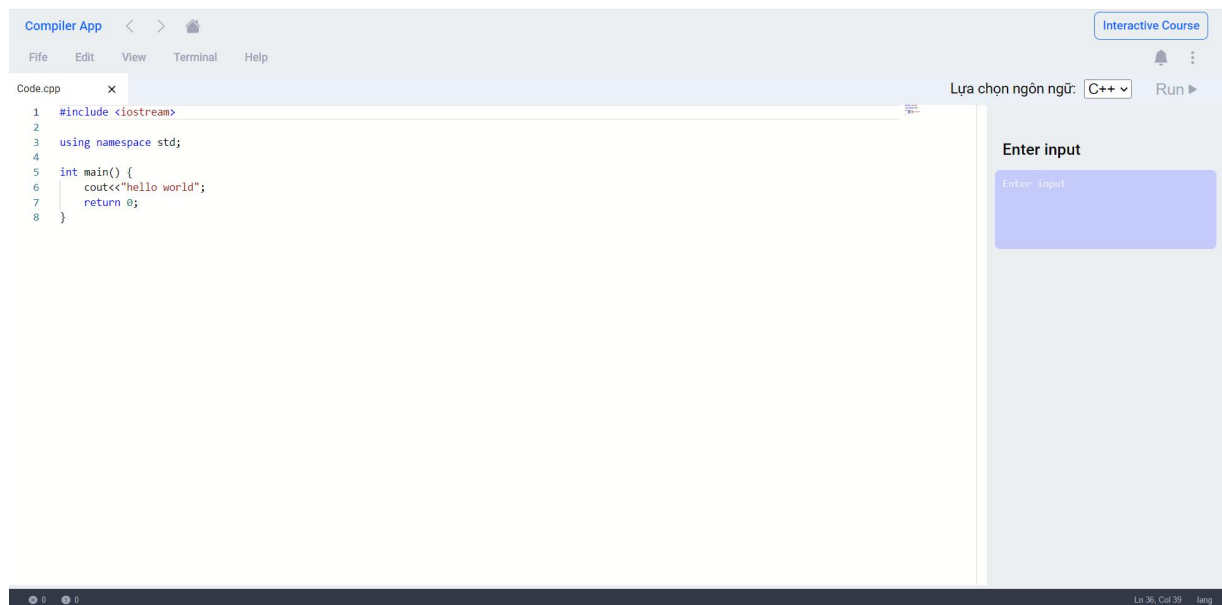
Khi nhấn vào đăng ký sẽ đưa người dùng đến trang đăng ký, tại đây người dùng phải nhập firstname, lastname, email, password và confirm password. Sau khi nhấn nút đăng ký, nếu đăng ký thành công lập tức đưa người dùng tới trang đăng nhập.

Hình 3: Trang register

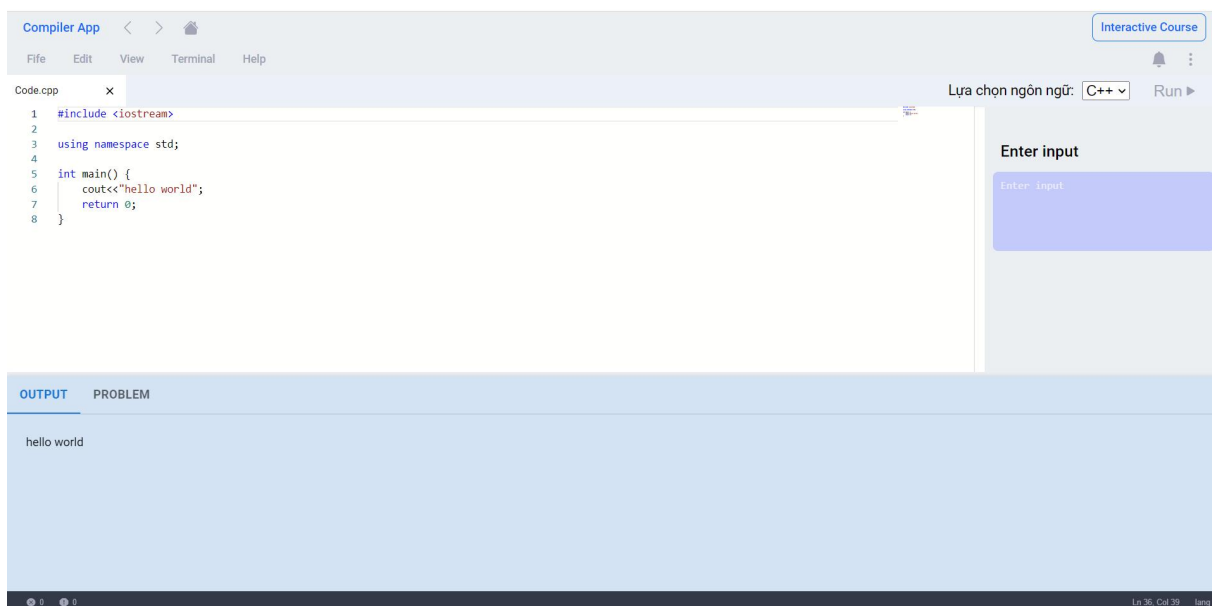
4.1.3. Trang code:

Khi bắt đầu vào trang code sẽ hiển thị lên giao diện như hình bên dưới. Phía bên góc trên bên phải màn hình là một combobox cho phép ta lựa chọn ngôn ngữ java, c++, c#, python. Giao diện chia làm ba phần chính. Phần đầu tiên là phần dành cho việc code ngôn ngữ, tại đây đối với mỗi ngôn ngữ nhóm đã format sẵn cấu trúc code chẳng hạn như đối với ngôn ngữ c++ thì format code như hình và tương ứng với từng ngôn ngữ đều có. Phần thứ hai là dữ liệu đầu vào, đối với những đoạn code không cần nhập đầu vào thì không cần nhập vào ô. Phần thứ ba, sau khi code xong chương trình và nhập dữ liệu đầu vào (nếu có) nhấn vào nút run một bên lựa chọn ngôn ngữ thì sẽ thực hiện compile code, lúc này sẽ hiển thị lên kết quả compile được ở phần output và lỗi ở phần problem.

Ví nhập lựa chọn ngôn ngữ c++ như hình và nhấn thực hiện đoạn code in ra màn hình dòng chữ “hello world” nhấn vào run thì lập tức dòng chữ “hello world” hiển thị ra ở phần output.



Hình 4: Trang compile code

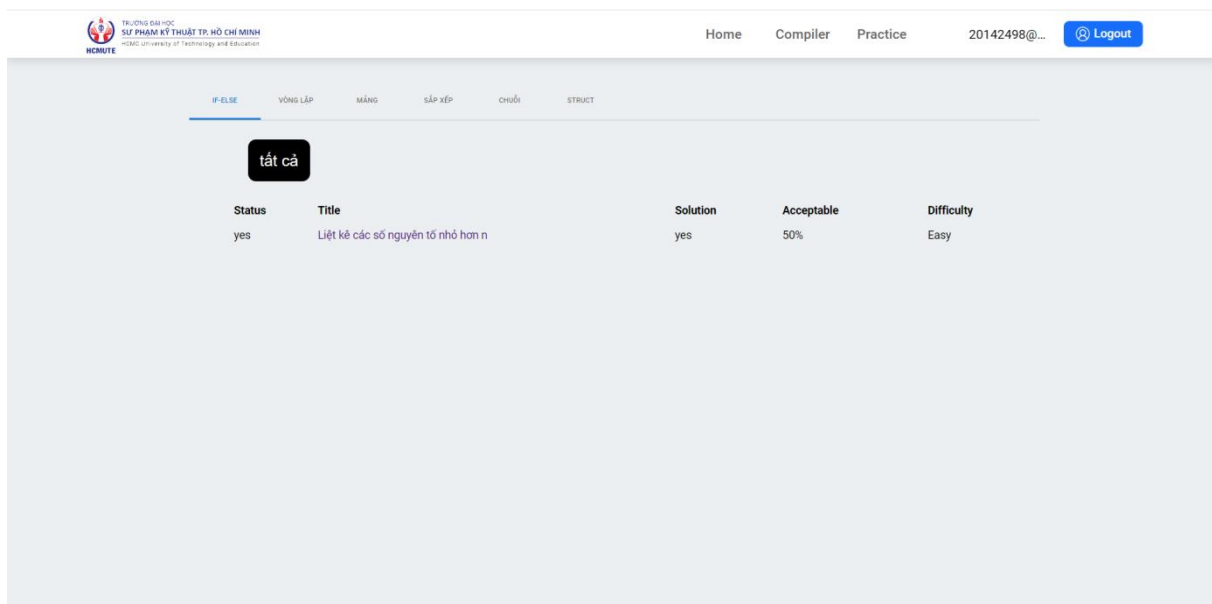


Hình 5: Trang show kết quả compile

4.1.4. Trang luyện tập:

Sau khi đã đăng nhập vào trang web thì sẽ hiển thị lên danh sách bài tập mà người dùng có thể luyện tập. Nếu chưa đăng nhập thì sẽ hiển thị lên thông báo yêu cầu người dùng đăng nhập. Trang này hiển thị lên tiêu đề của bài, độ khó của bài và trạng thái làm bài, nếu người dùng đã giải thành công thì trạng thái là “yes” ngược lại là “no”.

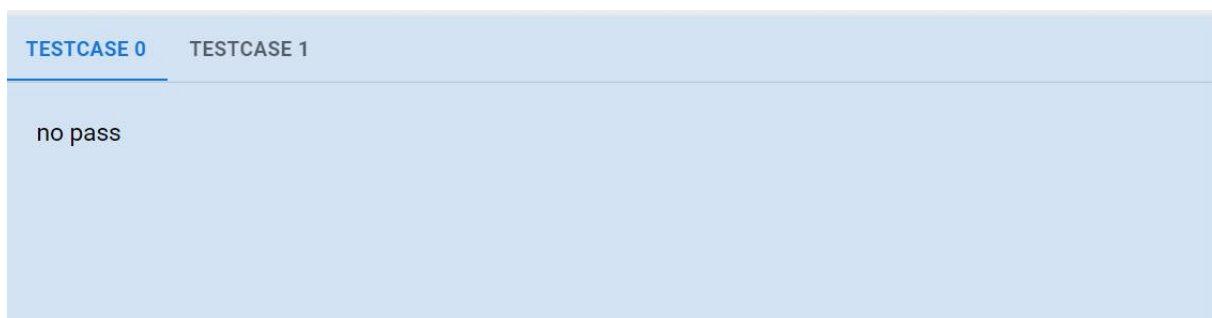
Khi nhấn vào title bài muốn làm thì sẽ dẫn người dùng đến trang solution là trang cho phép người dùng giải bài tập lựa chọn.



Hình 6: Trang luyện tập

Trong trang này sẽ có hai phần, phần bên trái sẽ là đề bài. Đề sẽ hiển thị lên đề bài các yêu cầu, ràng buộc về dữ liệu đầu vào, các ví dụ về input output để người dùng dễ hiểu và thực hiện đúng yêu cầu. Phần bên phải tương tự như trang compile cũng cho phép người dùng lựa chọn ngôn ngữ, có format code sẵn, nút run khi run thì từng testcase sẽ hiển thị lên là hoàn thành hay không.

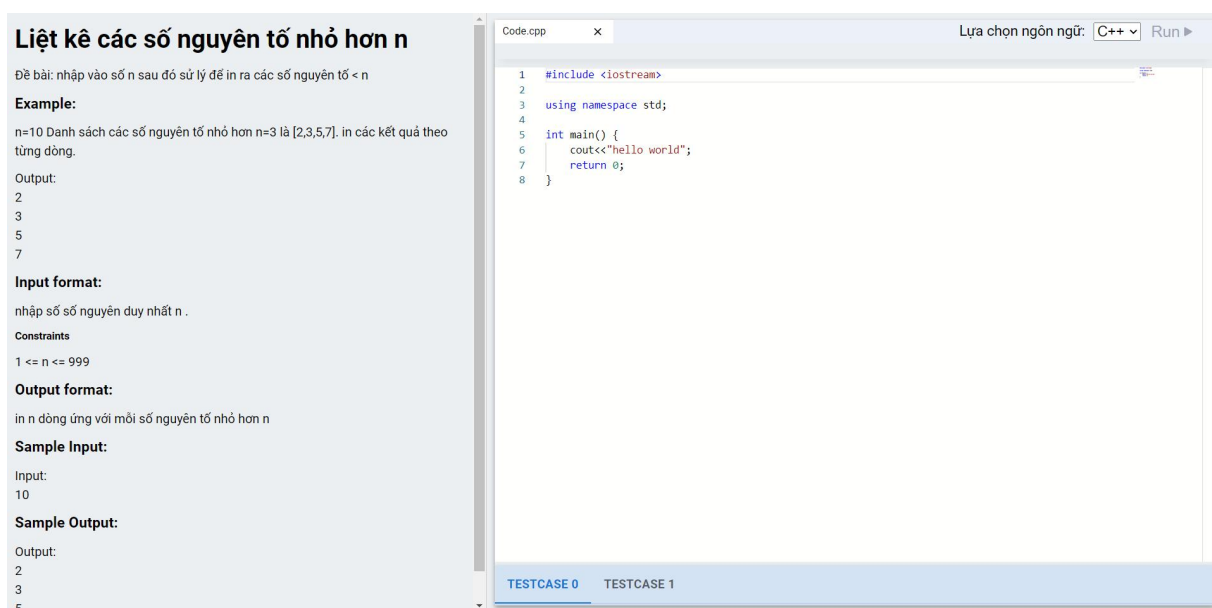
Ví dụ với đề bài như hình bên dưới, khi nhập code và run thì trả lại kết quả của từng testcase, đối với testcase đầu tiên thì báo “no pass” và test case thứ 2 cũng báo “no pass”.



Hình 7: Trang testcase 0



Hình 8: Trang testcase 1



Hình 9: Trang solution

4.2. Xây dựng ứng dụng:

Có 2 cách để deploy trang web sử dụng aws: cách đầu tiên tạo ra VPC, tạo ra security groups và tạo ra ec2 sử dụng VPC và security groups tạo ra. Còn nếu sử dụng tài khoản được cấp có thể sử dụng VPC có sẵn.

4.2.1. Tạo VPC:

Lựa chọn VPC của aws, sau đó nhấn vào your VPCs => create VPC sẽ hiển thị ra giao diện bên dưới.

VPC Management Console x IAM Management Console x IAM Management Console x +

https://us-east-1.console.aws.amazon.com/vpc/home?region=us-east-1#CreateVpccreateMode=vpcWithResources

Facebook Gmail YouTube My Drive - Google... Chi tiết khóa học... Như Như AUDIO - TOEIC FIRS...

aws Services Search [Alt+S] N. Virginia baotoan2016

Create VPC [Info](#)

A VPC is an isolated portion of the AWS Cloud populated by AWS objects, such as Amazon EC2 instances. Mouse over a resource to highlight the related resources.

VPC settings

Resources to create [Info](#)
Create only the VPC resource or the VPC and other networking resources.

☐ VPC only ☒ VPC and more

Name tag auto-generation [Info](#)
Enter a value for the Name tag. This value will be used to auto-generate Name tags for all resources in the VPC.

☒ Auto-generate
labVPC

IPv4 CIDR block [Info](#)
Determine the starting IP and the size of your VPC using CIDR notation.

10.0.0.0/16 65,536 IPs

IPv6 CIDR block [Info](#)
☒ No IPv6 CIDR block
☐ Amazon-provided IPv6 CIDR block

Tenancy [Info](#)
Default

Preview

Introducing the new create VPC experience
We've designed the new create VPC experience to make it easier to use. Now you can visualize the resources that will be created.

- New: Edit the name tag of individual resources. Uncheck "Auto-generate" and set each name tag in the visualizer directly.

Let us know what you think.

VPC [Show details](#)
Your AWS virtual network
labVPC-vpc

Subnets (4)
Subnets within this VPC
us-east-1a
labVPC-subnet-public1-us-east-1a
labVPC-subnet-private1-us-east-1a
us-east-1b
labVPC-subnet-public2-us-east-1b
labVPC-subnet-private2-us-east-1b

Route table
Route network tra
labVPC-rtb-pub
labVPC-rtb-priv
labVPC-rtb-priv

Feedback Looking for language selection? Find it in the new Unified Settings

© 2022, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

Number of Availability Zones (AZs)

[Info](#)
Choose the number of AZs in which to provision subnets. We recommend at least two AZs for high availability.

1 2 3

► **Customize AZs**

Number of public subnets [Info](#)
The number of public subnets to add to your VPC. Use public subnets for web applications that need to be publicly accessible over the internet.

0 2

Number of private subnets [Info](#)
The number of private subnets to add to your VPC. Use private subnets to secure backend resources that don't need public access.

0 2 4

▼ **Customize subnets CIDR blocks**

Public subnet CIDR block in us-east-1a
10.0.0.0/24 256 IPs

Public subnet CIDR block in us-east-1b
10.0.2.0/24 256 IPs

Private subnet CIDR block in us-east-1a
10.0.1.0/24 256 IPs

Private subnet CIDR block in us-east-1b
10.0.3.0/24 256 IPs

Preview

Introducing the new create VPC experience
We've designed the new create VPC experience to make it easier to use. Now you can visualize the resources that will be created.

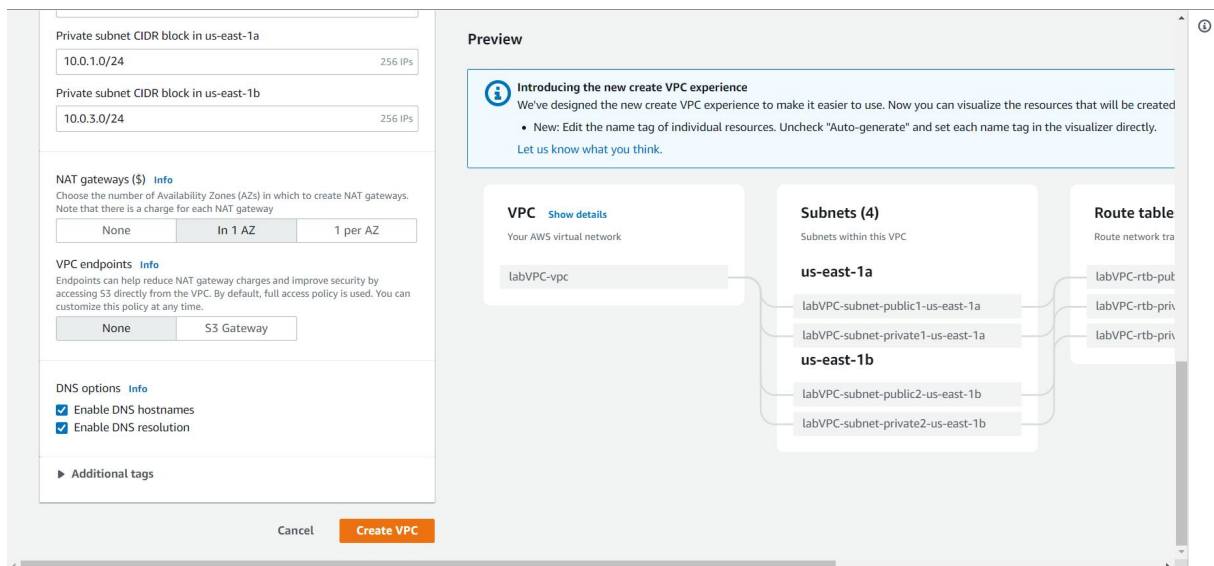
- New: Edit the name tag of individual resources. Uncheck "Auto-generate" and set each name tag in the visualizer directly.

Let us know what you think.

VPC [Show details](#)
Your AWS virtual network
labVPC-vpc

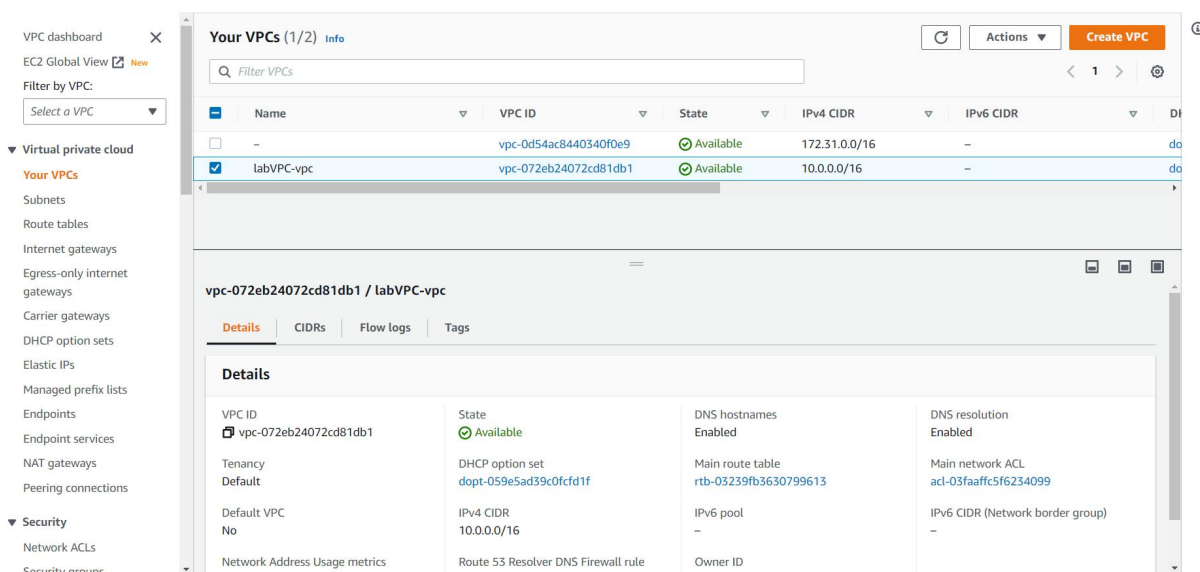
Subnets (4)
Subnets within this VPC
us-east-1a
labVPC-subnet-public1-us-east-1a
labVPC-subnet-private1-us-east-1a
us-east-1b
labVPC-subnet-public2-us-east-1b
labVPC-subnet-private2-us-east-1b

Route table
Route network tra
labVPC-rtb-pub
labVPC-rtb-priv
labVPC-rtb-priv



Hình 10: Trang VPC

Set up VPC như hình sau đó nhấn **create VPC**. Hiện thị VPC vừa mới tạo.



Hình 11: Trang VPC vừa tạo

4.2.2. Tạo Security groups:

Chọn Security groups trong thanh bên trái của dịch vụ VPC. Chọn create security group và nhập thông tin như hình.

VPC > Security Groups > Create security group

Create security group [Info](#)

A security group acts as a virtual firewall for your instance to control inbound and outbound traffic. To create a new security group, complete the fields below.

Basic details

Security group name [Info](#)

WebSecurityGroupLab

Name cannot be edited after creation.

Description [Info](#)

Allows SSH access to developers

VPC [Info](#)

vpc-072eb24072cd81db1

Inbound rules [Info](#)

This security group has no inbound rules.

Add rule

VPC [Info](#)

vpc-072eb24072cd81db1

Inbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Source Info	Description - optional Info	
All traffic	All	All	Anywh... 0.0.0.0/0		Delete
SSH	TCP	22	My IP 42.112.79.145/32		Delete
HTTP	TCP	80	Anywh... 0.0.0.0/0		Delete
HTTPS	TCP	443	Anywh... 0.0.0.0/0		Delete

Add rule

Outbound rules [Info](#)

Outbound rules [Info](#)

Type Info	Protocol Info	Port range Info	Destination Info	Description - optional Info	
All traffic	All	All	Anywhere... <input type="text" value="0.0.0.0/0"/>		Delete

[Add rule](#)

Tags - optional
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.

No tags associated with the resource.

[Add new tag](#)
You can add up to 50 more tags

Cancel [Create security group](#)

Hình 12: Trang security groups

Nhấp create security group và hiển thị.

Security group (sg-0a1b4bb5d6c774f21 | WebSecurityGroupLab) was created successfully

[Details](#)

VPC > Security Groups > sg-0a1b4bb5d6c774f21 - WebSecurityGroupLab

sg-0a1b4bb5d6c774f21 - WebSecurityGroupLab [Actions](#)

Details

Security group name WebSecurityGroupLab	Security group ID sg-0a1b4bb5d6c774f21	Description WebSecurityGroupLab	VPC ID vpc-072eb24072cd81db1
Owner 169844420023	Inbound rules count 4 Permission entries	Outbound rules count 1 Permission entry	

[Inbound rules](#) [Outbound rules](#) [Tags](#)

[You can now check network connectivity with Reachability Analyzer](#) [Run Reachability Analyzer](#)

Inbound rules (4) [Manage tags](#) [Edit inbound rules](#)

< 1 > ⚙

Hình 13: security groups sau khi tạo

4.2.3. Tạo EC2:

Lựa chọn dịch vụ EC2 của AWS chọn launch instances và nhập thông tin như hình.

Name and tags [Info](#)

Name
myweb [Add additional tags](#)

▼ Application and OS Images (Amazon Machine Image) [Info](#)
An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below

Quick Start

Amazon Linux macOS Ubuntu Windows Red Hat S [Browse more AMIs](#)
Including AMIs from AWS, Marketplace and the Community

Amazon Machine Image (AMI)
Ubuntu Server 22.04 LTS (HVM), SSD Volume Type
ami-0574da719dca65348 (64-bit (x86)) / ami-0e2b332e63c56bcb5 (64-bit (Arm))
Virtualization: hvm ENA enabled: true Root device type: ebs Free tier eligible

▼ Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)
ami-0574da719dca65348

Virtual server type (instance type)
t2.small

Firewall (security group)
WebSecurityGroupLab

Storage (volumes)
1 volume(s) - 10 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and

[Cancel](#) [Launch instance](#)

Canonical, Ubuntu, 22.04 LTS, amd64 jammy image build on 2022-12-01

Architecture [AMI ID](#)
64-bit (x86) ami-0574da719dca65348 [Verified provider](#)

▼ Instance type [Info](#)

Instance type
t2.small
Family: t2 1 vCPU 2 GiB Memory
On-Demand Linux pricing: 0.023 USD per Hour
On-Demand Windows pricing: 0.032 USD per Hour [Compare instance types](#)

▼ Key pair (login) [Info](#)
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.

Key pair name - *required*
mywebdeploy [Create new key pair](#)

▼ Network settings [Info](#)

▼ Summary

Number of instances [Info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)
ami-0574da719dca65348

Virtual server type (instance type)
t2.small

Firewall (security group)
WebSecurityGroupLab

Storage (volumes)
1 volume(s) - 10 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million IOs, 1 GB of snapshots, and

[Cancel](#) [Launch instance](#)

Network settings [info](#)

VPC - *required* [info](#)
vpc-072eb24072cd81db1 (labVPC-vpc)
10.0.0.0/16

Subnet [info](#)
subnet-0ff646afca9832536 labVPC-subnet-public1-us-east-1a
VPC: vpc-072eb24072cd81db1 Owner: 169844420023
Availability Zone: us-east-1a IP addresses available: 250 CIDR: 10.0.0.0/24

Auto-assign public IP [info](#)
Disable

Firewall (security groups) [info](#)
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
☐ Create security group ☒ Select existing security group

Common security groups [info](#)
Select security groups
WebSecurityGroupLab sg-0a1b4bb5d6c774f21 X
VPC: vpc-072eb24072cd81db1
Security groups that you add or remove here will be added to or removed from all your network interfaces.

► Advanced network configuration

Configure storage [info](#) [Advanced](#)

1x 10 GiB gp2 Root volume (Not encrypted)

Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage X

Add new volume

The selected AMI contains more instance store volumes than the instance allows. Only the first 0 instance store volumes from the AMI will be accessible from the instance

0 x File systems [Edit](#)

► Advanced details [info](#)

Summary

Number of instances [info](#)
1

Software Image (AMI)
Canonical, Ubuntu, 22.04 LTS, ...[read more](#)
ami-0574da719dca65348

Virtual server type (instance type)
t2.small

Firewall (security group)
WebSecurityGroupLab

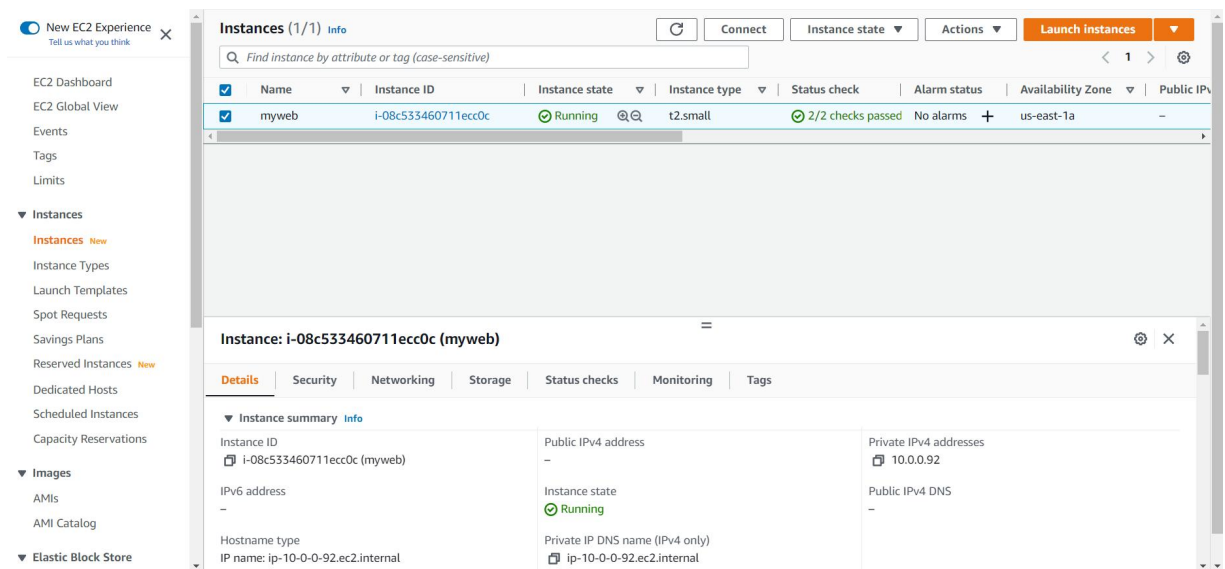
Storage (volumes)
1 volume(s) - 10 GiB

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 30 GiB of EBS storage, 2 million I/Os, 1 GB of snapshots, and X

Cancel [Launch Instance](#)

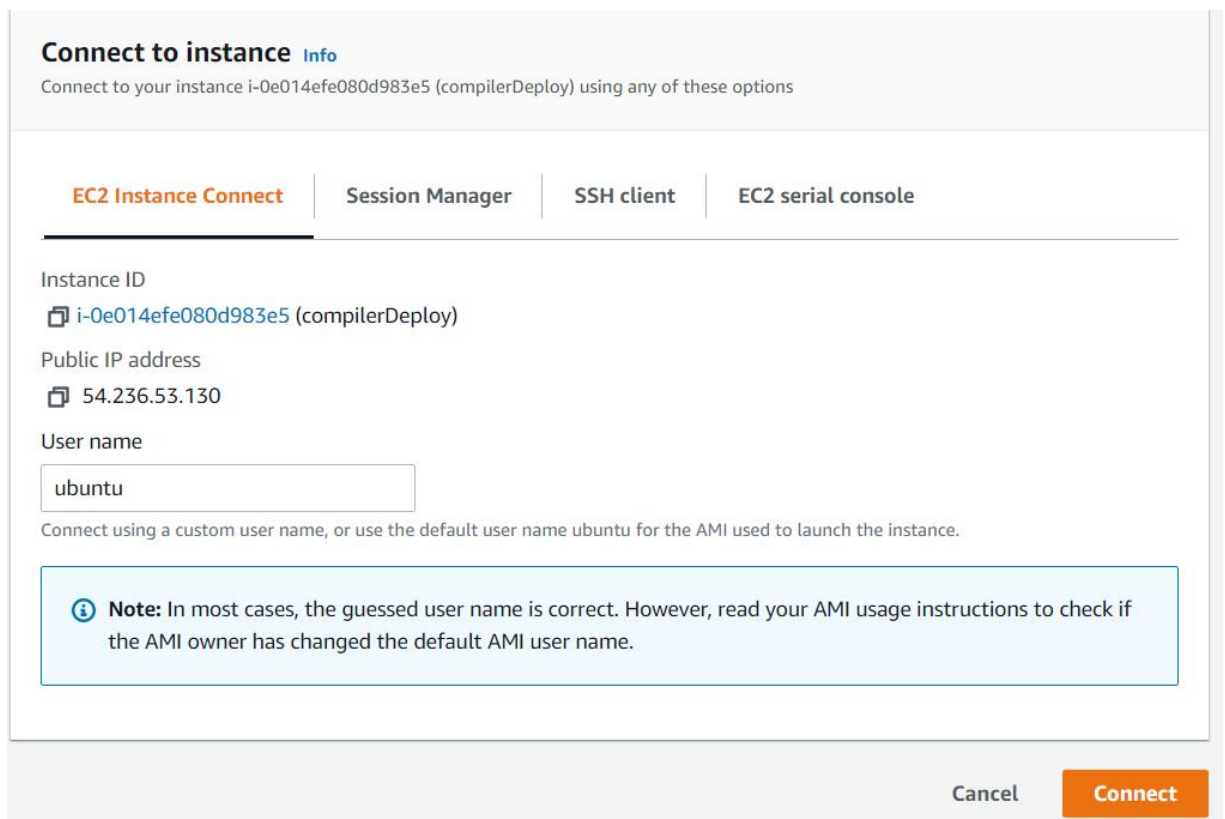
Hình 14: Tạo dịch vụ EC2

Sau đó nhấn launch instance



Hình 15: Dịch vụ EC2 sau khi tạo

Nhấn vào connect hiển thị trang như hình nhấn vào connect



```

Welcome to Ubuntu 22.04.1 LTS (GNU/Linux 5.15.0-1023-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Tue Nov 29 11:04:52 UTC 2022

System load:          0.333984375
Usage of /:           70.1% of 9.51GB
Memory usage:         15%
Swap usage:           0%
Processes:            107
Users logged in:      0
IPv4 address for br-07850556080b: 172.18.0.1
IPv4 address for docker0: 172.17.0.1
IPv4 address for eth0: 172.31.56.160

5 updates can be applied immediately.
To see these additional updates run: apt list --upgradable

The list of available updates is more than a week old.
To check for new updates run: sudo apt update

Last login: Tue Nov 29 11:04:53 2022 from 18.206.107.28
ubuntu@ip-172-31-56-160:~$

```

Hình 16: Connect EC2 vừa tạo

Sau khi hiển thị lên màn hình console thì cài docker và tải đồ án trên git về bằng những lệnh sau:

```
sudo apt-get update
```

```
sudo apt-get upgrade
```

```
sudo apt-get install \
```

```
    apt-transport-https \
```

```
    ca-certificates \
```

```
    curl \
```

```
    gnupg-agent \
```

```
    software-properties-common
```

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

```
apt-cache madison docker-ce
```

```
sudo apt-get install docker-ce docker-ce-cli containerd.io
```

```
sudo apt install docker.io
```

```
sudo apt install docker-compose
```

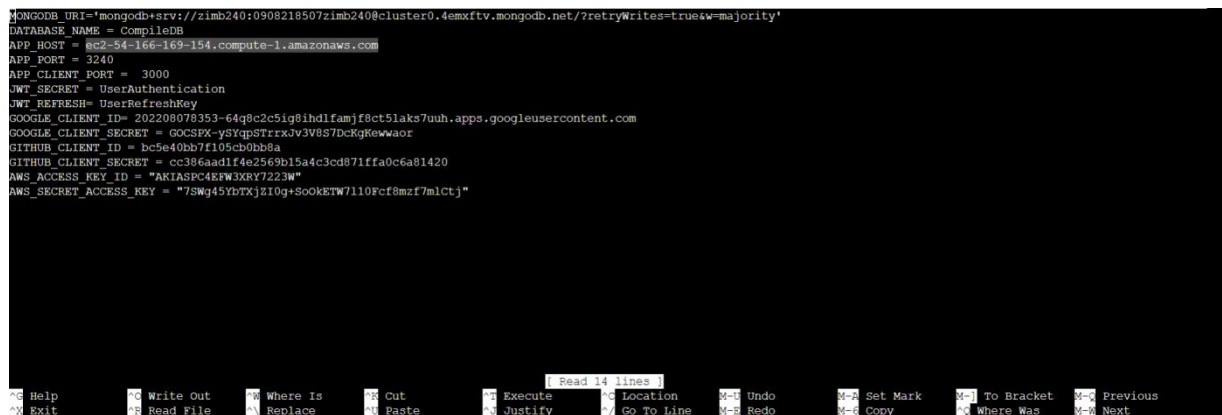
```
git clone https://github.com/Code-App-HCMUTE/CompilerGo
```

```
cd CompilerGo
```

```
cd server
```

```
mkdir temp
```

nano .env sẽ hiển thị như hình sau đó thay đổi APP_HOST thành Public IPv4 DNS



```
MONGODB_URI="mongodb+srv://zimb240:0908218507zimb240@cluster0.4emxftv.mongodb.net/?retryWrites=true&w=majority"
DATABASE_NAME = CompiledDB
APP_HOST = ec2-54-166-169-154.compute-1.amazonaws.com
APP_PORT = 3240
APP_CLIENT_PORT = 3000
JWT_SECRET = UserAuthentication
JWT_REFRESH = UserRefreshKey
GOOGLE_CLIENT_ID= 202208078353-64q8c2c5ig8ihd1famjf8ct5laks7uuh.apps.googleusercontent.com
GOOGLE_CLIENT_SECRET = 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
GITHUB_CLIENT_ID = bc5e40bb7f105cb0bb8a
GITHUB_CLIENT_SECRET = cc386aad1f4e2569b15a4c3cd871ffa0c6a81420
AWS_ACCESS_KEY_ID = "AKIASPC4EFW3KRY7223W"
AWS_SECRET_ACCESS_KEY = "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"
```

Hình 17: File .env bên server

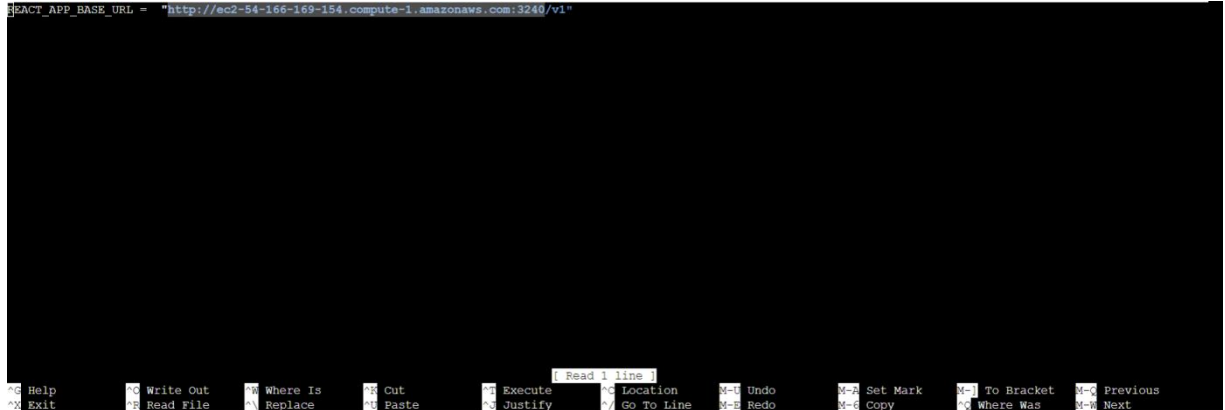
Sau đó lưu file

Sau đó dùng lệnh:

```
cd ..
```

```
cd client
```

nano .env sau đó thay đổi giá trị trong ngoặc thành Public IPv4 DNS:3240/v1



Hình 18: File .env bên client

Lưu file sau đó thực hiện tiếp tục lệnh:

`cd ..`

Sau đó dùng lệnh:

`sudo docker-compose build`

`sudo docker-compose up`

Sau đó truy cập vào đường link “`http://Public IPv4 DNS:3000`” ví dụ

Public IPv4 DNS = `ec2-54-236-53-130.compute-1.amazonaws.com`

Thì nhập vào đường link “`http://ec2-54-236-53-130.compute-1.amazonaws.com:3000`” để truy cập vào trang web và sử dụng.

4.2.4. Tạo ra DynamoDB:

Vào dịch vụ dynamodb của aws, chọn table, chọn create table và nhập như hình.

DynamoDB > Tables > Create table

Create table

Table details [Info](#)

DynamoDB is a schemaless database that requires only a table name and a primary key when you create the table.

Table name
This will be used to identify your table.

users

Between 3 and 255 characters, containing only letters, numbers, underscores (_), hyphens (-), and periods (.).

Partition key
The partition key is part of the table's primary key. It is a hash value that is used to retrieve items from your table and allocate data across hosts for scalability and availability.

_id String

1 to 255 characters and case sensitive.

Sort key - optional
You can use a sort key as the second part of a table's primary key. The sort key allows you to sort or search among all items sharing the same partition key.

Enter the sort key name String

1 to 255 characters and case sensitive.

Table settings

Capacity mode	Provisioned	Yes
Read capacity	5 RCU	Yes
Write capacity	5 WCU	Yes
Auto scaling	On	Yes
Local secondary indexes	-	No
Global secondary indexes	-	Yes
Encryption key management	Owned by Amazon DynamoDB	Yes
Table class	DynamoDB Standard	Yes

Tags
Tags are pairs of keys and optional values, that you can assign to AWS resources. You can use tags to control access to your resources or track your AWS spending.

No tags are associated with the resource.

Add new tag

You can add 50 more tags.

Cancel

Create table

Hình 19: Tạo bảng trong DynamoDB

Tạo ra 2 bảng users và practice như hình.

Tables (5) Info							
<input type="text" value="Find tables by table name"/>				Any table tag		< 1 >	
	Na...	Status	Partition key	Sort key	Indexes	Read capacity mode	Write capacity mode
<input type="checkbox"/>	Person	Active	FirstName (S)	LastName (S)	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)
<input checked="" type="checkbox"/>	practice	Active	_id (S)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)
<input type="checkbox"/>	user	Active	email_id (S)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)
<input checked="" type="checkbox"/>	users	Active	_id (S)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)
<input type="checkbox"/>	users1	Active	id (N)	-	0	Provisioned with auto scaling (1)	Provisioned with auto scaling (1)

Hình 20: Table vừa tạo

CHƯƠNG 5. KẾT LUẬN

5.1. Kết quả đạt được:

Về cơ bản nhóm tự nhận xét chương trình tạo ra đã giải quyết được những yêu cầu đề ra. Nhóm đã sử dụng reactjs và nodejs để xây dựng nên giao diện website và thực hiện được các tính năng đề ra như đăng nhập, đăng ký được tài khoản, compile code cùng nhiều chức năng khác mà yêu cầu đã đề ra. Hiểu rõ hơn về ngôn ngữ đang sử dụng. Bên cạnh đó, nhóm còn nâng cao kỹ năng làm việc nhóm, hiểu rõ hơn về các thành viên. Từ đó, nâng cao được nhiều kỹ năng cho bản thân mình.

5.2. Định hướng phát triển:

Nếu có nhiều thời gian hơn nhóm sẽ tiếp tục phát triển theo một số định hướng như sau:

- Tối ưu hóa code để thời gian chạy tốt hơn, dễ hiểu hơn thuận tiện cho việc phát triển sau này.
- Thiết kế lại giao diện đẹp hơn, các tính năng theo tác dễ hơn.
- Cấu trúc file dễ hiểu, dễ nhìn hơn.
- Tìm ra những trường hợp bị lỗi và sửa lại chúng. Sau đó phát triển thêm những tính năng mới.
- Phát triển trang web dành cho cả mobile

5.3. Ưu điểm, hạn chế:

Bên cạnh việc xây dựng trang web thì nhóm em nhận thấy được những ưu nhược điểm như sau.

5.3.1. Ưu điểm:

Xây dựng hoàn thiện trang giao diện web với đầy đủ tính năng đưa ra một cách hợp lý, phù hợp với người sử dụng để thao tác. Có thể compile được với nhiều ngôn ngữ, có bài tập để luyện tập với từng ngôn ngữ.

5.3.2. Hạn chế:

Vẫn còn lỗi ở một vài trường hợp có thể nhóm vẫn chưa tìm ra. Chỉ phù hợp sử dụng được cho website và chưa thiết kế để phù hợp với điện thoại.

TÀI LIỆU THAM KHẢO

- [1]. <https://aws.amazon.com/> , trang web chính thức của aws.
- [2]. <https://viblo.asia/p/tim-hieu-ve-amazon-ec2-maGK7jRe5j2> , Tìm hiểu về Amazon EC2?
- [3]. <https://viblo.asia/p/reactjs-uu-diem-va-nhuoc-diem-V3m5WzexlO7> , Bắt đầu với Reactjs.
- [4]. <https://reactjs.org/docs/add-react-to-a-website.html> , hướng dẫn cụ thể về reactjs.
- [5]. <https://docs.docker.com/get-started/> , hướng dẫn của docker
- [6]. <https://viblo.asia/p/gioi-thieu-ve-dynamodb-phan-1-jamoG87nMz8P>, “Giới thiệu về DynamoDB”, Lê Trung Kiên