**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**NĂM HỌC 2021 – 2022**

**ĐỀ TÀI**

**XÂY DỰNG DIỄN ĐÀN LẬP TRÌNH ONLINE TẠI PHÂN HIỆU**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TẠI TP. HỒ CHÍ MINH**

**Sinh viên thực hiện:**

|  |  |  |
| --- | --- | --- |
| **Võ Đoàn Hoàng Long** | **CQ.60.CNTT** | **Khoa: CNTT** |
| Nguyễn Thanh Hòa | CQ.60.CNTT | Khoa: CNTT |
| Bùi Văn Tân | CQ.60.CNTT | Khoa: CNTT |
| Nguyễn Ngọc Vũ Triều | CQ.60.CNTT | Khoa: CNTT |
| Đoàn Phước Nhật | CQ.60.CNTT | Khoa: CNTT |

**Người hướng dẫn:** ThS. Trần Thị Dung

TP. HCM, 2022

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**BÁO CÁO TỔNG KẾT**

**ĐỀ TÀI NGHIÊN CỨU KHOA HỌC CỦA SINH VIÊN**

**NĂM HỌC 2021 – 2022**

**ĐỀ TÀI**

**XÂY DỰNG DIỄN ĐÀN LẬP TRÌNH ONLINE TẠI PHÂN HIỆU**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI TẠI TP. HỒ CHÍ MINH**

**Sinh viên thực hiện:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Võ Đoàn Hoàng Long** | **Nam, Nữ: Nam** | **Dân tộc: Kinh** | **CQ.60.CNTT** |
| **Khoa: CNTT** | **Năm thứ: 3/4** | **Ngành học: Công nghệ thông tin** | |
| Nguyễn Thanh Hòa | Nam, Nữ: Nam | Dân tộc: Kinh | CQ.60.CNTT |
| Khoa: CNTT | Năm thứ: 3/4 | Ngành học: Công nghệ thông tin | |
| Bùi Văn Tân | Nam, Nữ: Nam | Dân tộc: Kinh | CQ.60.CNTT |
| Khoa: CNTT | Năm thứ: 3/4 | Ngành học: Công nghệ thông tin | |
| Nguyễn Ngọc Vũ Triều | Nam, Nữ: Nam | Dân tộc: Kinh | CQ.60.CNTT |
| Khoa: CNTT | Năm thứ: 3/4 | Ngành học: Công nghệ thông tin | |
| Đoàn Phước Nhật | Nam, Nữ: Nam | Dân tộc: Kinh | CQ.60.CNTT |
| Khoa: CNTT | Năm thứ: 3/4 | Ngành học: Công nghệ thông tin | |

**Người hướng dẫn:** ThS. Trần Thị Dung

TPHCM, 2022

**MỤC LỤC**

[**MỤC LỤC i**](#_Toc104358626)

[**DANH MỤC HÌNH ẢNH iii**](#_Toc104358627)

[**DANH MỤC BẢNG BIỂU iv**](#_Toc104358628)

[**MỞ ĐẦU v**](#_Toc104358629)

[**TỔNG QUAN vi**](#_Toc104358630)

[**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 1**](#_Toc104358631)

[1.1. Công nghệ sử dụng 1](#_Toc104358632)

[1.1.1. ReactJs 1](#_Toc104358633)

[1.1.2. Apollo Client 1](#_Toc104358634)

[1.1.3. Hasura 1](#_Toc104358635)

[1.1.4. GraphQL 1](#_Toc104358636)

[1.1.5. Docker 2](#_Toc104358637)

[1.1.6. PostgresQL 2](#_Toc104358638)

[1.2. Ngôn ngữ lập trình 2](#_Toc104358639)

[1.2.1. JavaScript 2](#_Toc104358640)

[1.2.2. Golang 2](#_Toc104358641)

[**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 3**](#_Toc104358642)

[2.1. Tổng quan về hệ thống 3](#_Toc104358643)

[2.2. Biểu đồ quan hệ (Entity relationship diagram) 3](#_Toc104358644)

[2.3. Biểu đồ Usecase 4](#_Toc104358645)

[2.4. Biểu đồ hoạt động 5](#_Toc104358646)

[2.4.1. Chức năng tạo bài luyện tập 5](#_Toc104358647)

[2.4.2. Chức năng tạo khóa học 6](#_Toc104358648)

[2.4.3. Chức năng tạo thử thách 6](#_Toc104358649)

[**CHƯƠNG 3: XÂY DỰNG GIAO DIỆN CHƯƠNG TRÌNH 8**](#_Toc104358650)

[3.1. Giao diện đăng nhập 8](#_Toc104358651)

[3.2. Giao diện danh sách bài tập 8](#_Toc104358652)

[3.3. Giao diện tạo bài tập 9](#_Toc104358653)

[3.4. Giao diện danh sách bài viết thảo luận 9](#_Toc104358654)

[3.5. Giao diện tạo bài viết mới 9](#_Toc104358655)

[3.6. Giao diện danh sách khóa học 10](#_Toc104358656)

[3.7. Giao diện tạo khóa học 10](#_Toc104358657)

[3.8. Giao diện học tập 11](#_Toc104358658)

[3.9. Giao diện từng bài học 11](#_Toc104358659)

[3.10. Giao diện luyện tập 12](#_Toc104358660)

[3.11. Giao diện làm bài 12](#_Toc104358661)

[3.12. Giao diện danh sách thử thách 13](#_Toc104358662)

[3.13. Giao diện các bài viết thảo luận 13](#_Toc104358663)

[**KẾT LUẬN VÀ KIẾN NGHỊ 15**](#_Toc104358664)

[Kết quả đạt được 15](#_Toc104358665)

[Nhược điểm 15](#_Toc104358666)

[Hướng phát triển 15](#_Toc104358667)

[**TÀI LIỆU THAM KHẢO 16**](#_Toc104358668)

**DANH MỤC HÌNH ẢNH**

[Hình 2. 1 Biểu đồ quan hệ (Entity relationship diagram) 5](#_Toc104318076)

[Hình 2. 2 Sơ đồ use case 5](#_Toc104318077)

[Hình 2. 3 Sơ đồ chức năng tạo bài luyện tập 6](#_Toc104318078)

[Hình 2. 4 Sơ đồ chức năng tạo khóa học 7](#_Toc104318079)

[Hình 2. 5 Sơ đồ chức năng tạo thử thách 8](#_Toc104318080)

[Hình 3. 1 Giao diện đăng nhập 9](#_Toc104318081)

[Hình 3. 2 Giao diện danh sách bài tập 9](#_Toc104318082)

[Hình 3. 3 Giao diện tạo bài tập 10](#_Toc104318083)

[Hình 3. 4 Giao diện danh sách bài viết thảo luận 10](#_Toc104318084)

[Hình 3. 5 Giao diện tạo bài viết mới 11](#_Toc104318085)

[Hình 3. 6 Giao diện danh sách khóa học 11](#_Toc104318086)

[Hình 3. 7 Giao diện tạo khóa học 12](#_Toc104318087)

[Hình 3. 8 Giao diện học tập 12](#_Toc104318088)

[Hình 3. 9 Giao diện từng bài học 13](#_Toc104318089)

[Hình 3. 10 Giao diện luyện tập 13](#_Toc104318090)

[Hình 3. 11 Giao diện làm bài 14](#_Toc104318091)

[Hình 3. 12 Giao diện danh sách thử thách 14](#_Toc104318092)

[Hình 3. 13 Giao diện các bài viết thảo luận 15](#_Toc104318093)

[Hình 3. 14 Giao diện bài viết 15](#_Toc104318094)

**DANH MỤC BẢNG BIỂU**

[Bảng 2. 1 Bảng các tác nhân 6](#_Toc104318107)

**MỞ ĐẦU**

Thời đại công nghệ số, bên cạnh tiếng mẹ để, ngoại ngữ thì ngôn ngữ lập trình dần trở thành như một loại ngôn ngữ mà hầu như bây giờ đã là gì đó không quá là xa lạ khi nói về nữa. Với lượng nhu cầu lớn như vậy thì các website lập trình online cũng dần trở nên thịnh hành. Như trên thế giới hiện có rất nhiều web lập trình online mà không cần cài thêm bất cứ gì cho máy tính như: Codepad, Paiza, Replit, W3school, Onlinegdb, …

Việt Nam – là một nước cũng đang trên đà phát triển và hội nhập Thế giới, cũng không thua kém gì ví dụ như: CodeLearn, F8fullstack, LuyenCode, … Cho thấy được hội nhập một cách nhanh chóng của Việt Nam. Đồng thời, cũng thấy được nhu cầu lập trình và học lập trình cũng cao không kém cạnh gì các lĩnh vực khác.

Trường Đại học Giao thông Vận tải – ngôi trường danh giá, đào tào hơn ngàn sinh viên mỗi năm, cũng đào tạo và bồi dưỡng các ngành về mặt công nghệ số cũng có website luyện code riêng là: laptrinhonline.club. Nhằm tạo một sản phẩn riêng cho UTC2 – Phân hiệu nhóm đưa tới một website diễn đàn lập trình online với mong muốn hỗ trợ thêm nhiều sinh viên định hình, mài giũa khả năng và tư duy lập trình.

**TỔNG QUAN**

1. Lý do chọn đề tài

Trong bất kỳ một quá trình giáo dục nào thì mục tiêu chính của nó cũng là nhằm tạo ra những biến đổi nhất định đối với người học. Và việc học ngày càng trở nên khô khan nếu đó chỉ là những lý thuyết trong sách vở. Với xã hội không ngừng phát triển, công nghệ ngày càng lấn áp thị trường, lúc này đây “thế giới số” không còn là cụm từ quá khó hiểu so với nhiều người, nhất là giới trẻ. Đúng vậy, học là phải đi đôi với hành, và việc học lập trình cũng như vậy.

Hiện nay website hỗ trợ công cụ lập trình online đang dẫn xuất hiện trong những năm gần. Những website đó nhằm hỗ trợ cho các lập trình viên giải quyết nhanh một số vấn đề nhỏ về lập trình mà không cần cài đặt gì rườm rà.

Website diễn đàn lập trình online sẽ giúp cho việc học lập trình sẽ trở nên một cách trực quan, thu hút hơn với giao diện và tính năng gần gũi, dễ sử dụng.

Với mục tiêu trên, việc xây dựng 1 website diễn đàn lập trình online là vô cùng cần thiết.

1. Mục tiêu nghiên cứu

* Giúp giảng viên trong khoa có thể tạo diễn đàn để sinh viên có thể dễ dàng luyện tập, trao đổi về vấn đề lập trình.
* Mở các lớp học trực tiếp nhằm bổ túc kiến thức nền tảng.
* Quản lý, kiểm duyệt kiến thức có chọn lọc để đưa đến cho sinh viên nói riêng và người lập trình viên tương lai nói chung.

1. Đối tượng nghiên cứu và phạm vi nghiên cứu

* Các nội dung của bài học của các môn học đang được triển khai ở ngành CNTT của bộ môn CNTT trường đại học Giao thông vận tải phân hiệu tại TP.HCM.
* Yêu cầu của giảng viên.

1. Phương pháp nghiên cứu

* Khảo sát nhu cầu thực tế của sinh viên.
* Tham khảo các trang web đã có.
* Luyện viết mã thông qua các bài tập đã được giảng viên đăng lên hoặc tham gia cuộc thi định kỳ được tổ chức trên diễn đàn.
* Lập trình Website trên Visual Studio Code.
* Phương pháp thu thập số liệu.
* Phương pháp điều tra
* Phương pháp thực nghiệm
* Phương pháp quan sát.

**CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

* 1. **Công nghệ sử dụng**

1. **ReactJs**

ReactJS là một thư viện JavaScript mã nguồn mở được thiết kế bởi Facebook để tạo ra những ứng dụng web hấp dẫn, nhanh và hiệu quả với mã hóa tối thiểu. Mục đích cốt lõi của ReactJS không chỉ khiến cho trang web phải thật mượt mà còn phải nhanh, khả năng mở rộng cao và đơn giản. [1]

Sức mạnh của nó xuất phát từ việc tập trung vào các thành phần riêng lẻ. Chính vì vậy, thay vì làm việc trên toàn bộ ứng dụng web, ReactJS cho phép một developer có thể phá vỡ giao diện người dùng phức tạp thành các thành phần đơn giản hơn.

1. **Apollo Client**

Apollo Client là một thư viện quản lý trạng thái toàn diện cho JavaScript cho phép bạn quản lý cả dữ liệu cục bộ và từ xa với GraphQL. Sử dụng nó để tìm nạp, lưu vào bộ nhớ cache và sửa đổi dữ liệu ứng dụng, tất cả trong khi tự động cập nhật giao diện người dùng của bạn. Apollo Client giúp bạn cấu trúc mã theo cách tiết kiệm, có thể dự đoán và khai báo phù hợp với thực tiễn phát triển hiện đại. Thư viện cốt lõi @apollo/clientcung cấp tích hợp sẵn với React và cộng đồng Apollo lớn hơn duy trì tích hợp cho các lớp chế độ xem phổ biến khác. [10]

1. **Hasura**

Hasura là một máy chủ GraphQL cực nhanh cung cấp cho bạn các API GraphQL thời gian thực, tức thì qua Postgres, với các trình kích hoạt webhook trên các sự kiện cơ sở dữ liệu và các lược đồ từ xa cho logic nghiệp vụ.

Hasura giúp bạn xây dựng các ứng dụng GraphQL được Postgres hỗ trợ hoặc từng bước chuyển sang GraphQL cho các ứng dụng hiện có sử dụng Postgres. [6]

1. **GraphQL**

GraphQL là một ngôn ngữ thao tác và truy vấn dữ liệu mã nguồn mở cho các API và một thời gian chạy để thực hiện các truy vấn với dữ liệu hiện có. Nó cung cấp một cách tiếp cận để phát triển các API web và đã được so sánh và đối chiếu với REST và các kiến ​​trúc dịch vụ web khác. Nó cho phép khách hàng xác định cấu trúc của dữ liệu được yêu cầu và cấu trúc tương tự của dữ liệu được trả về từ máy chủ, do đó ngăn không cho trả về một lượng dữ liệu quá lớn. GraphQL được Facebook phát triển nội bộ vào năm 2012 trước khi phát hành công khai vào năm 2015.[2]

1. **Docker**

Docker là một dự án mã nguồn mở giúp tự động triển khai các ứng dụng Linux và Windows vào trong các container ảo hóa. Docker cung cấp một lớp trừu tượng và tự động ảo hóa dựa trên Linux. Docker sử dụng những tài nguyên cô lập của Linux như cgroups, kernel, quản lý tệp để cho phép các container chạy độc lập bên trong một thực thể Linux. Các thay đổi được lưu trữ trong các Docker image, các lớp tệp hệ thống được tạo ra và lưu lại dựa theo từng lớp (layer). Điều này giúp cho Docker Image giảm dung lượng đáng kể so với máy ảo (Virtual Machine). Các ứng dụng muốn chạy bằng Docker phải là ứng dụng chạy được trên Linux. Gần đây, Docker có hỗ trợ thêm việc chạy ứng dụng Windows trong các Windows container. [9]

1. **PostgresQL**

PostgreSQL là một hệ quản trị cơ sở dữ liệu quan hệ và đối tượng dựa trên POSTGRES, bản 4.2, được khoa điện toán của đại học California tại Berkeley phát triển. POSTGRES mở đường cho nhiều khái niệm quan trọng mà các hệ quản trị dữ liệu thương mại rất lâu sau mới có. PostgreSQL là một chương trình mã nguồn mở xây dựng trên mã nguồn ban đầu của đại học Berkeley. Nó theo chuẩn SQL99 và có nhiều đặc điểm hiện đại. [5]

* 1. **Ngôn ngữ lập trình**

1. **JavaScript**

JavaScript là ngôn ngữ kịch bản cho phép tạo ra trang web động - cập nhật nội dung theo ngữ cảnh, điều khiển đa phương tiện, hoạt cảnh các hình ảnh. JavaScript là ngôn ngữ lập trình phổ biến nhất trên thế giới trong suốt 20 năm qua. [4]

1. **Golang**

Go là một ngôn ngữ lập trình mới do Google thiết kế và phát triển. Về mặt cú pháp thì Go rất giống ngôn ngữ C, tuy nhiên nó có nhiều thay đổi trong thiết kế để an toàn hơn và có cú pháp súc tích và dễ đọc. Nó được kỳ vọng sẽ giúp ngành công nghiệp phần mềm khai thác tối đa nền tảng đa lõi của bộ vi xử lý và hoạt động đa nhiệm tốt hơn. [11]

**CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG**

1. **Tổng quan về hệ thống**

Website diễn đàn lập trình online cho phép sinh viên nói riêng và người dùng nói chung luyện lập trình trên máy tính với các bài toán được cập nhật sẵn, các thông tin của hệ thống được đặt ở một sever cố định. Tập trung vào ba quy trình chính:

* Quy trình soạn đề: Dành cho giảng viên:

Giảng viên: Soạn các đề, có thể chọn từ ngân hàng câu hỏi hoặc tải từ file word/excel. Giáo viên sử dụng đề này cho các lớp mình phụ trách để thi giữa kỳ hoặc cho sinh viên ôn tập.

* Quy trình tổ chức các thử thách:

Dành cho giảng viên phụ trách tạo các thử thách theo tuần, tháng, quý cho các sinh viên thi đua với nhau. Giảng viên lên danh sách, tạo thử thách, chọn thời gian và tạo đề, giáo viên có thể chọn cho xem/không cho xem đáp án sau khi hoàn thành bài làm.

Trong thời gian diễn ra thử sách, sinh viên vào phòng để hoàn thành các chuỗi bài tập và giảng viên đưa ra. Đồng thời sinh viên phải tìm các cách để tối ưu về code của mình.

Khi hết thời gian làm bài, hệ thống thu bài làm, tính toán trả kết quả cho sinh viên và trả kết quả thống kê cho giảng viên.

* Quy trình ôn tập của sinh viên:

Dành cho sinh viên ôn tập. Các bộ đề có sẵn do giáo viên soạn. Sinh viên có thể ôn tập nhiều lần theo ý muốn. Hệ thống sẽ thống kê kết quả và đánh giá sau mỗi lần thi để sinh viên có thể biết được các phần mà bản thân cần cải thiện.

1. **Biểu đồ quan hệ (Entity relationship diagram)**

A picture containing timeline

Description automatically generated

Hình 2. 1 Biểu đồ quan hệ (Entity relationship diagram)

1. **Biểu đồ Usecase**

Diagram

Description automatically generated

Hình 2. 2 Sơ đồ use case

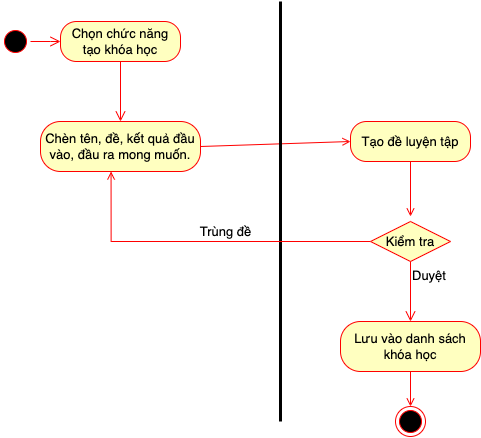
Các tác nhân

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên tác nhân** | **Quyền sử dụng** |
| 1 | Giảng viên | Quản lý bài luyện tập, quản lý cuộc thi, quản lý bài viết, quản lý khóa học. (Với quyền quản lý ở đây là tạo, cập nhật và xóa) |
| 2 | Người dùng | Tham gia các khóa học, luyện lập trình, trao đổi và đọc thêm tài liệu được cập nhật liên tục. |

Bảng 2. 1 Bảng các tác nhân

1. **Biểu đồ hoạt động**
2. **Chức năng tạo bài luyện tập**

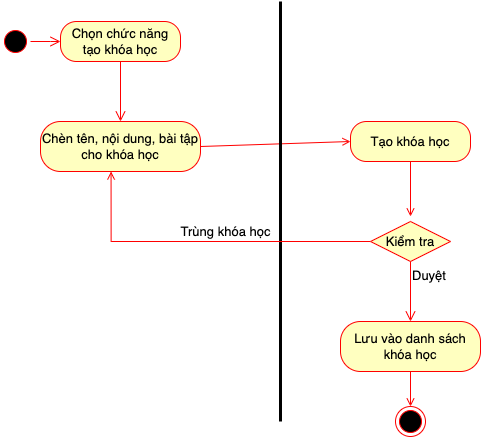
|  |  |
| --- | --- |
| Giảng viên | Website |



Hình 2. 3 Sơ đồ chức năng tạo bài luyện tập

1. **Chức năng tạo khóa học**

|  |  |
| --- | --- |
| Giảng viên | Website |



Hình 2. 4 Sơ đồ chức năng tạo khóa học

1. **Chức năng tạo thử thách**

|  |  |
| --- | --- |
| Giảng viên | Website |

Diagram

Description automatically generated

Hình 2. 5 Sơ đồ chức năng tạo thử thách

**CHƯƠNG 3: XÂY DỰNG GIAO DIỆN CHƯƠNG TRÌNH**

* 1. **Giao diện đăng nhập**
* Để sử dụng được website, người dùng cần đăng nhập vào hệ thống bằng email và pass đã đăng kí. Nếu chưa có tài khoản người dùng có thể đăng kí tài khoản bên dưới để có trải nghiệm tốt hơn.

A picture containing graphical user interface

Description automatically generated

Hình 3. 1 Giao diện đăng nhập

* 1. **Giao diện danh sách bài tập**
* Với quyền giảng viên; quản trị viên sẽ thêm sửa xóa các bài tập.

Graphical user interface, application

Description automatically generated

Hình 3. 2 Giao diện danh sách bài tập

* 1. **Giao diện tạo bài tập**
* Giao diện tạo bài tập cho sinh viên luyện tập. Giảng viên sẽ tạo đề, tạo các trường hợp, đầu ra, đầu vào mong muốn.

Graphical user interface, text, application

Description automatically generated

Hình 3. 3 Giao diện tạo bài tập

* 1. **Giao diện danh sách bài viết thảo luận**
* Giao diện danh sách bài viết thảo luận, giảng viên sẽ chỉnh sửa lại bài viết hoặc gỡ bài viết. Các bài viết tải lên cũng sẽ hiển thị ở giao diện này.

Graphical user interface, application

Description automatically generated

Hình 3. 4 Giao diện danh sách bài viết thảo luận

* 1. **Giao diện tạo bài viết mới**
* Giao diện tạo bài viết, giáo viên sẽ tạo bài viết trong trình soạn thảo văn bản. Giáo viên có thể đặt tiêu đề, chỉnh chữ đậm, nghiêng, gạch chân, chèn hình,… các chức năng đã được chuyển thành các biểu tượng cho trực quan hơn. Bên cạnh đó, giáo viên có thể xem trước được bài viết sau khi xong sẽ như thế nào thông qua màn hình bên cạnh. Sau đó, giáo viên nhấn vào nút “Tải lên” ở góc trên bên phải màn hình.

Text

Description automatically generated with medium confidence

Hình 3. 5 Giao diện tạo bài viết mới

* 1. **Giao diện danh sách khóa học**
* Giao diện danh sách khóa học, giảng viên sẽ chỉnh sửa lại khóa học hoặc xóa khóa học. Các khóa học tải lên cũng sẽ hiển thị ở giao diện này.

Graphical user interface, application

Description automatically generated

Hình 3. 6 Giao diện danh sách khóa học

* 1. **Giao diện tạo khóa học**
* Giao diện tạo khóa học, giảng viên sẽ tạo khóa học bằng cách thêm tiêu đề, ảnh bìa, mô tả về khóa học. Trong khóa học, giáo viên sẽ liên kết với phần bài tập trên để người dùng song song việc lý thuyết và thực hành.

Graphical user interface, application

Description automatically generated

Hình 3. 7 Giao diện tạo khóa học

* 1. **Giao diện học tập**
* Giao diện học tập, đây sẽ là nơi hiển thị trực quan các khóa học cho người dùng (sinh viên) học tập và hoàn thành các bài tập áp dụng trong mỗi khóa học đó. Giao diện chỉ cho phép học khi người dùng đã đăng nhập tài khoản vào website.

Graphical user interface, application

Description automatically generated

Hình 3. 8 Giao diện học tập

* 1. **Giao diện từng bài học**
* Giao diện từng bài học, người dùng sẽ đọc tài liệu vào áp dụng ngay kiến thức đó một cách trực quan.

Graphical user interface, website

Description automatically generated

Hình 3. 9 Giao diện từng bài học

* 1. **Giao diện luyện tập**
* Giao diện luyện tập, người dùng sẽ luyện tập lại kiến thức đã học theo từng cấp độ. Qua đó, sẽ có được cái nhìn tổng quan và tư duy sẽ được cải thiện theo từng cấp độ đó.

Graphical user interface, application

Description automatically generated

Hình 3. 10 Giao diện luyện tập

* 1. **Giao diện làm bài**
* Giao diện sẽ được chia đôi màn hình với bên trái sẽ đề bài và bên phải là giao diện lập trình. Website hỗ trợ hầu hết các ngôn ngữ lập trình phổ biến hiện nay như: C/C++, C#, JavaScript, Java, Ruby, Python, PHP, … Góc dưới bên phải sẽ là yêu cầu đầu vào đầu ra cho từng bài toán. Sau khi làm bài xong người dùng nhấp vào “Run Code” để kiểm tra kết quả, xem thời gian xử lý và bộ nhớ tiêu hao để tìm cách tối ưu hơn cho bài. Xong bài, người dùng nhấp vào “Submit” để nộp bài.

Graphical user interface, application

Description automatically generated

Hình 3. 11 Giao diện làm bài

* 1. **Giao diện danh sách thử thách**
* Giảng viên sẽ tạo các thử thách tương tự như các cuộc thi ngắn hạn để cho người học có nơi để thi thố, cạnh tranh để phát triển bản thân. Kết quả và xếp hạng sẽ được công bố khi các thử thách hết hạn.

A screenshot of a computer

Description automatically generated with medium confidence

Hình 3. 12 Giao diện danh sách thử thách

* 1. **Giao diện các bài viết thảo luận**
* Giao diện bài viết, người dùng sẽ đăng các bài viết chia sẻ kinh nghiệm, kiến thức, công nghệ mới ở đây. Các bài viết sẽ được chọn lọc và kiểm soát thông qua quản trị viên.

Graphical user interface, text, application, email, Teams

Description automatically generated

Hình 3. 13 Giao diện các bài viết thảo luận

Graphical user interface, text, application, email

Description automatically generated

Hình 3. 14 Giao diện bài viết

**KẾT LUẬN VÀ KIẾN NGHỊ**

**Kết quả đạt được**

Sau quá trình tìm hiểu và thực hiện, đề tài đã phần nào đáp ứng được những yêu cầu đã đặt ra ở mục tiêu đề ra:

* Xây dựng được website diễn đàn lập trình online với các quy trình chính:
* Quy trình soạn đề.
* Quy trình tổ chức cuộc thi hàng tuần, tháng, quý.
* Quy trình tổ chức các khóa học trực tuyến online.
* Quy trình tổ chức tạo các bài viết theo chủ để cho sinh viên trao đổi kiến thức.
* Quy trình ôn luyện thuật toán, các bài toán kinh điển cho sinh viên thông qua code trực tiếp trên trang web.

Hệ thống chạy ổn định với giao diện thân thiện, dễ sử dụng, tốc độ xử lí nhanh. Đồng thời website cũng thực hiện tốt trên các trình duyệt web phổ biến như hiện nay như: Chrome, Firefox, Opera.

**Nhược điểm**

* Do thời gian có hạn nên website có một vài tính năng chưa chạy ổn định.
* Dữ liệu bài tập còn khá ít.

**Hướng phát triển**

Từ những hạn chế trên, trong thời gian tới, nhóm sẽ bổ sung thêm dữ liệu, hoàn thiện các chức năng để có thể sử dụng vào thực tế.

**TÀI LIỆU THAM KHẢO**

[1]. Robin Wieruch, “The Road to React”, 2022

[2]. Robin Wieruch, “The Road to GraphQL”, 2018.

[3]. Basarat Ali Syed, “Beginning Node.js”, 2014.

[4]. Elisabeth Robson, Eric Freeman, “Headfirst JavaScript Programming”, 2014.

[5]. “PostgresQL”, <https://www.postgresql.org/docs>, ngày truy cập: 23/12/2021.

[6]. “Hasura”, <https://hasura.io/docs/latest/graphql/core/index>, ngày truy cập: 1/1/2022.

[7]. “Firebase”, <https://firebase.google.com/docs>, ngày truy cập: 1/1/2022.

[8]. “Judgeo CE”, <https://ce.judge0.com>, ngày truy cập: 1/1/2022.

[9]. “Docker”, <https://docs.docker.com>, ngày truy cập: 1/1/2022.

[10]. “Apollo”, <https://www.apollographql.com/docs/react>, ngày truy cập: 21/12/2021.

[11]. “Golang”, <https://go.dev/doc>, ngày truy cập: 29/11/2021.

[12]. “Axios”, <https://www.npmjs.com/package/axios>, ngày truy cập: 14/3/2022.

[13]. “Sass”, https://sass-lang.com/documentation, ngày truy cập: 1/1/2022.

[14]. “JWT”, https://en.wikipedia.org/wiki/JSON\_Web\_Token, ngày truy cập: 1/1/2022.