**ĐẠI HỌC ĐÀ NẴNG**

**TRƯỜNG ĐẠI HỌC SƯ PHẠM**

**KHOA TIN HỌC**

-----🙞🙜🕮🙞🙜-----



**BÁO CÁO MÔN HỌC**

**CƠ SỞ DỮ LIỆU NÂNG CAO**

**Đề tài:**

**XÂY DỰNG WEBSITE**

**CHIA SẺ KHÓA HỌC BẰNG API**

|  |  |  |  |
| --- | --- | --- | --- |
| **Giảng viên hướng dẫn** | **:** | **Võ Công Đình** |  |
| **Nhóm thực hiện** | **:** | **6** |  |
| **Sinh viên thực hiện** | **:** | **Nguyễn Hữu Phước**  **Lê Hữu Phước**  **Nguyễn Hữu Minh Quân**  **Nguyễn Dương Quý**  **Nguyễn Phú Quý**  **Nguyễn Khắc Tài**  **Lê Văn Tao** | **19CNTT2**  **19CNTT1**  **19CNTT1**  **19CNTT1**  **19CNTT1**  **19CNTT1**  **19CNTT2** |

**Đà Nẵ**

**Đà Nẵng - 2022**

MỤC LỤC

[BẢNG PHÂN CÔNG CÔNG VIỆC 1](#_Toc104640087)

[BẢNG TIẾN ĐỘ DỰ ÁN 2](#_Toc104640088)

[MỞ ĐẦU 4](#_Toc104640089)

[1. Lý do đề tài 4](#_Toc104640090)

[1.1. Lý do chọn cơ sở dữ liệu (API) 4](#_Toc104640091)

[1.2 Lý do chọn ứng dụng blog khóa học 4](#_Toc104640092)

[2. Mục đích của đề tài 4](#_Toc104640093)

[3. Đối tượng và phạm vi của đề tài 5](#_Toc104640094)

[4. Ứng dụng của đề tài 5](#_Toc104640095)

[5. Công cụ hỗ trợ 5](#_Toc104640096)

[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 6](#_Toc104640097)

[1.1 Giới thiệu về NodeJS 6](#_Toc104640098)

[1.1.1 NodeJS là gì? 6](#_Toc104640099)

[1.1.2 Lý do sử dụng NodeJS 6](#_Toc104640100)

[1.2 Giới thiệu về Rest API 7](#_Toc104640101)

[1.2.1 Rest API là gì? 7](#_Toc104640102)

[1.2.2 Cách thức hoạt động của Rest API 7](#_Toc104640103)

[1.3 Giới thiệu về VueJS 8](#_Toc104640104)

[1.3.1 VueJS là gì? 8](#_Toc104640105)

[1.3.2 Lý do sử dụng VueJS 8](#_Toc104640106)

[1.4 Khái niệm MongoDB 9](#_Toc104640107)

[1.5 Phần mềm Visual studio code 9](#_Toc104640108)

[CHƯƠNG 2: PHÂN TÍCH THIẾT KẾ 11](#_Toc104640109)

[2.1. Phân tích các chức năng của hệ thống 11](#_Toc104640110)

[2.2. Biểu đồ phân tích thiết kế hệ thống 13](#_Toc104640111)

[2.3. Biểu đồ tuần tự các chức năng 18](#_Toc104640112)

[2.4. Phân tích cơ sở dữ liệu 23](#_Toc104640113)

[KẾT LUẬN 26](#_Toc104640114)

[1. Bài học kinh nghiệm: 26](#_Toc104640115)

[2. Hướng nghiên cứu phát triển: 26](#_Toc104640116)

[TÀI LIỆU THAM KHẢO 27](#_Toc104640117)

DANH MỤC HÌNH ẢNH

[Hình 1.1 Mô hình hoạt động của Rest API 8](#_Toc104639982)

[Hình 2.1 Biểu đồ use-case chức năng hệ thống 12](#_Toc104639983)

[Hình 2.2 Sơ đồ hoạt động đăng kí thành viên 13](#_Toc104639984)

[Hình 2.3 Sơ đồ hoạt động đăng nhập 14](#_Toc104639985)

[Hình 2.4 Sơ đồ hoạt động chức năng đăng xuất 14](#_Toc104639986)

[Hình 2.5 Sơ đồ hoạt động đổi mật khẩu 15](#_Toc104639987)

[Hình 2.6 Sơ đồ hoạt động thêm khóa học 16](#_Toc104639988)

[Hình 2.7 Sơ đồ hoạt động sửa thông tin khóa học 17](#_Toc104639989)

[Hình 2.8 Sơ đồ hoạt động xóa khóa học 18](#_Toc104639990)

[Hình 2.9 Biểu đồ tuần tự chức năng tìm kiếm 18](#_Toc104639991)

[Hình 2.10 Biểu đồ tuần tự chức năng quản lí khóa học 19](#_Toc104639992)

[Hình 2.11 Biểu đồ tuần tự chức năng thêm mới khóa học 19](#_Toc104639993)

[Hình 2.12 Biểu đồ tuần tự chức năng sửa thông tin khóa học 20](#_Toc104639994)

[Hình 2.13 Biểu đồ hoạt động chức năng xóa khóa học 20](#_Toc104639995)

[Hình 2.14 Biểu đồ tuần tự chức năng đăng nhập 21](#_Toc104639996)

[Hình 2.15 Biểu đồ tuần tự chức năng đăng kí thành viên 22](#_Toc104639997)

[Hình 2.16 Dùng để lưu thông tin người dùng 23](#_Toc104639998)

[Hình 2.17 Dùng để lưu các video học trong bài học 23](#_Toc104639999)

[Hình 2.18 Dùng để lưu thông tin các khóa học 24](#_Toc104640000)

[Hình 2.19 Dùng để lưu các bài học của khóa học 24](#_Toc104640001)

[Hình 2.20 Dùng để lưu thông tin các blog 25](#_Toc104640002)

[Hình 2.21 Dùng để lưu thông tin các bình luận 25](#_Toc104640003)

# BẢNG PHÂN CÔNG CÔNG VIỆC

|  |  |
| --- | --- |
| **Họ và tên** | **Nhiệm vụ** |
| **Nguyễn Dương Quý** | - Viết API cho comment  - Triển khai ra ứng dụng chỉnh sửa thông tin người dùng, đổi mật khẩu, comment.  - Thực hiện chức năng tìm kiếm  - Thực hiện chức năng lọc blog, course bằng tag |
| **Lê Hữu Phước** | - Viết API cho Users  - Triển khai ra ứng dụng phần khóa học và bài học  - Reponsive website  - Viết báo cáo |
| **Nguyễn Phú Quý** | - Xây dựng cơ sở dữ liệu  - Viết API các phần khóa học và các bài học  - Triển khai ra ứng dụng phần blog |
| **Nguyễn Khắc Tài** | - Thiết kế cơ sở dữ liệu  - Kiểm thử API  - Thiết logo  - Triển khai ra ứng dụng phần learning |
| **Nguyễn Hữu Minh Quân** | - Tạo model cấu trúc folder  - Kiểm thử API  - Làm slide  - Viết API và triển khai ra ứng dụng phần reaction blog |
| **Nguyễn Hữu Phước** | - Viết API cho khóa học  - Triển khai ra ứng dụng phần xem thông tin người dùng  - Reponsive website |
| **Lê Văn Tao** | - Viết API cho các blog  - Triển khai ra ứng dụng phần đăng kí, đăng nhập  - Reponsive website |

**BẢNG TIẾN ĐỘ DỰ ÁN**

|  |  |
| --- | --- |
| **Dealine** | **Công Việc** |
| Từ 14/03/2022 đến 20/03/2022. | - Thiết kế cơ sở dữ liệu  - Xây dựng cơ sở dữ liệu  - Tạo model |
| Từ 21/03/2022 đến 07/04/2022. | - Tìm hiểu về các công nghệ cần sử dụng như Vue, NodeJS, MongBD…  - Viết API cho các chức năng của website   * API cho user * API cho course * API cho blog * API cho comment * API course part * API cho lesson |
| Từ 08/04/2022 đến 10/04/2022. | -Kiểm thử các API, chỉnh sửa và hoàn thiện API |
| Từ 11/04/2022 đến 14/04/2022 | - Thiết kế logo  - Design giao diện cho website |
| Từ 15/04/2022 đến 10/05/2022 | -Triển khai ứng dụng   * Phần user (Đăng nhập, đăng ký, chỉnh sửa thông tin người dùng) * Phần Course (Thêm, sửa, xóa course, course part, lesson) * Phần Blog (Thêm, sửa, xóa blog, reactions) * Phần Comment (Thêm, sửa, xóa comment) * 1 số chức năng như lọc course, blog, tìm kiếm course, blog… |
| Từ 11/05/2022 đến 15/05/2022 | - Reponsive website |
| Từ 16/05/2022 đến 20/05/2022. | - Kiểm tra logic, kiểm thử lại website, chỉnh sửa các bug phát sinh(nếu có), và hoàn thành ứng dụng. |
| Từ 23/05/2022 đến 28/05/2022 | -Viết báo cáo, làm slide, up source code lên github |

# MỞ ĐẦU

## 1. Lý do đề tài

### 1.1. **Lý do chọn cơ sở dữ liệu (API)**

**Web API** là một phương thức dùng để cho phép các ứng dụng khác nhau có thể giao tiếp, trao đổi dữ liệu qua lại. Dữ liệu được Web API trả lại thường ở dạng [JSON](https://topdev.vn/blog/json-la-gi/) hoặc XML thông qua giao thức HTTP hoặc HTTPS.

API cho phép lấy nội dung từ bất kỳ website hoặc ứng dụng nào một cách dễ dàng nếu được cho phép, tăng trải nghiệm người dùng. API hoạt động như một chiếc cổng, cho phép các công ty chia sẻ thông tin được chọn nhưng vẫn tránh được những yêu cầu không mong muốn.

API có chức năng thay đổi và cập nhật thay đổi theo thời gian thực. Với công nghệ này, dữ liệu sẽ được truyền đi tốt hơn, thông tin chính xác hơn, dịch vụ cung cấp linh hoạt hơn.

Web API được sử dụng hầu hết trên các ứng dụng desktop, ứng dụng mobile và ứng dụng website

Linh hoạt với các định dạng dữ liệu khi trả về client: Json, XML hay định dạng khác.

Giao tiếp hai chiều được xác nhận trong các giao dịch, đảm bảo độ tin cậy cao.

### 1.2 Lý do chọn ứng dụng blog khóa học

Ngày nay với với việc khoa học kĩ thuật ngày càng phát triển mạnh mẽ đặcbiệt là công nghệ thông tin, những gì mà công nghệ thông tin tạo ra giúp con người phát triển trong nhiều lĩnh vực.

Đặc biệt trong lĩnh vực học tập việc áp dụng công nghệ thông tin vào học tập giúp cho nhiều người được học tập thuận tiện và dễ dàng hơn, tiết kiệm thời gian.

## 2. Mục đích của đề tài

Thông tin hiện nay là một lĩnh vực rất quan trọng trong cuộc sống hiện nay, bên cạnh đó các nhu cầu nghe nhìn, giải trí, trao đổi thông tin, học tập,… Vì vậy nhóm em chọn xây dựng web blog các khóa học nhằm mục đích chia sẻ những kiến thức miễn phí đến tất cả mọi người và mọi người đều có thể chia sẻ kiến thức của mình hay tiếp thu những kiến thức của người khác…

## 3**.** Đối tượng và phạm vi của đề tài

- Người dùng

- Mạng internet

- Các thiết bị sử dụng mạng như máy tính, điện thoại,…

## 4. Ứng dụng của đề tài

-Website mà nhóm em xây dựng chưa thực sự hoàn chỉnh nhưng vẫn có thể giúp ích cho người sử dụng trao đổi kiến thức về vấn đề học tập, góp phần tiết kiệm thời gian và thuận tiện hơn cho việc trao đổi kiến thức.

-Đối với người dùng: Người dùng có thể xem và học những khóa học mà chúng em chia sẻ miễn phí và cho phép người dùng chia sẻ, trao đổi những kiến thức của mình hoặc tiếp thu các kiến thức của người khác.

-Đối với chúng em: Nâng cao trình độ phân tích website, và biết thêm nhiều kiến thức về API. Ngoài ra giúp em biết sâu hơn về những ngôn ngữ và phần mềm hỗ trợ trong quá trình làm đồ án.

## 5. Công cụ hỗ trợ

* Sử dụng visual studio code để viết chương trình.
* Dùng hệ quản trị cơ sở dữ liệu MongoDB.
* Ngôn ngữ: Javascript.
* Front-End: Vue JS.
* Back-End: NodeJs.
* Dùng git để làm việc chung với nhau.

# CƠ SỞ LÝ THUYẾT

## Giới thiệu về NodeJS

### 1.1.1 NodeJS là gì?

NodeJS là một mã nguồn được xây dựng dựa trên nền tảng Javascript V8 Engine, nó được sử dụng để xây dựng các ứng dụng web như các trang video clip, các forum và đặc biệt là trang mạng xã hội phạm vi hẹp. NodeJS là một mã nguồn mở được sử dụng rộng bởi hàng ngàn lập trình viên trên toàn thế giới.

NodeJS có thể chạy trên nhiều nền tảng hệ điều hành khác nhau từ WIndow cho tới Linux, OS X nên đó cũng là một lợi thế. NodeJS cung cấp các thư viện phong phú ở dạng Javascript Module khác nhau giúp đơn giản hóa việc lập trình và giảm thời gian ở mức thấp nhất.

Là một nền tảng phát triển ứng dụng mạnh mẽ, Nodejs có thể thỏa mãn mọi nhu cầu lập trình, phát triển ứng dụng. Các chuyên gia trong lĩnh vực lập trình, công nghệ khuyên dùng Nodejs khi phát triển các ứng dụng như Websocket server, Fast File Upload Client, Ad Server, Cloud Services, RESTful API, Any Real-time Data Application,...

### 1.1.2 Lý do sử dụng NodeJS

Nodejs được nhiều lập trình viên, nhà phát triển sử dụng trong thiết kế web hay phát triển ứng dụng bởi một loạt các lý do sau:

- Ứng dụng Nodejs phần đông đều được viết bằng ngôn ngữ lập trình javascript - một ngôn ngữ thông dụng, được sử dụng rộng rãi và chạy được trên nhiều trình duyệt, nền tảng, hệ điều hành,...

- Nodejs khá nhẹ nhưng lại hiệu quả nhờ vào cơ chế non-blocking I/O, chạy đa nền tảng trên Server và dùng Event-driven.

- Tương thích với nhiều thiết bị. Bạn có thể chạy các ứng dụng phát triển bởi Nodejs trên bất cứ thiết bị nào, dù là Mac, Window, Linux,...

- Cộng đồng Nodejs khá lớn và được cung cấp miễn phí cho người dùng.

- Ứng dụng NodeJS có khả năng chạy đa nền tảng, thiết bị, đồng thời đáp ứng được yêu cầu về thời gian thực.

- Node.js có tốc độ cực kỳ nhanh, xử lý được nhu cầu sử dụng của lượng khách truy cập ‘khổng lồ’ trong thời gian cực ngắn.

- Ứng dụng phát triển bởi Node.js có khả năng xử lý nhiều yêu cầu truy cập cùng lúc, ‘cứu’ website của bạn khỏi nguy cơ bị ‘sập’ khi lượng truy cập quá nhiều.

## Giới thiệu về Rest API

### 1.2.1 Rest API là gì?

Rest là từ viết tắt của Representational State Transfer: Nó là một trong những dạng chuyển đổi cấu trúc, với kiểu kiến trúc thường được viết API. Rest thường sử dụng dụng phương thức HTTP đơn giản để có thể tạo ra giao tiếp giữa các máy.Bởi vì thế, thay vì phải sử dụng một URL cho việc xử lý một số thông tin của người dùng thì Rest sẽ yêu cầu HTTP như: GET, POST, DELETE,... đến với bất kỳ một URL để được xử lý dữ liệu.

Rest API là một ứng dụng được dùng để chuyển đổi cấu trúc của dữ liệu có những phương thức giúp kết nối với các thư viện và ứng dụng khác nhau. Rest Api thường không được xem là công nghệ, mà nó là giải pháp giúp tạo ra những ứng dụng web services chuyên dụng để thay thế cho nhiều kiểu khác như: SOAP, WSDL,...

### 1.2.2 Cách thức hoạt động của Rest API

- REST thường hoạt động chủ yếu dựa vào những giao thức HTTP, các cơ sở hoạt động cơ bản nêu trên sẽ được sử dụng những phương thức HTTP riêng biệt:

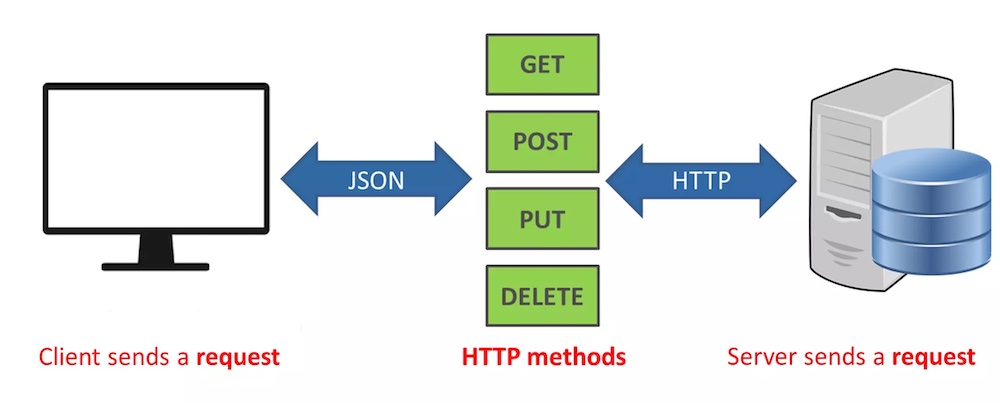
- GET: Trả về với một Resource hoặc có một danh sách Resource.

- POST: Nó hỗ trợ tạo mới một Resource.

- PUT: Thường hỗ trợ cho việc cập nhật các thông tin cho Resource.

- DELETE: Xóa một Resource.

-Đây là phương thức hoạt động được gọi là Crud tương ứng với một Create, Read, Update, Delete ý nghĩa theo thứ tự là: Tạo, đọc, sửa và xóa.Cho đến hiện tại, thì đa số các lập trình viên viết Restful API đều lựa chọn JSON làm format chính thức, tuy nhiên vẫn còn nhiều người lựa chọn XML.



Hình 1.1 Mô hình hoạt động của Rest API

## 1.3 Giới thiệu về VueJS

### 1.3.1 VueJS là gì?

VueJS là một framework Javascript tiên tiến trong xây dựng giao diện người dùng, không giống như các framework khác, Vue được xây dựng từ những dòng code cơ bản nhất nhằm tối ưu tốc độ. Thư viện của Vue chỉ tập trung vào lớp hiển thị, rất đơn giản để tiếp cận và dễ dàng tích hợp vào các hệ thống khác. Vue cũng có khả năng cung cấp các ứng dụng web đơn trang Single Page Application (SPA) cho phép kết hợp với nhiều các công cụ hiện đại**.**

### 1.3.2 Lý do sử dụng VueJS

Hiện nay VueJS đang được sử dụng phổ biến và rộng rãi vì:

- Vue có thể tạo cấu trúc project nhanh chóng hơn nhờ vào command line interface.

- Hiện nay, tài liệu  về Vue ngày càng đa dạng, rõ ràng về ngôn ngữ nên bạn có thể dễ dàng trở thành chuyên gia về nó.

- Vue sở hữu một hệ sinh thái vững chắc nên có thể cung cấp một số add-ons rất hữu ích cho việc xây dựng một ứng dụng fontend điển hình nhất. Nó có thể bao gồm: vue-router, vuex, vue-test-utils, vue-dev-tools, vue-cli,…

- Core Vue sở hữu tính năng tối thiểu bởi khả năng tập trung vào việc render giao diện cho người dùng và các tương tác của nó. Chính vì vậy, nó sẽ cung cấp tối thiểu những tính năng cần thiết cho việc thiết kế và xây dựng kiến trúc. Nó sẽ giúp bạn loại bỏ được các tính năng không cần thiết ra khỏi thư viện trong lõi Vue.js và đảm bảo cho framework nhỏ gọn và mềm dẻo hơn.

## 1.4 Khái niệm MongoDB

**MongoDB** là một database hướng tài liệu (document), một dạng NoSQL database. Vì thế, MongoDB sẽ tránh cấu trúc table-based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. [**MongoDB**](https://topdev.vn/viec-lam-it/mongodb-kt90) sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh.

Dữ liệu lưu trữ phi cấu trúc, không có tính ràng buộc, toàn vẹn nên tính sẵn sàng cao, hiệu suất lớn và dễ dàng mở rộng lưu trữ.

Dữ liệu được caching (ghi đệm) lên RAM, hạn chế truy cập vào ổ cứng nên tốc độ đọc và ghi cao.

## 1.5 Phần mềm Visual studio code

Visual Studio Code là một trình soạn thảo, biên tập code hoàn toàn miễn phí dành được Microsoft phát triển cho các lập trình viên và có mặt trên hầu hết các hệ điều hành phổ biến như: [**Windows**](https://wiki.tino.org/he-dieu-hanh-windows-la-gi/)**,**[**Linux**](https://wiki.tino.org/tim-hieu-ve-cloud-linux/)và macOS. Có thể nói rằng, Visual Studio Code là một sự kết hợp độc đáo – đỉnh cao giữa [**IDE**](https://wiki.tino.org/ide-la-gi/)và Code Editor.

Không chỉ là soạn thảo, chỉnh sửa code, Visual Studio Code còn có thể hỗ trợ bạn làm được rất nhiều việc như: đổi [**theme**](https://wiki.tino.org/theme-la-gi/), hàng loạt phím tắt tiện dụng, có chức năng debug đi kèm, hỗ trợ Git, syntax highlighting hỗ trợ quá trình gõ code, phần gợi ý code thông minh,…

Visual Studio Code là một phần mềm của Microsoft. Vì thế, phần mềm này tương thích cực kỳ mạnh mẽ với hệ điều hành Windows. Ngoài ra, bạn còn có thể tải và sử dụng Visual Studio Code trên 2 hệ điều hành vô cùng phổ biến khác là Linux và MacOS.

Bạn sử dụng nhiều thiết bị với nhiều hệ điều hành khác nhau? Không sao, Visual Studio Code hỗ trợ bạn tối đa có thể trên cả 3 hệ điều hành phổ biến nhất thế giới.

Visual Studio Code được xây dựng dựa trên Electron cùng kết hợp với nhiều công nghệ tiên tiến khác như: JavaScript, Node.js cùng khả năng tăng tốc vô cùng mạnh mẽ của bản thân Visual Studio Code. Người dùng sẽ có được những trải nghiệm tốt nhất trong quá trình phát triển dự án của mình.

Visual Studio Code là một phần mềm được Microsoft rất “cưng chiều” cùng sự nhiệt tình ủng hộ cho sự phát triển của Visual Studio Code từ các cộng đồng lập trình viên lớn. Vì thế, Visual Studio Code liên tục được nâng cấp, được trang bị các công nghệ tối ưu nhất và hỗ trợ tốt nhất cho các lập trình viên như:

- Khả năng mở rộng ngôn ngữ.

- Intellisense.

- Cung cấp kho tiện ích mở rộng khổng lồ.

- Hỗ trợ đa nền tảng.

- Hỗ trợ tối ưu hóa việc code.

- Tối ưu hóa cho việc xây dựng website.

- Khả năng tích hợp đầu cuối.

- Một màn hình với nhiều chức năng.

- Hỗ trợ sử dụng Git.

# PHÂN TÍCH THIẾT KẾ

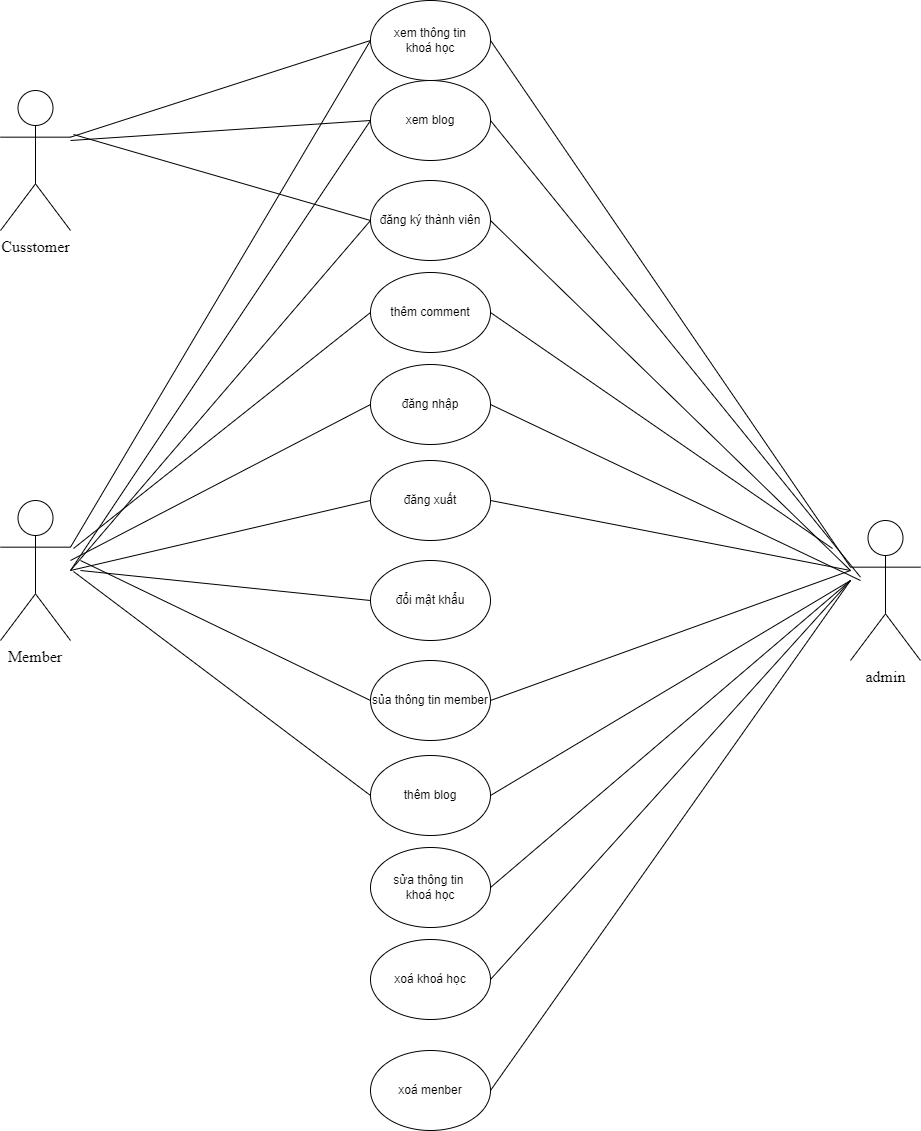
## 2.1. Phân tích các chức năng của hệ thống

Các tác nhân của hệ thống gồm có:



* Administrator: là thành viên quản trị của hệ thống, có các quyền và chức năng như: tạo tài khoản, quản lý khoá học, quản lý member, quản lý blog…
* Member: là hệ thống thành viên có chức năng: Đăng kí, đăng nhập, tìm kiếm, xem, sửa thông tin cá nhân, xem khoá học, thêm blog,comment
* Customer: Là khách vãng lai có chức năng: Đăng kí, tìm kiếm, xem thông tin khoá học

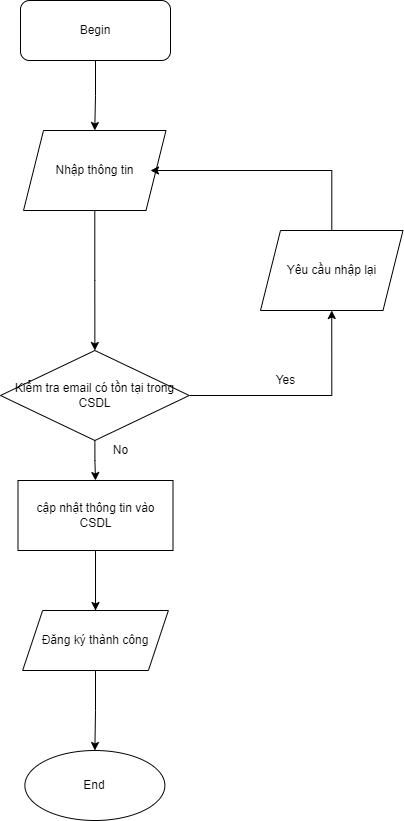
Biểu đồ use-case:



Hình 2.1 Biểu đồ use-case chức năng hệ thống

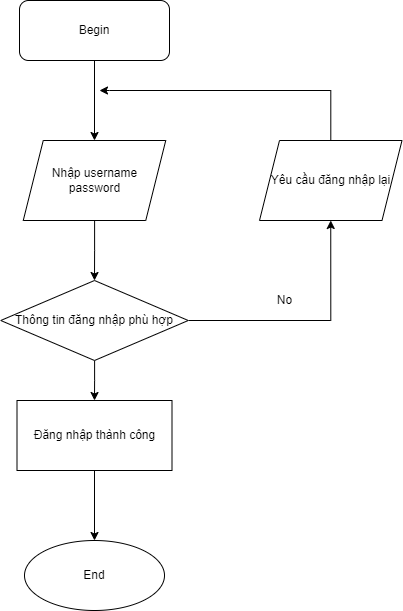
## 2.2. Biểu đồ phân tích thiết kế hệ thống

**Đăng ký thành viên**



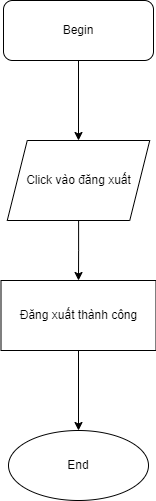
Hình 2.2 Sơ đồ hoạt động đăng kí thành viên

**Đăng nhập:**



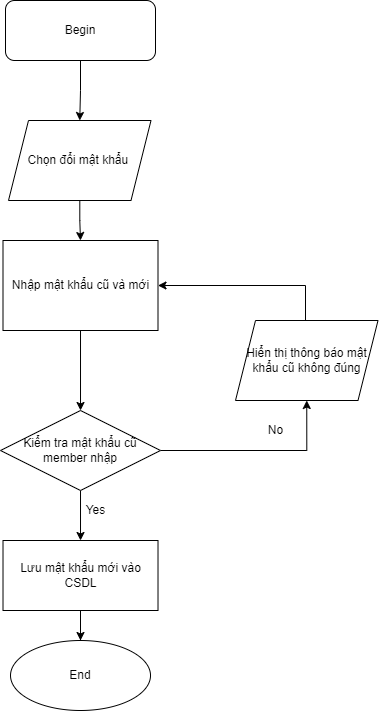
Hình 2.3 Sơ đồ hoạt động đăng nhập

**Đăng xuất:**



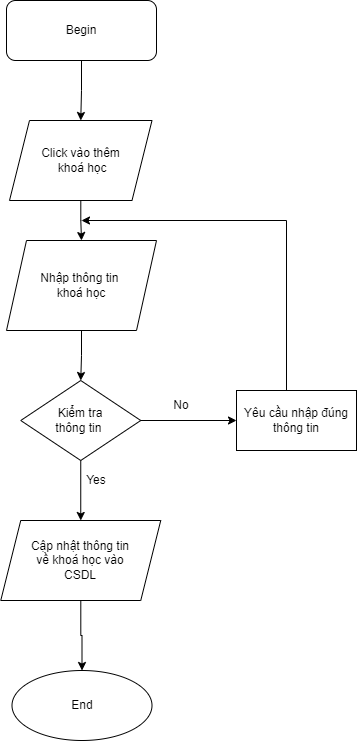
Hình 2.4 Sơ đồ hoạt động chức năng đăng xuất

**Đổi mật khẩu:**

****

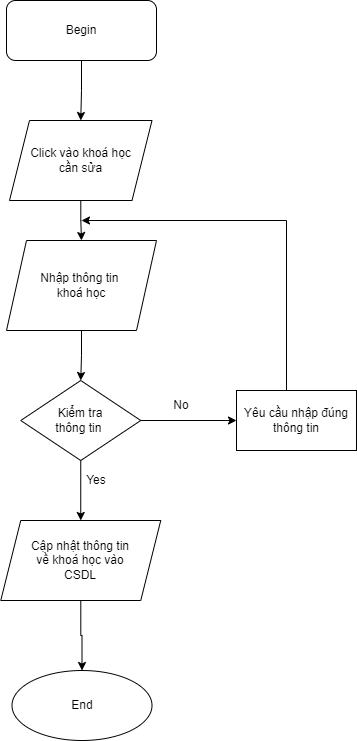
Hình 2.5 Sơ đồ hoạt động đổi mật khẩu

**Thêm khoá học:**



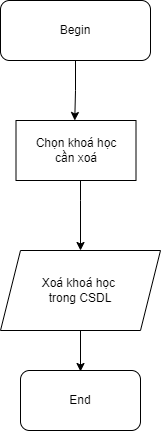
Hình 2.6 Sơ đồ hoạt động thêm khóa học

**Sửa thông tin khoá học**



Hình 2.7 Sơ đồ hoạt động sửa thông tin khóa học

**Xoá khoá học**



Hình 2.8 Sơ đồ hoạt động xóa khóa học

## 2.3. Biểu đồ tuần tự các chức năng

**Chức năng tìm kiếm:**



Hình 2.9 Biểu đồ tuần tự chức năng tìm kiếm

**Chức năng quản lí khóa học**



Hình 2.10 Biểu đồ tuần tự chức năng quản lí khóa học

**Chức năng thêm mới khóa học**



Hình 2.11 Biểu đồ tuần tự chức năng thêm mới khóa học

**Chức năng chỉnh sửa khóa học**



Hình 2.12 Biểu đồ tuần tự chức năng sửa thông tin khóa học

**Chức năng xóa khóa học**



Hình 2.13 Biểu đồ hoạt động chức năng xóa khóa học

**Chức năng đăng nhập**



Hình 2.14 Biểu đồ tuần tự chức năng đăng nhập

**Chức năng đăng kí thành viên**



Hình 2.15 Biểu đồ tuần tự chức năng đăng kí thành viên

## 2.4. Phân tích cơ sở dữ liệu

**Bảng user**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Not null** |
| Name | String | x |
| Username | String | x |
| password | String | x |
| Email | String | x |
| Address | String | x |
| Phone | String | x |
| Gender | String | x |
| Avatar | String |  |
| isAdmin | Boolean | x |
| isActive | Boolean | x |
| Courses | Array |  |
| Blogs | Array |  |

Hình 2.16 Dùng để lưu thông tin người dùng

**Bảng lessons**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Not null** |
| name | String | X |
| coursePart | String | X |
| videoId | String | X |
| isSelect | Boolean | X |

Hình 2.17 Dùng để lưu các video học trong bài học

**Bảng course ( khoá học )**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Not null** |
| Name | String | X |
| Description | String | X |
| imgThumbanil | String | X |
| Author | String | X |
| Slug | String | X |
| outstandingMark | Int | X |
| tags | String |  |
| courseParts | Array |  |

Hình 2.18 Dùng để lưu thông tin các khóa học

**Bảng CouserPart**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Not null** |
| name | String | X |
| courseId | String | X |
| lessons | Array | X |
| **isSelect** | **Boolean** | **X** |

Hình 2.19 Dùng để lưu các bài học của khóa học

**Bảng Blog**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Not null** |
| Title | String | X |
| Content | String | X |
| MetaDesciption | String | X |
| ImageThumbnail | String | X |
| Reactions | String | X |
| Tag | String | X |
| OutstandingMark | Number | X |
| Author | String | X |
| comments | Array |  |
| Slug | String | X |

Hình 2.20 Dùng để lưu thông tin các blog

**Bảng Comment**

|  |  |  |
| --- | --- | --- |
| **Tên thuộc tính** | **Kiểu dữ liệu** | **Not null** |
| postId | String | X |
| courseId | String | X |
| authorId | String | X |
| commentContent | String | X |

Hình 2.21 Dùng để lưu thông tin các bình luận

# KẾT LUẬN

## 1. Bài học kinh nghiệm:

Một thời gian tập trung triển khai đề tài, em đã hoành thành được trang website chia sẻ khóa học, với giao diện đẹp, thao tác sử dụng dễ dàng, than thiện với người dùng.

Và em có thêm những kỹ những kỹ năng nền tảng để xây dựng một website bằng Nodejs, Vuejs, mongoDB … đã học vào việc phát triển website. Bên cạnh đó, em cũng được mở rộng thêm kiến thức chuyên ngành nhờ việc tìm hiểu kiến thức lập trình để áp dụng và phát triển ứng dụng này. Nhờ vậy, kỹ năng tự học và vận dụng các kỹ năng mới của em được nâng cao hơn.

## 2. Hướng nghiên cứu phát triển:

* Tìm hiểu sâu hơn về ngôn ngữ để có thể đáp ứng nhiều hơn nữa nhu cầu của người sử dụng, phát triển và tối ưu hóa hệ thống,
* Tìm hiểu thêm một số ngôn ngữ, các phần mềm ứng dụng để nâng cao giao diện đồ họa đẹp mắt, thân thiện hơn…
* Xây dựng trang Web quy mô lớn hơn với nhiều ứng dụng, chức năng...

# TÀI LIỆU THAM KHẢO

1. <https://www.nodebeginner.org/index-vi.html>
2. <https://viblo.asia/p/mot-so-kien-thuc-co-co-ban-ve-vuejs-yMnKMjpgZ7P>
3. <https://janeto.gitbook.io/mongodb-toan-tap/>
4. <https://vuejs.org/guide/introduction.html>
5. <https://mongoosejs.com/docs/guide.html>
6. <https://expressjs.com/>