TRƯỜNG ĐẠI HỌC THUỶ LỢI

**KHOA CÔNG NGHỆ THÔNG TIN**

**---------🙞🙜🕮🙞🙜---------**



ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC

**XÂY DỰNG WEBSITE ĐÁNH GIÁ KHOÁ HỌC DIGITAL PAINTING**

Sinh viên thực hiện: Nguyễn Trần Tuấn Đạt

Giảng viên hướng dẫn: PGS.TS Lê Đức Hậu

HÀ NỘI, NĂM 2023

# MỤC LỤC

## DANH MỤC CÁC TỪ VIẾT TẮT VÀ GIẢI THÍCH CÁC THUẬT NGỮ

**Chương I. Tổng quan đồ án tốt nghiệp**

* 1. **Giới thiệu đồ án**

## Đặt vấn đề

Trong những năm gần đây, sự phát triển nhanh chóng của công nghệ đang ảnh hưởng tới mọi mặt của đời sống xã hội. Trong đó các ngành như hội họa, mĩ thuật, thiết kế đang trải qua những thay đổi lớn do sự tiến triển nhanh chóng của công nghệ và sự phổ cập của các phương tiện truyền thông xã hội. Chuyển từ việc vẽ truyền thống sang vẽ kĩ thuật số là xu hướng mới mang lại nhiều lợi ích và đang ngày càng được ưu chuộng trong cộng đồng bởi sự tiện ích, linh hoạt, tích kiệm và dễ dàng tiếp cận. Thông qua sự kết hợp giữa nghệ thuật và công nghệ, ngành hội họa, mỹ thuật, và thiết kế đang trở thành một lĩnh vực sáng tạo đa dạng và phát triển mạnh mẽ. Sự đổi mới trong công nghệ mở ra không gian cho sự sáng tạo không giới hạn và mang lại nhiều cơ hội cho những người đam mê ngành nghề này.

Năm 2019, bùng phát đại dịch COVID-19 để kiểm soát đại dịch nhiều quốc gia đã áp đặt các biện pháp giãn cách xã hội và hạn chế di chuyển, điều này đã làm giảm số lượng công việc mà người ta có thể làm ngoài nhà hoặc buộc nhiều người phải làm việc từ xa (Remote Work). Với sự giãn cách xã hội và hạn chế di chuyển, nhiều doanh nghiệp đã chuyển từ mô hình làm việc truyền thống sang mô hình làm việc từ xa. Điều này đã tạo ra nhu cầu lớn cho các kỹ năng kỹ thuật số, trong đó Digital Painting đóng một vai trò quan trọng. Nhìn nhận điều đó Digital Painting đã trở thành một phần quan trọng trong cuộc sống và làm việc hàng ngày của nhiều nghệ sĩ và nhà thiết kế.

Nắm bắt điều đó cùng với sự phát triển hiện nay mục tiêu của đồ án này tạo ra là cung cấp thông tin cho những người đang muốn học các khoá học Digital Painting 1 cách dễ dàng. Các thông tin sẽ được tổng hợp và đánh giá qua sự chia sẻ cộng đồng Artist ở một số nền tảng mạng xã hội như Facebook, Instagram, ArtStation, DeviantArt. việc cung cấp thông tin chi tiết và hấp dẫn về Digital Painting không chỉ giúp người học dễ dàng tiếp cận và phát triển kỹ năng, mà còn tạo ra một không gian thú vị để khám phá và đam mê nghệ thuật số.

## Mô tả dự án

Hệ thống tích hợp các công cụ tìm kiếm mạnh mẽ, hỗ trợ người dùng tìm kiếm cho phép cập nhật thông tin , truy vấn tìm kiếm thông tin khoá học thuận tiện ,nhanh chóng và chính xác.

Xây dựng website tìm kiếm khoá học Digital Painting sẽ giúp nâng cao chất lượng quản lý có hệ thống , tăng năng suất lao động con người ,tìm kiếm và lưu trữ thông tin một cách dễ dàng và chính xác.Dự án với thiết kế đơn giản nhưng bắt mắt, thân thiện, dễ dàng sử dụng sẽ tối ưu hóa trải nghiệm giúp thu hút một lượng lớn người dùng.

Website với việc ưu tiên các bảo mật an toàn thông tin khóa học cũng như tính chân thật của thông tin từ cộng đồng người học. Nhằm đảm bảo bản quyền và tuân thủ pháp luật với sự tôn trọng chất xám và công sức của người dạy tự mở hoặc các trung tâm đào tạo thì người dùng sẽ đến trang chính thống của khóa học digital painting theo thông tin mà hệ thông cung cấp với các đánh giá của cộng đồng để đăng ký khóa học theo nhu cầu và mong muốn.

## Mục tiêu của đồ án

Tạo ra một hệ thống trực tuyến giúp cấp nhật thông tin cho người dùng đang muốn học các khoá học Digital Painting một cách dễ dàng với các đánh giá dựa trên thông tin cung cấp về khoá học bao gồm mục tiêu, nội dung và lợi ích mà họ có thể nhận được từ cộng đồng người học phù hợp với nhu cầu và mong muốn của người dùng.

## Phạm vi đề tài

Báo cáo sẽ tập chung vào trình bày các kết quả với các nội dung sau: Laravel framework, Laragon, Bootstrap framework và Model MVC. Từng phần sẽ được giới thiệu sơ lược và trình bày với những nội dung cơ bản nhất và lợi ích, điểm mạnh cho nhà phát triển phần mềm.

Phạm vi chung: Các loại khóa học hệ thống tập chung vào các khóa học Digital Painting bao gồm các chủ đề kỹ thuật vẽ, màu sắc, kỹ năng tay và nghệ thuật số. Người dùng có khả năng tìm kiếm các khóa học theo nhiều tiêu chí khác nhau và có thể đánh giá, bình luận về các khóa học theo cộng đồng người học. Hệ thống còn hỗ trợ cả người dùng đăng ký và người dùng không đăng ký. Tuy nhiên, chức năng đánh giá và bình luận có thể yêu cầu đăng nhập.

* 1. **Cơ sở lý thuyết**

## Nguyên lý xây dựng hệ thống tìm kiếm

* + - 1. **Quản trị cơ sở dữ liệu(DBMS)**
         1. **Tổng quan**

Hệ quản trị cơ sở dữ liệu (DBMS) là một phần mềm quản lý và tổ chức cơ sở dữ liệu, giúp lưu trữ và truy xuất thông tin một cách hiệu quả. Nó cung cấp môi trường để định nghĩa, thay đổi và truy xuất dữ liệu từ cơ sở dữ liệu. DBMS giúp bảo vệ dữ liệu, duy trì tính nhất quán và đồng bộ trong hệ thống.

DBMS có nhiệm vụ quản lý dữ liệu, cho phép người dùng truy cập, khóa, sửa đổi dữ liệu và xác định cấu trúc logic của cơ sở dữ liệu. Ba yếu tố then chốt này đã cung cấp các thủ tục quản trị dữ liệu thống nhất đảm bảo mọi dữ liệu được bảo mật và toàn vẹn. Bên cạnh đó DBMS còn còn có chức năng thay đổi quyền quản lý, giám sát, điều chỉnh hiệu suất, bảo mật, sao lưu và phục hồi dữ liệu.

Đặc biệt một trong những chức năng hữu ích mà DBMS mang đến cho người dùng là cho phép nhiều người truy cập vào cơ sở dữ liệu ở các vị trí khác nhau bằng một phương thức được kiểm soát. Ngoài ra DBMS còn giới hạn những vùng dữ liệu mà người dùng có thể xem đồng thời nó cũng giới hạn cách mà họ tương tác với dữ liệu (thêm, xóa, chỉnh sửa…).

* + - * 1. **Lịch sử phát triển**

Vào những năm 60 thời kì đầu, CSDL xuất hiện vào những năm 60 khi máy tính trở nên phổ biến. Hệ thống CSDL lớn đầu tiên được phát triển bởi IBM để hỗ trợ dự án Apollo của NASA, với tên gọi IMS (Information Management System) và sử dụng mô hình dữ liệu phân cấp. Charles Bachman phát triển Integrated Data Store (IDS) sử dụng mô hình dạng mạng, mô hình sau đó được tiêu chuẩn hóa bởi CODASYL.

Năm 1970, Edgar Codd (IBM) đề xuất mô hình quan hệ (relational database), một bước quan trọng trong phát triển CSDL. Ông cũng giới thiệu các khái niệm như các quan hệ, khóa chính, và ngôn ngữ truy vấn. IBM phát triển dự án System R để nghiên cứu mô hình quan hệ và tạo ra ngôn ngữ SEQUEL (tiền thân của SQL).

Năm 1976 Peter Chen hoàn thiện mô hình quan hệ bằng cách giới thiệu mô hình thực thể - liên kết (Entity-Relationship hay E-R).

Năm 1979 bước đầu của CSDL quan hệ, Relational Software Inc (sau đổi tên là Oracle) là công ty đầu tiên sử dụng mô hình quan hệ để thương mại hóa và chiếm lĩnh thị trường. IBM cũng đã bắt đầu thương mại hóa SQL, ngôn ngữ truy vấn được tạo ra để làm việc với CSDL quan hệ.

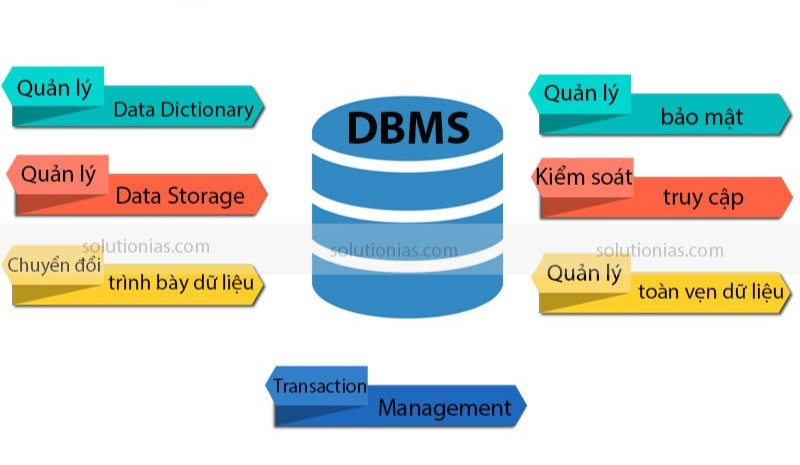
Từ thập niên 1980, hệ CSDL hướng đối tượng (Object-Oriented DBMS) được nghiên cứu và trong những năm 1990, chúng bắt đầu xuất hiện trên thị trường. Cùng với sự xuất hiện các loại dữ liệu đa dạng, khác biệt như hình ảnh, bản vẽ từ phần mềm CAD, dẫn đến sự phát triển của HQTCSDL hướng đối tượng.

Bắt đầu thế kỷ 21 Các CSDL mã nguồn mở như MySQL, PostgreSQL, SQLite trở nên phổ biến, cùng với sự xuất hiện của các CSDL NoSQL như Cassandra, Redis. CSDL NoSQL giúp đáp ứng nhu cầu xử lý dữ liệu lớn và linh hoạt trong môi trường web đương đại.

* + - * 1. **Một số thành phần chính**

Hệ thống quản trị cở sở dữ liệu DBMS là một phần mềm của hệ thống bao gồm nhiều thành phần tích hợp cung cấp một môi trường nhất quán, được quản lý để tạo, truy cập và sửa đổi dữ liệu trong cơ sở dữ liệu. Thành phần bao gồm:

* Công cụ lưu trữ: Được sử dụng để lưu trữ dữ liệu, DBMS phải giảo tiếp với một hệ thống tệp ở cấp hệ điều hành để lưu trữ nó. Nó có thể sử dụng các thành phần bổ sung để lưu trữ dữ liệu hoặc giao diện với dữ liệu thực tế ở cấp hệ thống tệp.
* Danh mục siêu dữ liệu: Được hoạt động như một kho lưu trữ cho tất cả các đối tượng cơ sở dữ liệu đã được tạo. Khi cơ sở dữ liệu và các đối tượng khác được tạo, DBMS sẽ tự động đăng ký thông tin về chúng trong danh mục siêu dữ liệu. DBMS sử dụng danh mục này để xác minh yêu cầu dữ liệu của người dùng và người dùng có thể truy vấn danh mục để biết thông tin về các đối tượng cơ sở dữ liệu, lược đồ, chương trình, bảo mật, hiệu suất, giao tiếp và các chi tiết môi trường khác về cơ sở dữ liệu mà nó quản lý.
* Ngôn ngữ truy cập cơ sở dữ liệu: DBMS phải cung cấp API để truy cập dữ liệu, thường ở dạng ngôn ngữ truy cập cơ sở dữ liệu để truy cập và sửa đổi dữ liệu nhưng cũng có thể được sử dụng đẻ tạo các đối tượng cơ sở dữ liệu và bảo mật cũng như cho phép truy cập dữ liệu.
* Công cụ tối ưu hóa: DBMS cung cấp một công cụ tối ưu hóa, được sử dụng để phân tích cú pháp các yêu cầu ngôn ngữ truy cập cơ sở dữ liệu và biến chùng thành các lệnh khả thi để truy cập và sửa đổi dữ liệu.
* Bộ xử lý truy vấn: Sau khi truy vấn được tối ưu hóa, DBMS phải cung cấp phương tiện để chạy truy vấn và trả về kết quả.
* Trình quản lý khóa: DBMS quản lý truy cập đồng thời vào cùng một dữ liệu. Khóa được yêu cầu để đảm bảo nhiều người dùng không cố gắng sửa đổi cùng một dữ liệu.
* Trình quản lý nhật ký: DBMS ghi lại tất cả các thay đổi được thực hiện đối với dữ liệu do DBMS quản lý. Bản ghi các thay đổi được gọi là nhật ký và thành phần quản lý nhật ký của DBMS được sử dụng để đảm bảo rằng các bản ghi nhật ký được thực hiện hiệu quả và chính xác.
* Tiện ích dữ liệu: DBMS cung cấp một tập hợp các tiện ích để quan lý và kiểm soát các hoạt động của cơ sở dữ liệu.

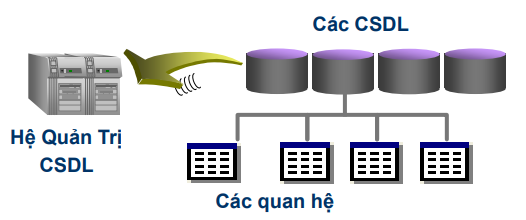


Hình 1: Các thành phần của DBMS

* + - * 1. **Áp dụng vào hệ thống**

Sử dụng quản trị cơ sở dữ liệu để quản lý thông tin về khóa học, đảm bảo cấu trúc dữ liệu hợp lý và tối ưu hóa truy vấn dữ liệu. DBMS đóng vai trò quan trọng trong việc:

* Lưu trữ thông tin khóa học: Định cấu trúc dữ liệu để lưu trữ thông tin chi tiết về các khóa học Digital Painting bao gồm mô tả, chủ đề, tác giả và đánh giá.
* Tối ưu hóa truy vấn dữ liệu: Sử dụng các index và chỉ mục để tối ưu hóa truy vấn, giảm thời gian đáp ứng khi người dùng tìm kiếm thông tin về khóa học.
* Quản lý đánh giá và bình luận: Lưu trữ thông tin đánh giá và bình luận của người dùng một cách an toàn và có thể truy xuất dễ dàng.
* Bảo mật dữ liệu: Áp dụng các biện pháp bảo mật để đảm bảo an toàn và riêng tư của thông tin trong cơ sở dữ liệu.
* Tương tác với ORM: Sử dụng Eloquent ORM trong Laravel để thực hiện các thao tác CRUD một cách thuận tiện, giúp tương tác với cơ sở dữ liệu trở nên đơn giản và linh hoạt.



Hình 2. Các lược đồ quan hệ và các mối liên kết giữa chúng trong HT quản lý

Một trong những lợi thế khi sử dụng DBMS là nó cho phép người dùng và người lập trình ứng dụng truy cập và sử dụng đồng thời cùng một dữ liệu trong khi quản lý tính toàn vẹn của dữ liệu. DBMS cũng cung cấp một kho dữ liệu mà nhiều người dùng có thể truy cập một cách có kiểm soát.

* + - 1. **Hệ thống tìm kiếm**
         1. **Tổng quan lý thuyết.**

Ngôn ngữ lập trình là công cụ để viết mã máy và tương tác với máy tính.

Framework là một bộ công cụ, thư viện, và quy tắc mà theo đó, các nhà phát triển có thể xây dựng và phát triển ứng dụng một cách nhanh chóng và hiệu quả. Nó cung cấp cấu trúc và hỗ trợ chuẩn hóa cho quá trình phát triển, giúp giảm thiểu công sức và thời gian cần thiết.

Công nghệ tìm kiếm là tập hợp các thuật toán và kỹ thuật được áp dụng để cải thiện quá trình truy xuất thông tin từ một kho lưu trữ dữ liệu, nhằm tối ưu hóa khả năng tìm kiếm, chính xác, và hiệu suất.

* + - * 1. **Áp dụng vào hệ thống**

Lựa chọn ngôn ngữ lập trình và framework phù hợp mang lại cho ta sự dễ dàng khi áp dụng hệ thống và nó có tính chất quan trọng:

* Hiệu suất: Sự kết hợp giữa ngôn ngữ lập trình và framework cần đảm bảo hiệu suất ứng dụng, đặc biệt là trong việc xử lý truy vấn đến cơ sở dữ liệu và tương tác người dùng.
* Tính linh hoạt: Ngôn ngữ và framework cần cung cấp tính linh hoạt cho lập trình viên, cho phép họ tùy chỉnh và mở rộng chức năng theo yêu cầu cụ thể của dự án.
* Cộng đồng hỗ trợ: Cộng đồng lớn và tích cực hỗ trợ ngôn ngữ và framework giúp giải quyết vấn đề nhanh chóng và cung cấp nguồn tư liệu đa dạng.
* Bảo trì: Mã nguồn được xây dựng trên framework và ngôn ngữ có tính nhất quán, giúp dễ bảo trì và phát triển trong tương lai.

Nhằm tìm kiếm được thuận lợi, dễ dàng thì sử dụng công nghệ tìm kiếm để cải thiện khả năng tìm kiếm trên trang web, với các tính năng như tìm kiếm nâng cao và gợi ý từ khóa. Một vài công nghệ ta có thể sử dụng như Eloquent ORM, Laravel Scout, Algolia, ElasticSearch hoặc tìm kiếm Full-Text trong MySQL.

* + 1. **Model MVC**
       1. **Tổng quan**

Mô hình Model-View-Controller (MVC) là một mẫu kiến trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View và Controller. Do đó viết tắt MVC. Mỗi thành phần kiến trúc được xây dựng để xử lý khía cạnh phát triển cụ thể của một ứng dụng. MVC tách lớp logic nghiệp vụ và lớp hiển thị ra riêng biệt.

MVC cũng được sử dụng rộng rãi trong phát triển **web**, sự khác biệt được tùy chỉnh liên quan đến sự có mặt của **server - client**.

Việc sử dụng mô hình MVC sẽ giúp tổ chức ứng dụng một cách rõ ràng và dễ dàng bảo trì, cũng như tạo ra một quy trình phát triển hiệu quả và linh hoạt.

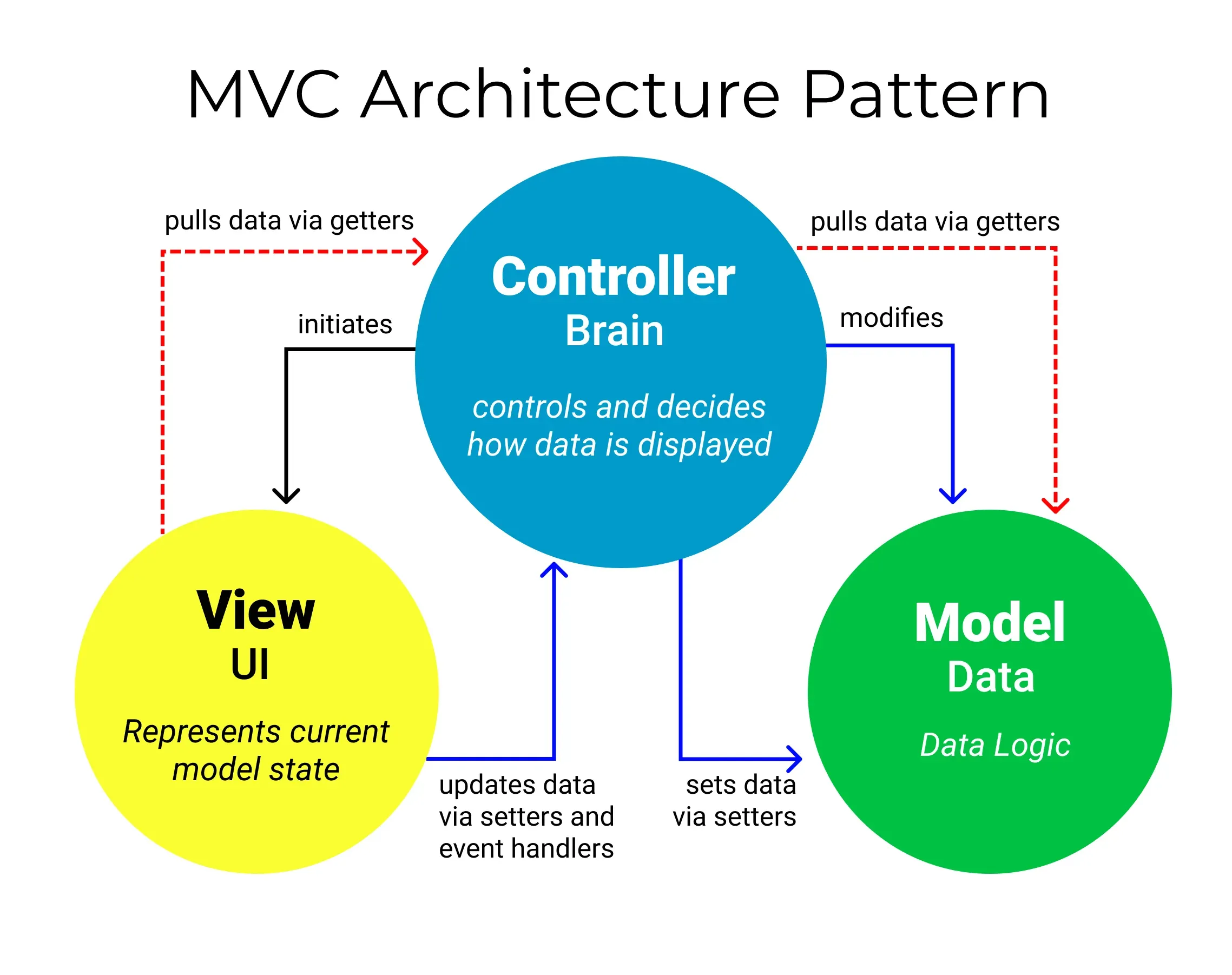
* + - 1. **Lịch sử phát triển**

MVC được tiến sĩ Trygve Reenskaug đưa vào ngôn ngữ lập trình Smalltalk-76 khi ông đến trung tâm Nghiên cứu Xerox Palo Alto (PARC) vào giữa năm 1970. Sau đó, việc triển khai trở nên phổ biến trong các phiên bản khác của Small- Talk. Năm 1988, các bài báo “The Journal of Object Technology” – JOT mang lại bước tranh toàn cảnh về MVC mang liệu sự hiệu quả tốt nhất.

* + - 1. **Các thành phần chính**

Mô hình MVC là một mô hình kiến trúc phần mềm được sử dụng để tổ chức mã nguồn của ứng dụng. Mô hình này tách biệt ứng dụng thành ba thành phần chính: Model (Mô hình), View (Giao diện), và Controller (Bộ điều khiển).

* Model(Mô hình): Chứa dữ liệu và logic xử lý dữ liệu của ứng dụng. Nhiệm vụ quản lý trạng thái ứng dụng và dữ liệu, thực hiện các thao tác đối với cơ sở dữ liệu, cung cấp các phương thức để đọc và cập nhật dữ liệu.
* View(Giao diện): Đại diện cho phần giao diện người dùng, hiển thị thông tin từ Model và tương tác với người dùng. Nhiệm vụ là hiển thị dữ liệu từ Model theo cách được định nghĩa, nhận thông báo từ Controller về các sự kiện người dùng, không chứa logic xử lý dữ liệu.
* Controller (Bộ điều khiển): Quản lý sự kiện và tương tác giữa Model và View, là "não bộ" của hệ thống. Nhiệm vụ tiếp nhận yêu cầu từ người dùng thông qua View, xử lý logic kinh doanh và tương tác với Model, cập nhật dữ liệu Model và thông báo cho View về sự thay đổi.



Hình 3. Thành phần của model MVC.

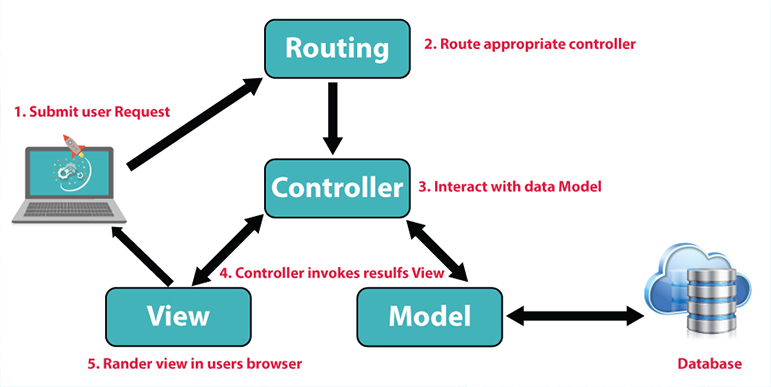
Ưu điểm mô hình: Tách biệt rõ ràng giữa logic xử lý dữ liệu, giao diện và quản lý sự kiện, mã nguồn được tổ chức thành các thành phần riêng lẻ, dễ bảo trì và mở rộng, cho phép nhiều nhóm phát triển làm việc đồng thời trên các thành phần khác nhau.

Với sự độc lập và phát triển song song thì mỗi nhà phát triển phần mềm có thể đảm nhiệm một thành phần và không ảnh hưởng đến nhau khiến quá trình phát triển diễn ra nhanh chóng và dễ dàng. Cùng với các kỹ thuật bất đồng bộ khiến các ứng dụng được load nhanh hơn đơn giản vì tiến hành chạy nhiều câu lệnh cùng lúc.

* + - 1. **Luồng xử lý**

MVC được thiết kế xử lý các luồng với các request và sẽ response để hiển thị view, nó xử lý như sau:

* Khi một yêu cầu của từ máy khách (Client) gửi đến Server. Thì bị Controller trong MVC chặn lại để xem đó là URL request hay sự kiện.
* Sau đó, **Controller** xử lý **input** của user rồi giao tiếp với**Model** trong MVC.
* Model chuẩn bị data và gửi lại cho Controller.
* Cuối cùng, khi xử lý xong yêu cầu thì Controller gửi dữ liệu trở lại View và hiển thị cho người dùng trên trình duyệt.



Hình 4: luồng xử lý của Model MVC

* 1. **Công nghệ sử dụng**

## Tổng quan về Framework Laravel

* + - 1. **Khái niệm**

Laravel là một framework phát triển ứng dụng web mạnh mẽ được viết bằng PHP. Nó dễ dàng tiếp cận và linh hoạt sử dụng cung cấp các công cụ và thư viện hữu ích giúp tạo nên các web có độ phức tạp cao. Điều mà tạo nên framework mạnh mẽ như vậy gắn liền với một số điểm quan trọng sau.

* Mô hình MVC(Model - View - Controller): Laravel tuân thủ mô hình thiết kế kiến trúc MVC gồm tách biệt xử lý dữ liệu (Model), giao diện người dùng (View) và điều khiển luồng ứng dụng (Controller)
* Composer: Một trình quản lý gói phụ thuộc thuộc PHP, để quản lý các thư viện và gói mở rộng của ứng dụng
* Routing: Laravel cung cấp một hệ thống định tuyến mạnh mẽ, cho phép bạn xác định các tuyến đường (routes) để điều hướng yếu cầu HTTP đến các phương thức xử lý tương ứng
* Blade Template Engine: Là công cụ động lực hoá việc tạo ra giao diện người dùng, blade cung cấp cú pháp dễ đọc và linh hoạt
* Atisan: Công cụ hỗ trợ các dòng lệnh được cung cấp bởi Laravel thực hiện các tác vụ như tạo mới các file, quản lý cơ sở dữ liệu và thực hiện các tác vụ tự động hoá.
* Eloquent ORM: Đây là một ORM (Object-Relational Mapping) mạnh mẽ, cho phép bạn tương tác với cơ sở dữ liệu bằng cách sử dụng các đối tượng PHP thay vì truy vấn SQL trực tiếp
* Các tính năng khác: Laravel cung cấp nhiều tính năng hữu ích khác như quản lý sự kiện, gửi email, gọi API và nhiều hơn nữa.



Hình 5: Các ưu điểm của Laravel framework

* + - 1. **Lịch sử phát triển**

Laravel được tạo ra bởi Taylor Otwell bởi sự nỗ lực với mục đích cung cấp một giải pháp thay thế nâng cao cho Codelgniter framework vì nó không cung cấp một số tính năng nhất định như là hỗ trợ tích hợp xác thực (Authentication) và ủy quyền (Authorization) người dùng.

**Bản phát hành beta đầu tiên của Laravel** được phát hành vào **ngày 9/6/2011**, tiếp theo là bản phát hành **Laravel 1** sau đó trong cùng một tháng. Laravel 1 bao gồm hỗ trợ tích hợp cho xác thực, bản địa hóa (localisation), models, views, sessions, định tuyến (routing) và các cơ chế khác, nhưng thiếu hỗ trợ cho các bộ điều khiển (controller) ngăn cản nó trở thành một MVC framework thực sự.

Vào tháng 9/2011, **Laravel 2** được phát hành mang lại nhiều cải tiến khác nhau từ tác giả và cộng đồng. Với tính năng chính mới bao gồm sự hỗ trợ cho controller đã làm cho Laravel 2 trở thành một framework tuân thủ đầy đủ MVC, hỗ trợ sẵn cho nguyên lý điều khiển (IoC) và một hệ thống templating gọi là Blade. Nhược điểm, hỗ trợ cho các gói của bên thứ ba (third-party packages) đã được gỡ bỏ trong Laravel 2.

Tháng 2/2012, phát hành **Laravel 3** với một loạt các tính năng mới bao gồm giao diện dòng lệnh - command-line interface (CLI) có tên **Artisan**, hỗ trợ sẵn cho nhiều hệ thống quản lý cơ sở dữ liệu, database migrations, hỗ trợ xử lý events và hệ thống đóng gói được gọi là Bundles. Kể từ bước nhảy vọt ở phiên bản này, ta chứng kiến sự gia tăng người dùng (developer) và mức độ phổ biến của Laravel tăng chóng mặt.

Tháng 5/2013, **Laravel 4** đặt tên **Illuminate** phát hành được viết thành một khung hoàn chỉnh của Laravel framework, di chuyển bố cục của nó thành một tập các gói riêng biệt được phân phối thông qua [Composer](https://chungnguyen.xyz/go/Tl5p32), phục vụ như một trình quản lý gói ứng dụng. Cách bố trí như vậy đã cải thiện khả năng mở rộng của Laravel 4. Các tính năng mới khác trong bản phát hành Laravel 4 bao gồm database seeding, hỗ trợ hàng đợi (queue) tin nhắn, hỗ trợ cho việc gửi các loại email khác nhau và hỗ trợ xóa các bản ghi cơ sở dữ liệu dạng SoftDelete (không xóa hẳn khỏi database).

Tháng 2/2015, phát hành **Laravel 5** với các tính năng mới trong bản phát hành Laravel 5 bao gồm hỗ trợ lập kế hoạch (scheduling) các tác vụ được thực hiện định kỳ thông qua gói có tên **Scheduler**, một lớp trừu tượng (abstraction layer) có tên **Flysystem** cho phép lưu trữ từ xa cách dùng tương tự như trên local, cải thiện việc xử lý tài sản gói thông qua **Elixir** (giờ đổi tên thành [Laravel Mix](https://chungnguyen.xyz/go/32dmZZ)), hỗ trợ xác thực bên thứ 3 đơn giản thông qua gói Socialite (xem thêm [Laravel đăng nhập thông qua các mạng xã hội Facebook, Google, Twitter, Github](https://chungnguyen.xyz/go/sDukJw)). Laravel 5 cũng giới thiệu cấu trúc cây thư mục mới cho các ứng dụng đã phát triển.

Tháng 6/2015, **Laravel 5.1** phát hành là bản phát hành đầu tiên của Laravel để nhận hỗ trợ dài hạn (LTS), với các bản sửa lỗi có sẵn trong hai năm và các bản vá bảo mật trong ba năm.

Ngày 23/8/2016, phát hành **Laravel 5.3** tập chung vào việc cải thiện tốc độ phát triển bằng cách bổ sung thêm các cải tiến cho các tác vụ phổ biến.

Tháng 1/2017, phát hành **Laravel 5.4** cho ra nhiều tính năng mới như Laravel Dusk, Laravel Mix, Blade Components và Slots, Markdown Emails, Automatic Facades, Route Improvements, Higher Order Messaging cho Collections, và nhiều thứ khác.

Tháng 8/2017, phát hành **Laravel 5.5** là phiên bản LTS thứ 2 và được cho là phiên bản LTS Release tiếp theo.

Tháng 2/2018, phát hành **Laravel 5.6** cải thiện và thêm mới một số tính năng như Laravel Nova, job Middleware, Eloquent Date Casting, Resource Response Headers và cung cấp một bộ dụng cụ có sẵn thiết lập với Boostrap 4.

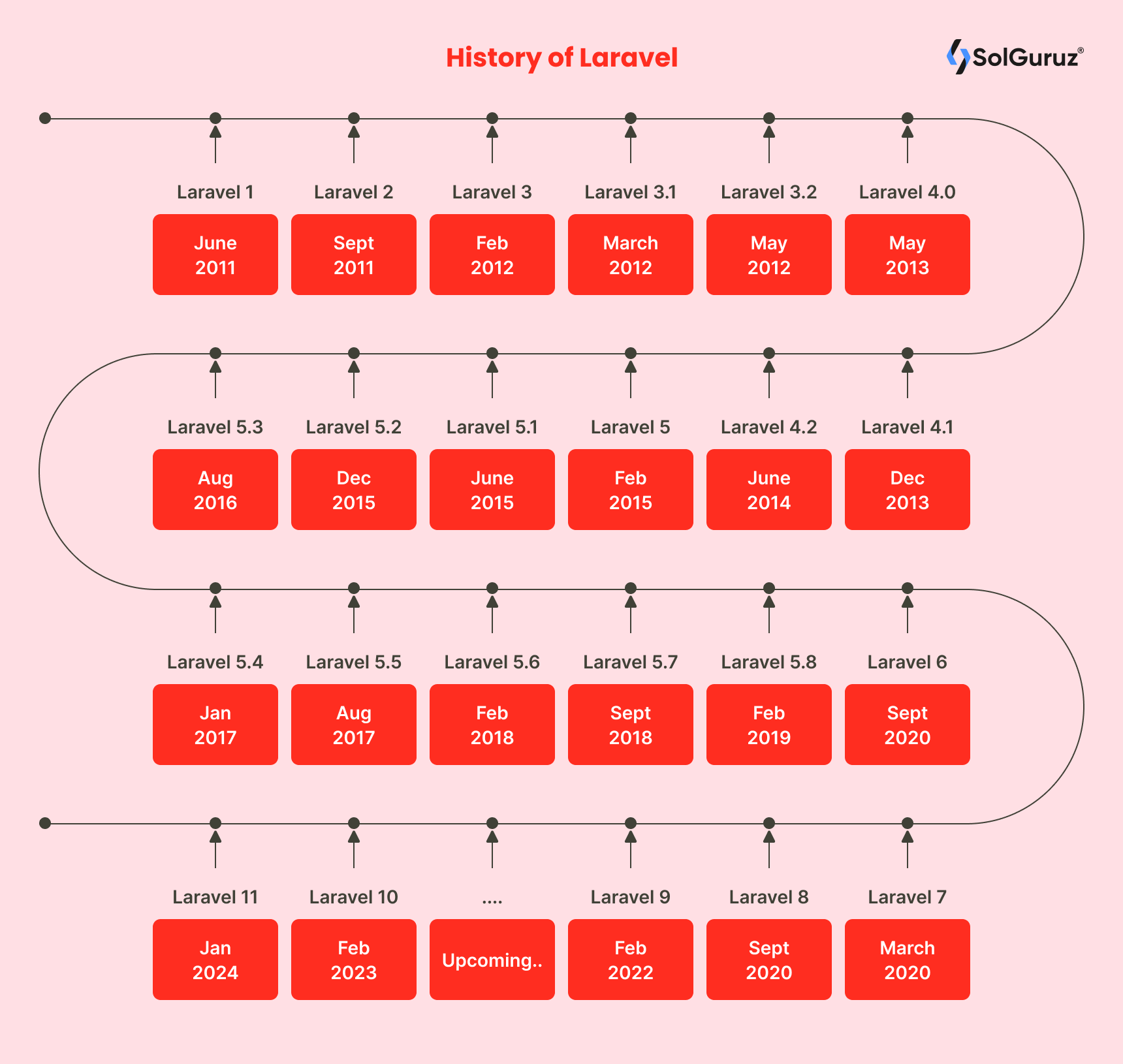
Tháng 9/2018 phát hành **Laravel 5.7** với những cập nhật như cải thiện thông báo lỗi, Callable Action URLs, Email Verification, bổ sung phương thức mới cho tùy chỉnh phân trang và thay đổi cấu trúc thư mục Resources.

Tháng 2/2019, **Laravel 5.8** là phiên bản quan trọng của Laravel với nhiều cập nhật như tính năng caching scaffolding, HasOneThrough, cải thiện và quản lý các phản hồi xử lý công viêc linh hoạt hơn, cải thiện php artisan serve hỗ trợ với nhiều máy chủ cùng một lúc.

Tháng 9/2019, phát hành **Laravel 6** cập nhật một số tính năng như job Middleware, Eloquent firstWhere Method, Semantic Versioning.

Năm 2020, là năm phát hành 2 phiên bản **Laravel 7** và **Laravel 8** cập nhật thêm nhiền tính năng quan trọng tiêu biểu như HTTP Client ,Multiple Mail Drivers, Laravel jetstream, Dynamic Blade Components và Time Testing Helpers.

Laravel vẫn phát triển và cập nhật nhiều phiên bản mới như năm 2022 phát hành **Laravel 9** và năm nay chúng ta có **Laravel 10** rất tiện ích nhằm thoải mãn các nhà phát triển phần mềm tập chung chủ yếu vào PHP và sắp tới năm 2024 dự kiến phát hành **Laravel 11**.



Hình 6: Lịch sử phát triển của Laravel theo từng giai đoạn

* + - 1. **Các lợi thế của Laravel**

Laravel cung cấp sự đơn giản và hiệu suất, đồng thời tích hợp nhiều tính năng mạnh mẽ giúp giảm bớt công sức và thời gian phát triển ứng dụng. Sự mở rộng, tích hợp dễ dàng, và cộng đồng hỗ trợ lớn làm cho Laravel trở thành lựa chọn phổ biến cho phát triển web trong cộng đồng PHP.

Laravel có tốc độ xử lý nhanh chóng đáp ứng được nhu cầu xây dựng website hay các dự án lớn trong thời gian ngắn, sử dụng dễ dàng dựa theo mẫu thiết kế mô hình 3 lớp MVC, hỗ trợ tốt các backend cache và nó tính bảo mật được đảm bảo. Với nhưng điều đó giúp Laravel càng ngày ngày phát triển đi đôi với sự tiện ích mà nó mang lại.

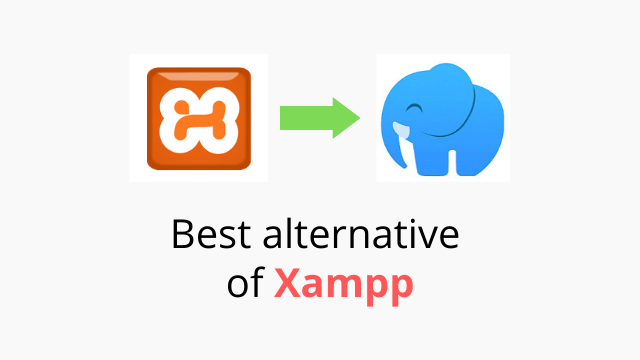
* + - 1. **Ưu điểm của Laravel**
* Cú pháp đẹp và dễ đọc
* sử dụng mô hình kiến trúc MVC, giúp tách biệt logic dữ liệu, giao diện người dùng và điều khiển luồng ứng dụng, giúp mã nguồn dễ quản lý hơn.
* Tương tác với cơ sở dữ liệu bằng cách sử dụng các đối tượng PHP thay vì truy vấn SQL trực tiếp.
* hệ thống định tuyến linh hoạt và dễ sử dụng, giúp quản lý các tuyến đường của ứng dụng một cách dễ dàng.
* hỗ trợ tích hợp các công cụ front-end để giúp xây dựng giao diện người dùng hấp dẫn.
  + - 1. **Nhược điểm của Laravel**
* việc học cú pháp và cách sử dụng một số tính năng phức tạp của Laravel có thể tốn thời gian
* Hiệu xuất đôi khi có thể chậm hơn các framework khác do tính năng mạnh mẽ và đa dạng của nó
* Laravel yêu cầu một số tài nguyên hệ thống tương đối lớn, đặc biệt khi xử lý các ứng dụng lớn và phức tạp.
* việc tuỳ chỉnh quá mức hoặc muốn thực hiện một số tính năng không tiêu chuẩn có thể gây khó khăn

## Tổng quan về Laragon

* + - 1. **Khái niệm**

Laragon là một môi trường phát triển web (local server) dành cho các nhà phát triển web sử dụng các công nghệ như PHP, MySQL, Apache (hoặc Nginx), và nhiều công cụ hữu ích khác. Điều mà tạo nên Laragon như vậy gắn liền với một số điểm quan trọng sau

* Hỗ trợ nhiều ngôn ngữ lập trình: Laragon chủ yếu được thiết kế cho phát triển web bằng PHP, nhưng nó cũng hỗ trợ các ngôn ngữ lập trình khác như Python, Ruby, Java, và nhiều ngôn ngữ khác thông qua các plugin bổ sung.
* Hỗ trợ nhiều cơ sở dữ liệu: Laragon hỗ trợ nhiều loại cơ sở dữ liệu như MySQL, MariaDB, PostgreSQL, MongoDB, Redis, và nhiều cơ sở dữ liệu khác.
* Hỗ trợ nhiều hệ điều hành: Laragon có phiên bản dành cho cả Windows và MacOS, giúp các nhà phát triển sử dụng môi trường phát triển thích hợp với hệ điều hành mà họ sử dụng.
* Giao diện người dùng đơn giản: Laragon đi kèm với giao diện người dùng thân thiện và dễ sử dụng, giúp người dùng quản lý các máy chủ và cài đặt một cách dễ dàng.
* Quản lý dự án dễ dàng: Laragon hỗ trợ tạo và quản lý nhiều dự án web cùng một lúc. Bạn có thể dễ dàng tạo mới các thư mục dự án, cài đặt các framework và thư viện, và mở các trình duyệt web một cách nhanh chóng.
* Cập nhật tự động và plugin mở rộng: Laragon thường cập nhật các phiên bản mới và hỗ trợ cài đặt các plugin bổ sung để mở rộng tính năng của nó.
* Hỗ trợ công cụ phát triển phổ biến: Laragon được tích hợp sẵn với các công cụ phát triển phổ biến như Git, Composer, Node.js, và nhiều công cụ khác để hỗ trợ quá trình phát triển ứng dụng.
* Hỗ trợ HTTPS và múi ngôn ngữ đa quốc gia: Laragon hỗ trợ cài đặt HTTPS cho các dự án, cũng như hỗ trợ nhiều ngôn ngữ khác nhau cho các ứng dụng web đa quốc gia
  + - 1. **Các lợi thế của Laragon**
* Laragon là ứng dụng tương thích với Laravel và được cho là sẽ thay thế Xampp bởi sự tiện dụng và đa dạng cách dùng.
* Dễ cài đặt và sử dụng: Laragon được thiết kế để đơn giản và dễ sử dụng, giúp nhà phát triển tiết kiệm thời gian trong việc cấu hình mô trường phát triển
* Tích hợp đầy đủ công cụ: Tích hợp sẵn các công cụ quan trọng như Composer, Artisan và các trình quản lý cơ sở dữ liệu, giúp tăng cường hiệu xuất và tiện ích.
* Tương thích tốt với Laravel: Laragon được tối ưu hóa cho phát triển Laravel, mang lại trải nghiệm phát triển mạnh mẽ và linh hoạt.
* Cộng đồng hỗ trợ tốt: Có một cộng đồng sử dụng rộng lớn và tích hợp sẵn trong cộng đồng Laravel, giúp người dùng nhanh chóng giải quyết vấn đề và chia sẻ kinh nghiệm.



Hình 7: Laragon được sánh hơn với Xampp

* + - 1. **Ưu điểm của Laragon**
* Dễ cài đặt và sử dụng
* Hỗ trợ nhiều ngôn ngữ và cơ sở dữ liệu
* Hỗ trợ nhiều hệ điều hành
* Giao diện người dùng đẹp và dễ dùng
* Hỗ trợ HTTPS và múi ngôn ngữ đa quốc gia
  + - 1. **Nhược điểm của Laragon**
* Hạn chế về tích hợp thư viện và gói mở rộng
* Khả năng mở rộng có hạn
* Có thể không phù hợp cho các dự án đòi hỏi cấu hình phức tạp
  + 1. **Tổng quan về Bootstrap Framework**
       1. **Khái niệm**

Bootstrap là font-end framework, là một dụng cụ sưu tập miễn phí các công cụ để tạo ra các trang web và các ứng dụng web. Framework bao gồm HTML và CSS dựa trên các mẫu thiết kế cho kiểu chữ, hình thức, các button, các thành phần giao diện khác và mở rộng hơn tùy chọn JavaScript.

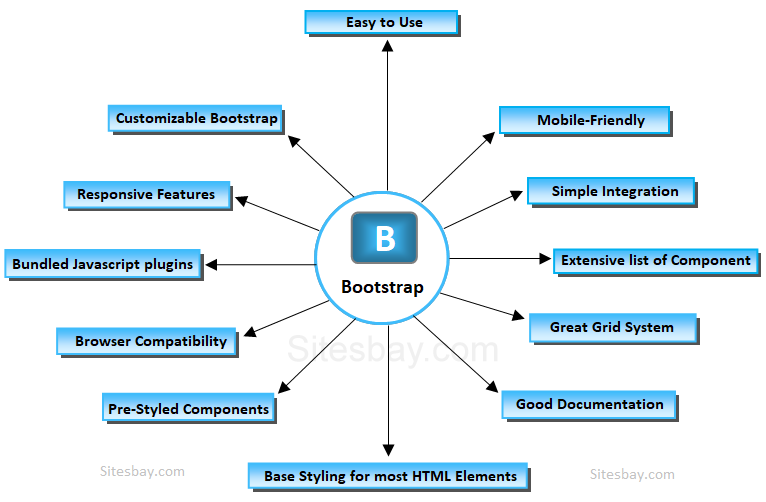
Bootstrap được đinh nghĩa sẵn các class CSS giúp người thiết kế giao diện website tiết kiệm được rất nhiều thời gian. Các thư viện Bootstrap có những đoạn mã có sẵn để áp dụng vào website mà không tốn quá nhiều thời gian để tự viết.

Với Framework việc phát triển giao diện website để phú hợp với đa thiết bị trở nên dễ dàng hơn bao giờ hết. Đây cũng là xu hướng phát triển giao diện website đang được ưa chuộng hiện nay. Bootstrap cung cấp tính năng responsive và mobile first, nghĩa là làm cho trang web có thể tự co giãn dể tương thích với mọi loại thiết bị khác nhau, từ điện thoại di động đến máy tính bảng, hay máy tính bàn, máy tính xách tay.

* + - 1. **Lịch sử phát triển**
* Ngày 19 tháng 8 năm 2011: Bootstrap được giới thiệu lần đầu tiên bởi Twitter với tên gọi Twitter Blueprint.
* Ngày 24 tháng 1 năm 2012: Sau một thời gian sử dụng nội bộ tại Twitter, Bootstrap được chính thức công bố là một dự án mã nguồn mở.
* Bootstrap 2 (Ngày 31 tháng 1 năm 2012): Bootstrap 2 được phát hành với nhiều cải tiến so với phiên bản trước đó, bao gồm cải thiện hiệu suất, bảng điều khiển (dashboard), và nhiều thành phần mới.
* Bootstrap 3 (Ngày 19 tháng 8 năm 2013): Bootstrap 3 mang lại sự linh hoạt và tính đáp ứng cao hơn. Nó chuyển sang sử dụng Flat Design và loại bỏ hỗ trợ cho trình duyệt Internet Explorer 7 và 8.
* Bootstrap 4 (Ngày 19 tháng 1 năm 2018): Bootstrap 4 là một bản cập nhật lớn với nhiều thay đổi đáng kể, bao gồm việc chuyển sang Flexbox, thay đổi đáng kể trong hệ thống lưới, và một loạt các cải tiến khác về hiệu suất và tính năng.
* Bootstrap 5 (Ngày 5 tháng 5 năm 2021): Bootstrap 5 tiếp tục mở rộng tính năng và cải thiện hiệu suất. Một số thành phố như jQuery đã được loại bỏ và thay thế bằng Vanilla JavaScript. Bootstrap 5 cũng chú trọng vào tích hợp tốt hơn với CSS custom properties.
  + - 1. **Các lợi thế của framework**

Bootstrap là một trong những framework được sử dụng nhiều nhất trên thế giới để xây dựng nên một website. Bootstrap đã xây dựng nên 1 chuẩn riêng và rất được người dùng ưa chuộng. Chính vì thế, chúng ta hay nghe tới một cụm từ rất thông dụng "Thiết kế theo chuẩn Bootstrap". Vì thế ta có thể thấy rõ những thuận lợi khi dùng bootstrap:

* Rất dễ để sử dụng: Nó đơn giản vì nó được base trên HTML, CSS và Javascript chỉ cẩn có kiến thức cơ bản về 3 cái đó là có thể sử dụng bootstrap tốt.
* Responsive: Bootstrap xây dựng sẵn reponsive css trên các thiết bị Iphones, tablets, và desktops. Tính năng này khiến cho người dùng tiết kiệm được rất nhiều thời gian trong việc tạo ra một website thân thiện với các thiết bị điện tử, thiết bị cầm tay.
* Tương thích với trình duyệt: Nó tương thích với tất cả các trình duyệt (Chrome, Firefox, Internet Explorer, Safari, and Opera). Tuy nhiên, với IE browser, Bootstrap chỉ hỗ trợ từ IE9 trở lên. Điều này vô cùng dễ hiểu vì IE8 không support HTML5 và CSS3.

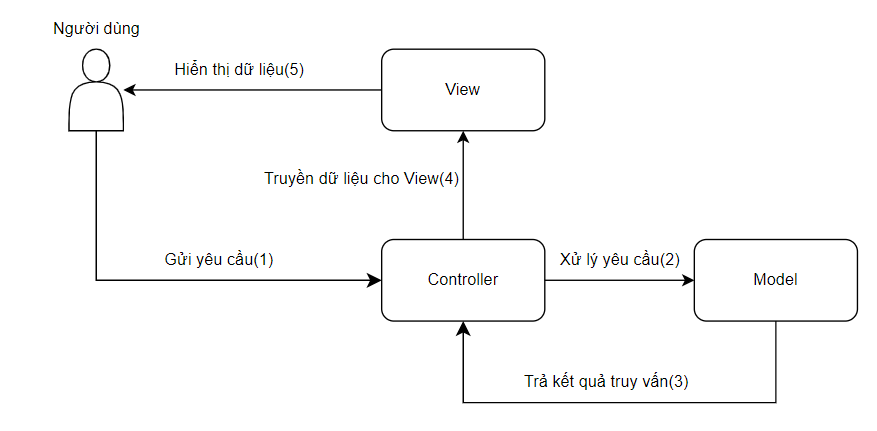


Hình 8: Các tính năng của Bootstrap Framework

**Chương II. Phân tích thiết kế hệ thống**

**2.1. Đặc tả hệ thống**

**2.1.1 Tổng quan hệ thống**



Hình 2.1: Tổng quan hệ thống

Hệ thống gồm 2 trang là trang quản trị và trang chính(HomePage)

Trang quản trị gồm

* Trang đăng nhập, đăng ký
* Trang quản lý người dùng
* Trang quản lý danh mục
* Trang quản lý khóa học
* Trang quản lý đánh giá

Trang quản lý có các chức năng như: tìm kiếm ,phân trang, thêm, sửa, xóa mềm, …

Trang chính gồm

* Trang chủ: gồm có danh sách các khóa học, các bên tài trợ, đối tác, thông tin website.
* Trang đăng ký, đăng nhập
* Trang chi tiết khóa học: hiển thị chi tiết như tên, tác giả, thông tin, mô tả, đánh giá, bình luận, tranh, gợi ý các đề xuất khóa học khác.
* Trang thông tin cá nhân: gồm có thống kê đánh giá mà người dùng đã đánh giá, số like mà người dùng like, thay đổi thông tin cá nhân, đổi mật khẩu.
* Trang tìm kiếm: Gồm có danh sách các khóa học.
* Trang hỗ trợ người dùng

**2.1.2. Cơ cấu hệ hống**

Hệ thống có 4 loại người dùng

|  |  |  |
| --- | --- | --- |
| **STT** | **Kiểu người dùng** | **Chức năng** |
| 1 | Người dùng không có tài khoản trên hệ thống | Khi truy cập vào website có thể xem chi tiết khóa học và tìm kiếm các khóa học. |
| 2 | Người dùng có tài khoản trên hệ thống | Ngoài sử dụng các chức năng cơ bản như xem chi tiết khóa học và tìm kiếm thì có thể bình luận, đánh giá, có trang cá nhân, like, sử dụng trang hỗ trợ người dùng. |
| 3 | Người dùng là Admin quản trị chính của hệ thống | Quản lý tài khoản người dùng và các nội dung chính trên hệ thống bao gồm khóa học, danh mục, đánh giá và hỗ trợ. |
| 4 | Người dùng là Moderator quản trị phụ của hệ thống | Kể thừa quản trị từ Admin bao gồm các nội dung chính nhưng không quản trị trang người dùng. |

Bảng 2.1: Phân loại người dùng hệ thống

**2.2. Sơ đồ Usecase**

**2.2.1. Usecase chi tiết**

Bảng vai trò người dùng có trong hệ thống.

|  |  |  |
| --- | --- | --- |
| **STT** | **Actor** | **Vai trò** |
| 1 | User | User là đối tượng chính tác động trực tiếp và nhiều nhất vào hệ thống.  User có thể thực hiện hầu hết các chức năng chính của hệ thống. |
| 2 | Admin | Admin là actor kế thừa của user và thêm các chức năng quản trị. |
| 3 | Moderator | Moderator là actor kế thừa của user và admin được thêm một vài các chức năng quản trị từ admin cấp quyền. |

Bảng 2.2: Danh sách người dùng

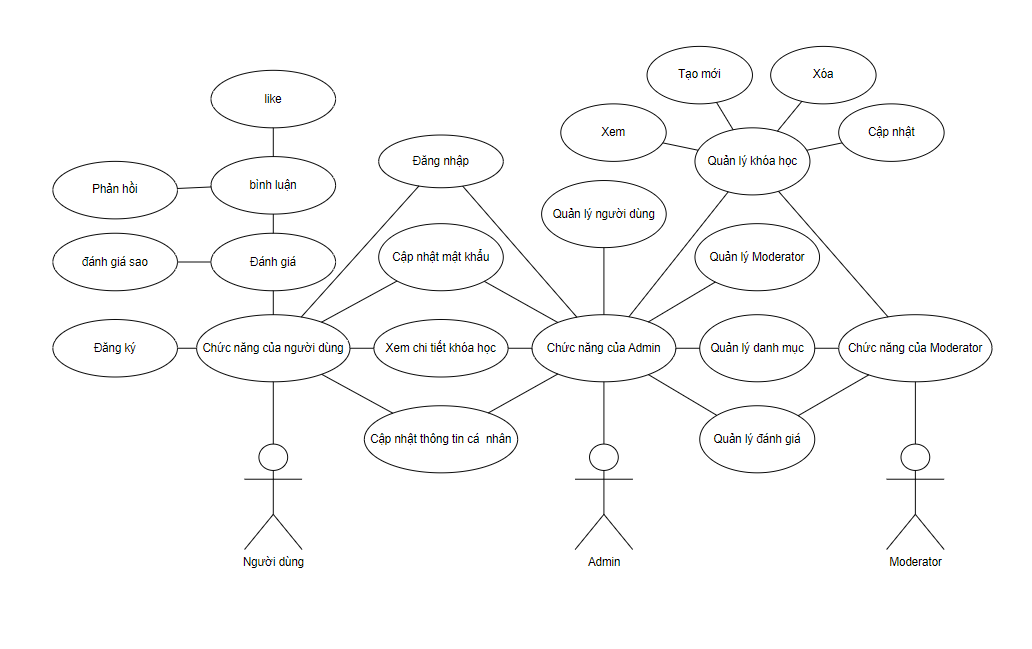
Bảng trình bày các chức năng chính của hệ thống.

|  |  |  |
| --- | --- | --- |
| **STT** | **Chức năng** | **Mô tả** |
| 1 | Register | Đăng ký tài khoản để sử dụng hệ thống. |
| 2 | Login | Phân quyền và xác thực cho người dùng để sử dụng hệ thống. |
| 3 | Logout | Thoát khỏi phiên làm việc |
| 4 | Change password | Thay đổi mật khẩu của người dùng |
| 5 | Add category | Thêm các danh mục mới vào kho dữ liệu |
| 6 | Edit category | Chỉnh sửa các danh mục đã có |
| 7 | Delete category | Xóa danh mục đã có trong lưu trữ |
| 8 | Add course | Thêm các khóa học mới vào kho dữ liệu |
| 9 | Edit course | Chỉnh sửa các khóa học có trong lưu trữ |
| 10 | Delete course | Xóa khóa học đã có trong lưu trữ |
| 11 | Add user | Thêm người dùng mới vào kho dữ liệu |
| 12 | Edit user | Chỉnh sửa người dùng có trong lưu trữ |
| 13 | Delete user | Xóa người dùng đã có trong lưu trữ |
| 14 | Search course | Tìm kiếm các khóa học có trong kho lưu trữ theo các định dạng như tên, tác giả, danh mục … |
| 15 | Review course | Đánh giá khóa học bao gồm có tên(nghệ danh), bình luận và đánh giá số sao(trên thang điểm 10) |
| 16 | Like course | Nhấn tăng số like bình luận trong chi tiết khóa học |
| 17 | Add reply | Thêm phản hồi vào bình luận trong chi tiết khóa học |
| 18 | Edit comment and reply | Chỉnh sửa bình luận và phản hồi bình luận có kho dữ liệu lưu trữ. |
| 19 | Delete comment and reply | Xóa bình luận và phản hổi bình luận có kho dữ liệu lưu trữ. |

Bảng 2.3: Danh sách các usecase

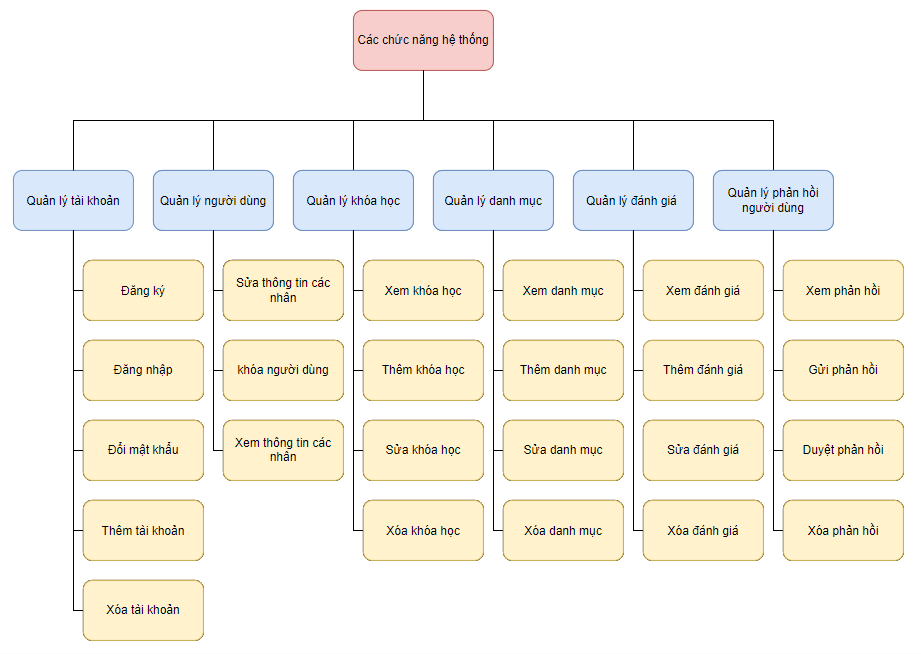
**2.2.2. Usecase quan hệ**

Sơ đồ mối quan hệ giữa actor với actor, actor với use case và use case với use case

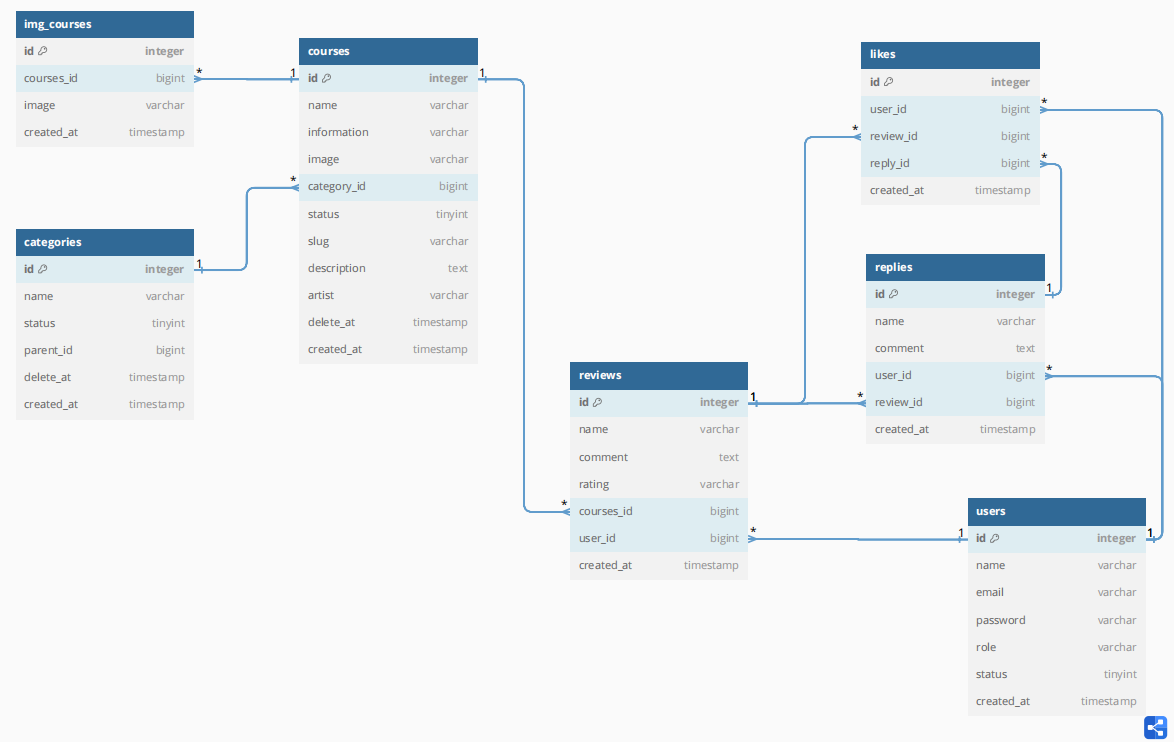


**2.3. Tổng quan các biểu đồ**

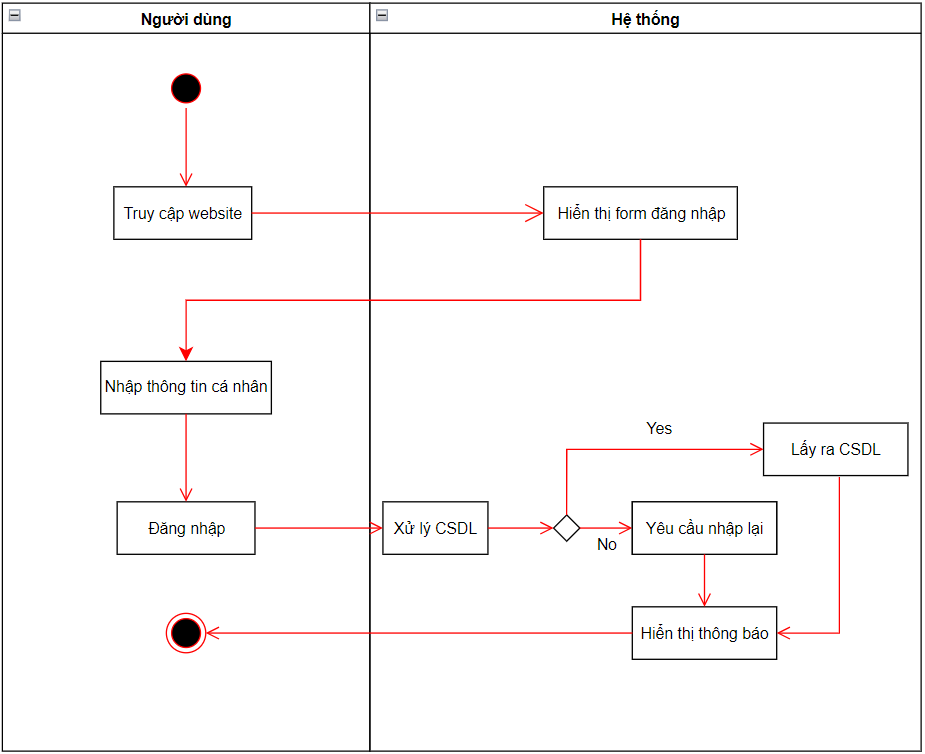
**2.3.1. Biểu đồ phân rã chức năng (FDD)**



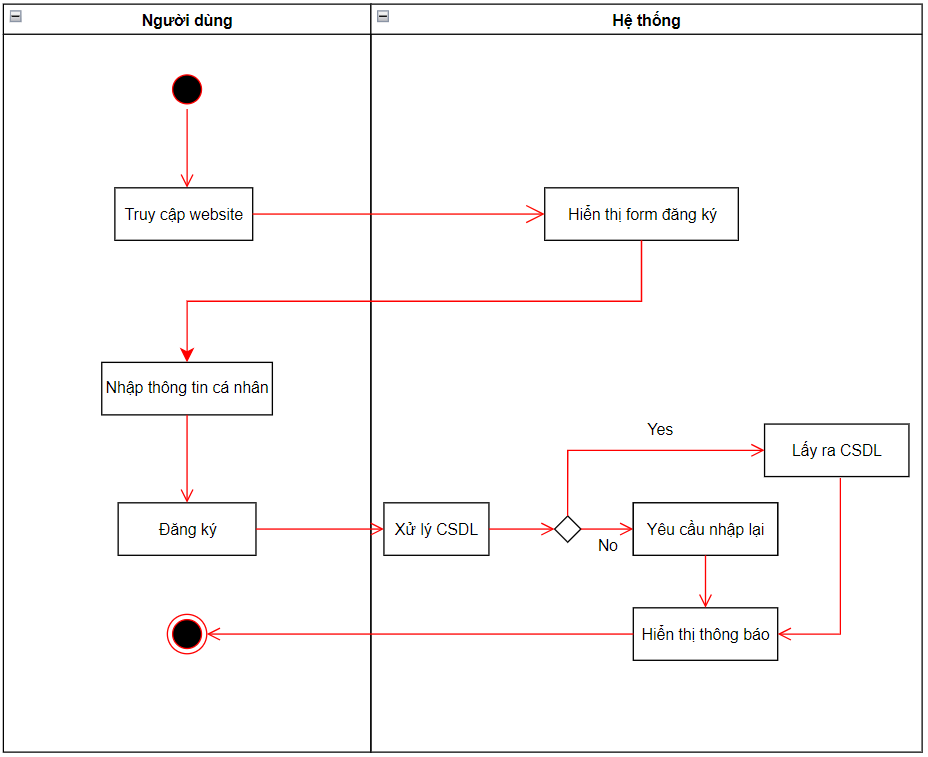
**2.3.2. Biểu đồ quan hệ thực thể (ERD)**



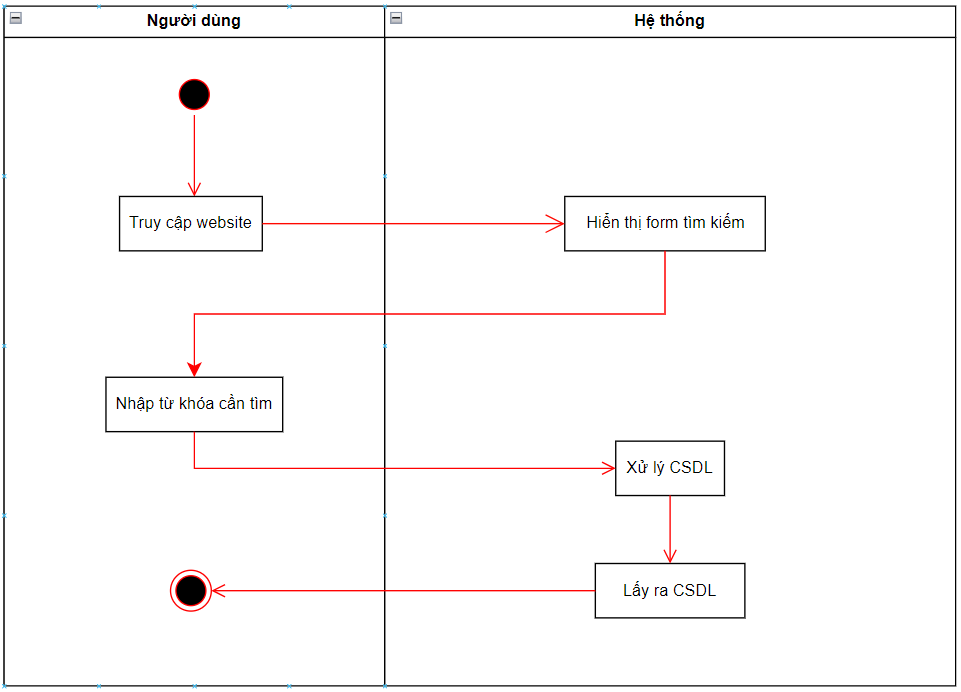
**2.3.3. Biểu đồ DFD**



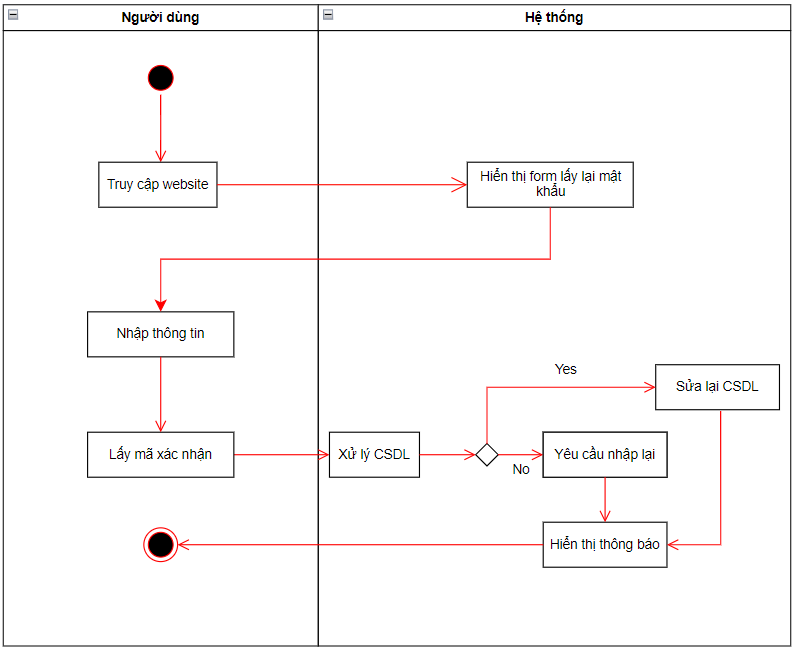
Hình 1: Sơ đồ luồng dữ liệu đăng nhập



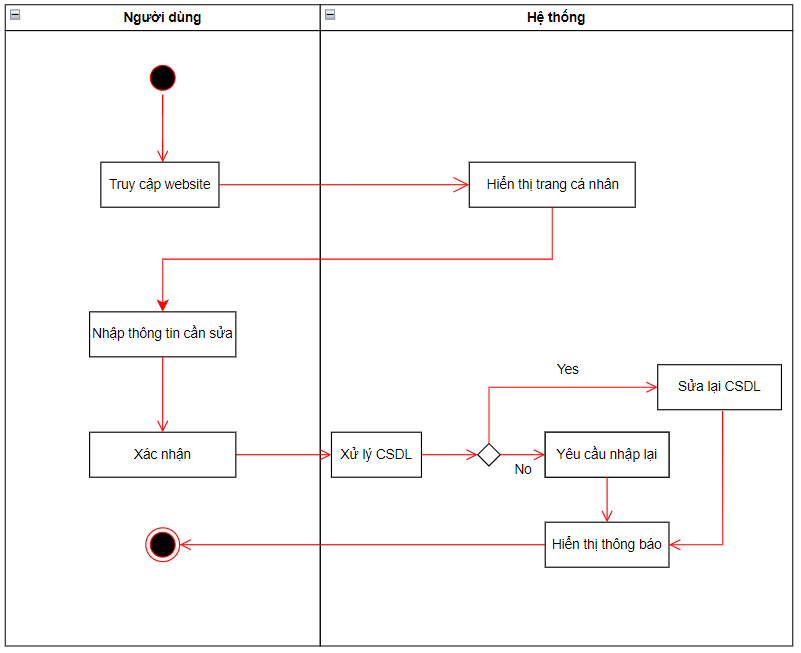
Hình 2: Sơ đồ luồng dữ liệu đăng ký



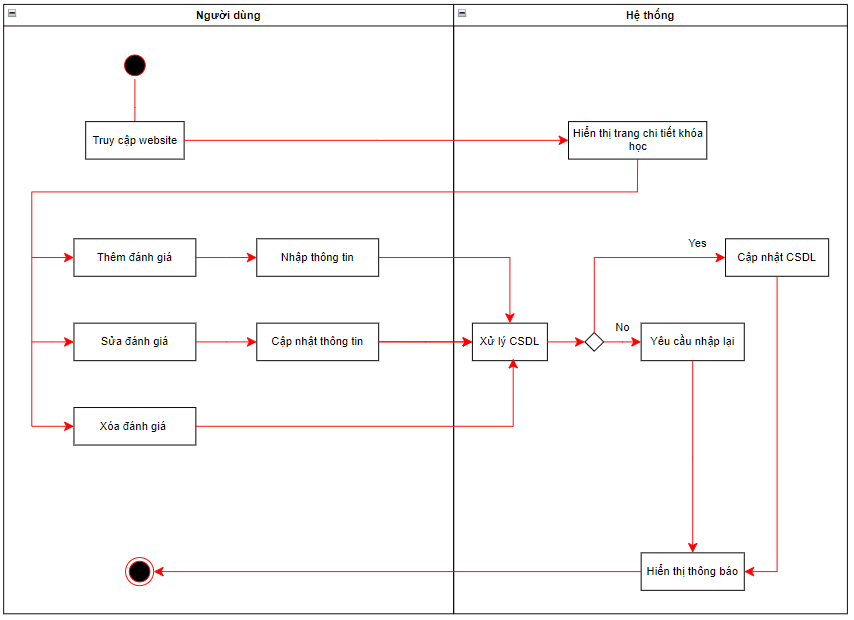
Hình 3:Sơ đồ luồng dữ liệu tìm kiếm



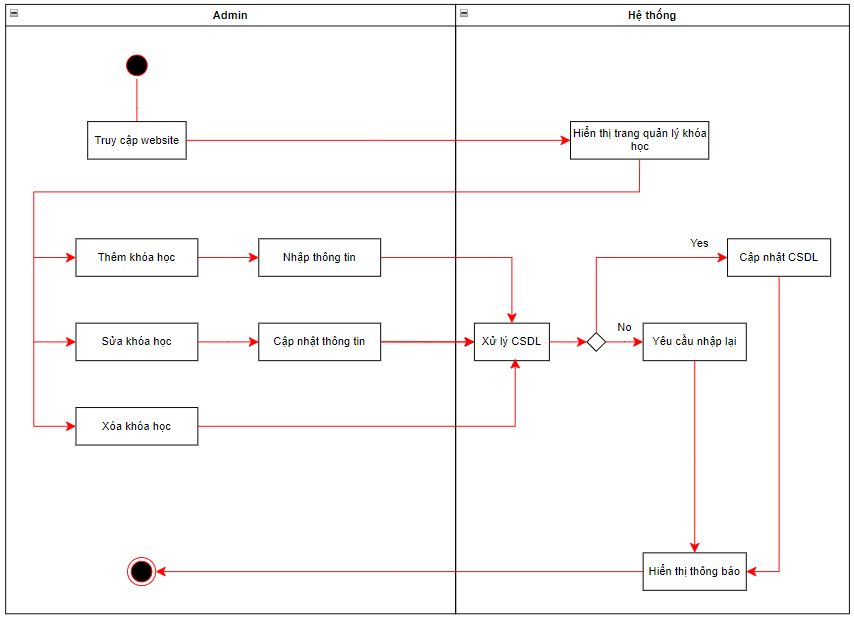
Hình 4: Sơ đồ luồng dữ liệu quên mật khẩu



Hình 5:Sơ đồ luồng dữ liệu chỉnh sửa thông tin cá nhân



Hình 6: Sơ đồ luồng dữ liệu đánh giá



Hình 7: Sơ đồ luồng dữ liệu quản lý khóa học

**2.4. Thiết kế cơ sở dữ liệu**

**2.4.1. Tổng hợp các bảng dữ liệu**

Bảng 1.1: Bảng Categories

Dùng để lưu trữ các danh mục có chứa các phong cách và trường phái nghệ thuật.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** |
| 1 | Id | BIGINT |
| 2 | Name | VARCHAR |
| 3 | Status | TINYINT |
| 4 | Parent\_id | BIGINT |

Bảng 1.2: Bảng Courses

Dùng để lưu trữ các khóa học vẽ Digital Painting.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** |
| 1 | Id | BIGINT |
| 2 | Name | VARCHAR |
| 3 | Information | VARCHAR |
| 4 | Image | VARCHAR |
| 5 | Category\_id | BIGINT |
| 6 | Status | TINYINT |
| 7 | Slug | VARCHAR |
| 8 | Description | TEXT |
| 9 | Artist | VARCHAR |

Bảng 1.3: Bảng Img\_courses

Dùng để lưu trữ ảnh mô tả của các khóa học.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** |
| 1 | Id | BIGINT |
| 2 | Course\_id | BIGINT |
| 3 | Image | VARCHAR |

Bảng 1.4: Bảng Users

Dùng để lưu trữ các tài khoản người dùng.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** |
| 1 | Id | BIGINT |
| 2 | Name | VARCHAR |
| 3 | Email | VARCHAR |
| 4 | Password | VARCHAR |
| 5 | Role | TINYINT |
| 6 | Status | TINYINT |

Bảng 1.5: Bảng Reviews

Dùng để lưu trữ các đánh giá như tên, bình luận, đánh giá sao.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** |
| 1 | Id | BIGINT |
| 2 | Name | VARCHAR |
| 3 | Comment | TEXT |
| 4 | Rating | VARCHAR |
| 5 | Course\_id | BIGINT |
| 6 | User\_id | BIGINT |

Bảng 1.6: Bảng Replies

Dùng để lưu trữ các hồi đáp lại các bình luận ở bảng Reviews.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** |
| 1 | Id | BIGINT |
| 2 | Name | VARCHAR |
| 3 | Comment | TEXT |
| 4 | User\_id | BIGINT |
| 5 | Review\_id | BIGINT |

Bảng 1.7: Bảng Likes

Dùng để lưu trữ số like có trong bình luận.

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên field** | **Kiểu dữ liệu** |
| 1 | Id | BIGINT |
| 2 | User\_id | BIGINT |
| 3 | Review\_id | BIGINT |
| 4 | Reply\_id | BIGINT |

**2.4.2. Quan hệ cơ sở dữ liệu**

**Chương 3.Thực nghiệm**