# CHƯƠNG 2: KIẾN THỨC NỀN TẢNG

Bằng việc sử dụng những kiến thức đã tiếp thu được trong quá trình tìm hiểu và học hỏi tại thời điểm hiện tại, em đã tổng hợp được kiến thức nền tảng quan trọng trong xây dựng chương trình và thiết kế cơ sở dữ liệu.



## 2.1. Cơ sở lý thuyết

Để xây dựng một website, có 3 yếu tố không thể thiếu đó là phân tích thiết kế, cơ sở dữ liệu và ngôn ngữ lập trình. Dựa vào các yếu tố trên em đã nghiên cứu các tài liệu và nêu ra các trọng điểm chính sau.

### 2.1.1. Tìm hiểu phân tích thuyết kế hướng đối tượng UML

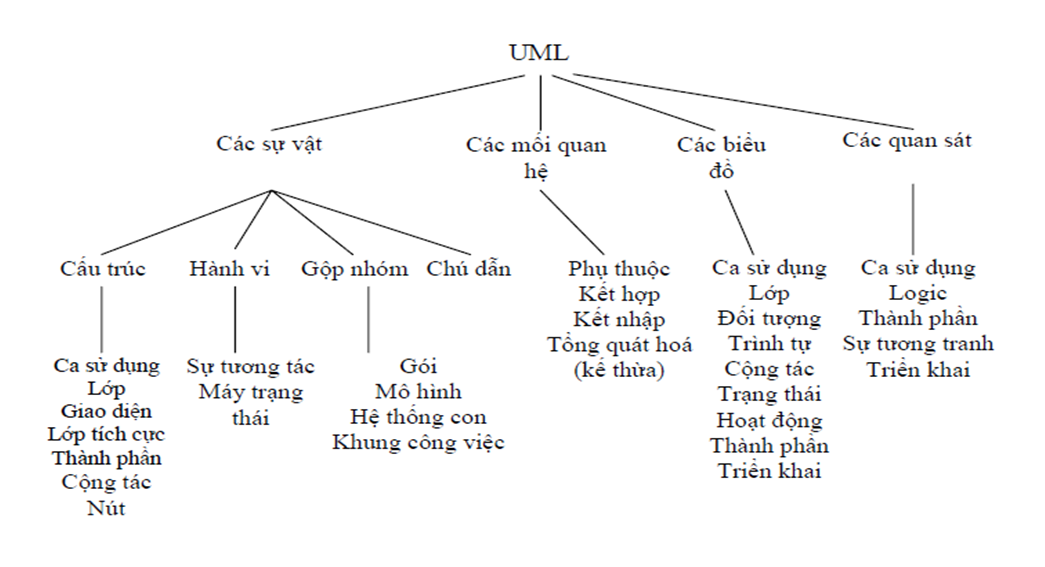
Khái niệm phân tích thiết kế hướng đối tượng(OOAD): phân tích xem hệ thống gồm những đối tượng nào trong trong hệ thống và chúng tương tác, liên kết với nhau như thế nào, từ việc mô tả được tất cả các đối tượng và sự tương tác của chúng sẽ giúp chúng ta hiểu rõ về hệ thống và cài đặt được nó.

Ưu điểm của phân tích thiết kế hướng đối tượng: gần gũi với thế giới thực, dễ tái sử dụng, có thể thừa kế từ đó làm giảm chi phí và tính mở cao, hệ thống đáng tin cậy hơn.

Nhược điểm: phức tạp, khó theo dõi luồng dữ liệu hơn phân tích thiết kế hệ thống hướng chức năng.

Khái niệm UML(Unified Modeling Language): Ngôn ngữ mô hình hóa thống nhất, là ngôn ngữ dùng để đặc tả, trực quan hóa và tư liệu hóa phần mềm hướng đối tượng [1].

Các phần tử của UML



**Hình 2. 1.** Các phần tử của UML [3]

* + ***Các quan sát (góc nhìn, view)*** theo các phương diện khác nhau của hệ thống cần phân tích, thiết kế. Dựa vào các quan sát để thiết lập kiến trúc cho hệ thống cần phát triển. Có năm loại quan sát: quan sát theo ca sử dụng, quan sát logic, quan sát thành phần, quan sát tương tranh và quan sát triển khai [3]*.*
    - Quan sát các ca sử dụng ( Usecase View): Mô tả các chức năng, nhiệm vụ của hệ thống. Quan sát này thể hiện mọi yêu cầu của hệ thống.
    - Quan sát Logic (Logical View) biểu diễn cách tổ chức logic của các lớp và các quan hệ của chúng với nhau. Nó mô tả cấu trúc tĩnh của các lớp, đối tượng và sự liên hệ của chúng thể hiện mối liên kết động thông qua sự trao đổi các thông điệp.
    - Quan sát thành phần (Component View) xác định các mô đun vật lý hay tệp mã chương trình và sự liên hệ giữa chúng để tổ chức thành hệ thống phần mềm.
    - Quan sát tiến trình (Process view) biểu diễn sự phân chia các luồng thực hiện công việc, các lớp đối tượng cho các tiến trình và sự đồng bộ giữa các luồng trong hệ thống.
    - Quan sát triển khai( Deployment view) mô tả sự phân bổ tài nguyên và nhiệm vụ trong hệ thống. Nó liên quan đến các tầng kiến trúc của phần mềm.
* **Biểu đồ (Diagram) là**: đồ thị biểu diễn đồ họa về tập các phần tử trong mô hình và mối quan hệ của chúng. Biểu đồ chứa đựng các nội dung của các quan sát dưới các góc độ khác nhau, một thành phần của hệ thống có thể xuất hiện trong một hay nhiều biểu đồ, có 9 loại biểu đồ khác nhau và được sử dụng kết hợp với nhau trong các trường hợp để cung cấp tất cả các hướng nhìn của một hệ thống[3].
* Biểu đồ casử dụng (Usecase diagram) mô tả sự tương tác giữa các tác nhân ngoài và hệ thống thông qua các ca sử dụng.
* Biểu đồ lớp (Class diagram): mô tả cấu trúc tĩnh, mô tả mô hình khái niệm bao gồm các lớp đối tượng và các mối quan hệ của chúng trong hệ thống hướng đối tượng.
* Biểu đồ đối tượng (Object diagram): là một phiên bản của biểu đồ lớp và thường cũng sử dụng các ký hiệu như biểu đồ lớp
* Biểu đồ trình tự (Sequence diagram) thể hiện sự tương tác của các đối tượng với nhau.
* Biểu đồ cộng tác (Collaboration diagram) tương tự như biểu đồ trình tự nhưng nhấn mạnh vào sự tương tác của các đối tượng trên cơ sở cộng tác với nhau bằng cách trao đổi các thông điệp để thực hiện các yêu cầu theo ngữ cảnh công việc.
* Biểu đồ trạng thái (State diagram): thể hiện chu kỳ hoạt động của các đối tượng, của các hệ thống con và của cả hệ thống.
* Biểu đồ hành động (Activity diagram) chỉ ra dòng hoạt động của hệ thống.
* Biểu đồ thành phần (Component diagram) chỉ ra cấu trúc vật lý của các thành phần trong hệ thống.
* Biểu đồ triển khai (Deployment Diagram) chỉ ra cách bố trí vật lý các thành phần theo kiến trúc được thiết kế của hệ thống.
* ***Các mối quan hệ (Relationship)***
  + **Hiện thực hoá (Realization):** là quan hệ ngữ nghĩa giữa giao diện và lớp.
  + **Phụ thuộc (Dependency):** là quan hệ ngữ nghĩa giữa hai phần tử, trong đó sự thay đổi của một tử sẽ tác động đến ngữ nghĩa của phần tử phụ thuộc.
  + **Kế thừa (Generalization):**  là quan hệ mô tả sự khái quát hoá mà trong đó một số đối tượng cụ thể (của lớp con) sẽ được kế thừa các thuộc tính, các phương thức của các đối tượng tổng quát (lớp cơ sở).
  + **Kết hợp (Association)** là quan hệ cấu trúc xác định mối liên kết giữa các lớp đối tượng.
* **Các sự vật (Các phần tử mô hình\ model element):** Các khái niệm được sử dụng trong các biểu đồ được gọi là các phần tử mô hình, thể hiện các khái niệm hướng đối tượng quen thuộc. Một phần tử mô hình thường được sử dụng trong nhiều biểu đồ khác nhau, nhưng nó luôn luôn có chỉ một ý nghĩa và một kí hiệu.

### 2.1.2. Ngôn ngữ thiết kế website HTML

HTML (HyperText Markup Language – ngôn ngữ siêu văn bản) là một trong các loại ngôn ngữ được sử dụng trong lập trình web. HTML giúp người dùng tạo và cấu trúc các thành phần trong trang web hoặc ứng dụng, phân chia các đoạn văn, heading, links, blockquotes,..

HTML không phải là ngôn ngữ lập trình, đồng nghĩa với việc nó không thể tạo ra các chức năng “động” được và chỉ có thể được sử dụng để bố cục và định dạng trang web. Ưu điểm nổi trội nhât và cũng là thế mạnh của HTML là khả năng xây dựng cấu trúc và khiến trang web đi vào quy củ một hệ thống hoàn chỉnh [6].

### 2.1.3. CSS

CSS là viết tắt của Cascading Style Sheets. Đây là một ngôn style sheet được sử dụng để mô tả giao diện và định dạng của một tài liệu viết bằng ngôn ngữ đánh dấu (markup), cung cấp một tính năng bổ sung cho HTML. CSS thường được sử dụng với HTML để thay đổi phong cách của trang web và giao diện người dùng. Nó cũng có thể được sử dụng với bất kỳ loại tài liệu XML nào bao gồm cả XML đơn giản, SVG và XUL [6].

Trước đây khi chưa có CSS, những người thiết kế web phải trộn lẫn giữa các thành phần trình bày và nội dung với nhau. Nhưng với sự xuất hiện của CSS, người ta có thể tách rời hoàn toàn phần trình bày và nội dung. Giúp cho phần code của trang web cũng gọn hơn và quan trọng hơn cả là dễ chỉnh sửa hơn.

### 2.1.4. JAVASCRIPT và JQUERY

JavaScript là một ngôn ngữ lập trình được sử dụng để tạo ra những trang web tương tác, được tích hợp và nhúng trong HTML. JavaScript cho phép kiểm soát các hành vi của trang web tốt hơn so với khi chỉ có một mình HTML. JavaScript kết hợp vào HTML, chạy trên Windows, Macintosh và các hệ thống hỗ trợ Netscape khác [7].

JQuery là một thư viện JavaScript nhanh, nhỏ, đa nền tảng và giàu tính năng. JQuery được thiết kế để đơn giản hóa client-side scripting của HTML. Nó thực hiện những nhiệm vụ như HTML document traversal và manipulation, animation, event handling, AJAX một cách rất đơn giản bằng một API dễ sử dụng và có khả năng hoạt động trên nhiều loại trình duyệt khác nhau. Mục đích chính của JQuery là làm việc sử dụng JavaScript trên trang web trở nên dễ dàng hơn, giúp website có tính tương tác và hấp dẫn hơn.

### 2.1.5. PHP

PHP là viết tắt của chữ "Hypertext Preprocessor", đây là một ngôn ngữ lập trình được sử dụng để viết ở phía máy chủ (lập trình web). Hiện nay có rất nhiều ngôn ngữ có thể viết ở máy chủ như C#, Java, Python và PHP chính là một trong những ngôn ngữ của nhóm này.

PHP có cú pháp đơn giản, tốc độ nhanh và nhỏ gọn, dễ học và thời gian để tiếp cập PHP ngắn hơn so với các ngôn ngữ lập trình khác nên hiện nay PHP đang chiếm số lương lập trình viên tương đối lớn, có thể gọi là lớn nhất. Hiện PHP có rất nhiều các CMS mã nguồn mở như WordPress, OpenCart nên nó rất đa dạng và đáp ứng hầu hết các website thông thường từ blog cá nhân cho đến website giới thiệu công ty và bán hàng [2].

### 2.1.6. Cơ sở dữ liệu

Cơ sở dữ liệu là một hệ thống các thông tin có cấu trúc, được lưu trữ trên các thiết bị lưu trữ nhằm thỏa mãn yêu cầu khai thác thông tin đồng thời của nhiều người sử dụng hay nhiều chương trình ứng dụng chạy cùng một lúc với những mục đích khác nhau [1]

Việc sử dụng hệ thống CSDL này sẽ khắc phục được những khuyết điểm của cách lưu trữ dưới dạng hệ thống tập tin đó là: Giảm trùng lặp thông tin ở mức thấp nhất, đảm bảo tính nhất quán và toàn vẹn dữ liệu, đảm bảo dữ liệu được truy xuất theo nhiều cách khác nhau, từ nhiều người khác nhau và nhiều ứng dụng khác nhau, tăng khả năng chia sẻ thông tin.

#### *2.1.6.1 Tính chất*

Một cơ sở dữ liệu có các tính chất sau:

Một cơ sở dữ liệu biểu thị một khía cạnh nào đó của thế giới thực như hoạt động của một công ty, một nhà trường, một ngân hàng… Những thay đổi của thế giới thực phải được phản ánh một cách trung thực vào trong cơ sở dữ liệu. Những thông tin được đưa vào trong cơ sở dữ liệu tạo thành một không gian cơ sở dữ liệu hoặc là một “thế giới nhỏ” (miniworld).

Một cơ sở dữ liệu là một tập hợp dữ liệu liên kết với nhau một cách logic và mang một ý nghĩa cố hữu nào đó. Một cơ sở dữ liệu không phải là một tập hợp tuỳ tiện.

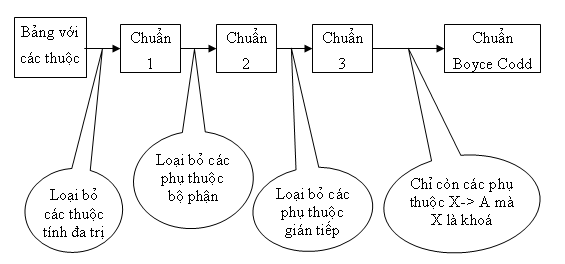
Một cơ sở dữ liệu được thiết kế và được phổ biến cho một mục đích riêng. Nó có một nhóm người sử dụng có chủ định và có một số ứng dụng được xác định phù hợp với mối quan tâm của người sử dụng. Nói cách khác, một cơ sở dữ liệu có một nguồn cung cấp dữ liệu, một mức độ tương tác với các sự kiện trong thế giới thực và một nhóm người quan tâm tích cực đến các nội dung của nó.

Một cơ sở dữ liệu có thể có cỡ tuỳ ý và có độ phức tạp thay đổi. Có những cơ sở dữ liệu chỉ gồm vài trăm bản ghi (như cơ sở dữ liệu phục vụ việc quản lý lương ở một cơ quan nhỏ), và có những cơ sở dữ liệu có dung lượng rất lớn (như các cơ sở dữ liệu phục vụ cho việc tính cước điện thoại, quản lý nhân sự trên một phạm vi lớn). Các cơ sở dữ liệu phải được tổ chức quản lý sao cho những người sử dụng có thể tìm kiếm dữ liệu, cập nhật dữ liệu và lấy dữ liệu ra khi cần thiết. Một cơ sở dữ liệu có thể được tạo ra và duy trì một cách thủ công và cũng có thể được tin học hoá. Một cơ sở dữ liệu tin học hoá được tạo ra và duy trì bằng bằng một nhóm chương trình ứng dụng hoặc bằng một hệ quản trị cơ sở dữ liệu [3].

#### *2.1.6.2. Chuẩn hóa cơ sở dữ liệu*

Chuẩn hoá là quá trình tách bảng (phân rã) thành các bảng nhỏ hơn dựa vào các phụ thuộc hàm. Các dạng chuẩn là các chỉ dẫn để thiết kế các bảng trong CSDL.

Mục đích của chuẩn hoá là loại bỏ các dư thừa dữ liệu và các lỗi khi thao tác dư thừa và các lỗi khi thao tác dữ liệu (Insert, Delete, Update). Nhưng chuẩn hoá làm tăng thời gian truy vấn.



**Hình 2. 2.** Chuẩn hóa dữ liệu [3]

## 2.2. Công cụ sử dụng

Để thực thiết kế được website, công cụ chủ yếu mà chúng em sẽ dùng đến là MySQL, cùng với framework đơn giản và dễ tiếp cận - CodeIgniter và phần mềm hỗ trợ phân tích thiết kế IBM Rational Rose.

### MySQL

Mysql là một hệ quản trị CSDL được sử dụng rất rộng rãi trên thế giới, đặc biệt là các ứng dụng website và nó thường đi kết hợp với ngôn ngữ lập trình PHP để xây dựng các ứng dụng website. Các hệ thống web ưa chuộng MYSQL là tại vì tốc độ xử lý của nó cao, tính dễ sử dụng và thương thích với các hệ điều hành thông dụng hiện nay như Linix, Window, v.v. Mysql có sử dụng ngôn ngữ truy vấn T-SQL để thao tác dữ liệu [3].

***SQL*** (Struct Querry Language): là ngôn ngữ truy vấn mang tính cấu trúc. Nó được thiết kế để quản lý dữ liệu trong một hệ thống quản lý cơ sở dữ liệu quan hệ (RDBMS). SQL là ngôn ngữ cơ sở dữ liệu, được sử dụng để tạo, xóa trong cơ sở dữ liệu, lấy các hàng và sửa đổi các hàng [3].

 SQL được sử dụng để điều khiển tất cả các chức năng mà một hệ quản trị cơ sở dữ liệu cung cấp cho người dùng bao gồm:

* ***Định nghĩa dữ liệu****:* SQL cung cấp khả năng định nghĩa các cơ sở dữ liệu, các cấu trúc lưu trữ và tổ chức dữ liệu cũng như mối quan hệ giữa các thành phần dữ liệu.
* ***Đảm bảo toàn vẹn dữ liệu****:* SQL định nghĩa các ràng buộc toàn vẹn trong cơ sở dữ liệu nhờ đó đảm bảo tính hợp lệ và chính xác của dữ liệu trước các thao tác cập nhật cũng như các lỗi của hệ thống.
* ***Điều khiển truy cập****:* SQL có thể được sử dụng để cấp phát và kiểm soát các thao tác củangười sử dụng trên dữ liệu, đảm bảo sự an toàn cho cơ sở dữ liệu.
* ***Truy xuất và thao tác dữ liệ****u:* Với SQL, người dùng có thể dễ dàng thực hiện các thao tác truy xuất, bổ sung, cập nhật và loại bỏ dữ liệu trong các cơ sở dữ liệu.

Các lệnh trong SQL: Căn cứ vảo bản chất của các lệnh để phân chia ra làm các loại

* Ngôn ngữ định nghĩa dữ liệu (Data Definition Language - DLL): gồm có các lệnh:
* Create: Lệnh tạo mới một đối tượng trong Database.
* Alter: Lệnh sửa đổi một đối tượng trong Database.
* Drop: Lệnh xóa một đối tượng trong Database.
* Ngôn ngữ điều khiển dữ liệu (Data Control Language – DCL): gồm có các lệnh:
* Grant: Lệnh trao quyền cho người dùng.
* Revoke: Lệnh thu hồi quyền đã trao cho người dùng.
* Ngôn ngữ thao tác dữ liệu (Data Manipulation Language - DML) gồm có các lệnh:
* Select : Lấy bản ghi.
* Insert : Tạo một bản ghi.
* Update: Sửa dổi bản ghi.
* Delete: Xóa bản ghi.
* Truncate: Xóa toàn bộ dữ liệu trong bảng.
* Ngôn ngữ điều khiển giao dịch (Transaction Control Language - TCL)
* Commit: Ủy thác giao dịch.
* Rollback: Lùi ngược lại giao dịch.
* Save Transaction: Đánh dâu một điểm trong giao dịch.
* Ngôn ngữ lập trình: Gồm các lệnh:
* Declare: Khái báo biến.
* Execute: Thực thi câu lệnh.
* Open: Mở một con trỏ để truy vấn kết quả truy vấn.
* Close: Đóng một con trỏ.
* Fetch: Đọc một dòng trong kết quả truy vấn.

**Một số khái niệm liên quan tới MySQL**

* Cơ sở dữ liệu ( Database) là một tập hợp dữ liệu đã được tổ chức sắp xếp có tính hệ thống. Được lưu trữ trên các thiết bị lưu trữ nhằm thõa mãn yêu cầu khai thác thông tin đồng thời của nhiều người sử dụng hay nhiều chương trình ứng dụng chạy cùng một lúc với những mục đích khác nhau:
* Bảng ( Table) : Là một thành phần cơ bản trong chương trình quản trị cơ sở dữ liệu quan hệ. Bảng được hình thành khi sắp xếp các thông tin có liên quan với nhau theo hàng và cột. Các hàng tương ứng với các bản ghi (record) dữ liệu và các cột tương ứng với trường dữ liệu.
* Bản ghi (Record) : là một đơn vị hoàn chỉnh nhỏ nhất của dữ liệu, được lưu trữ trong những trường hợp dữ liệu đã được đặt tên.
* Trường dữ liệu ( Field): là không gian dành cho một mẫu thông tin.
* Kiểu dữ liệu (Datatype) : Là một kiểu giá trị mà một cột có thể chứa, MySQL cung cấp cho chúng ta các loại kiểu dữ liệu như:
* Binary : Là kiểu dữ liệu chứa dạng số ở hệ Hexa, gồm 3 kiểu dữ liệu Binary, Varbinary, Image.
* Text : Là kiểu ký tự, chữ cái, kí hiệu, bao gồm các kiểu dữ liệu : Char, Varchar, Nvarchar, Text, Ntext.
* Data/Time: Kiểu dữ liệu thời gian, gồm: Datetime, SmallDatatTime.
* Numeric: Kiểu dữ liệu số, gồm : Int, SmallInt, TinyInt, BigInt, Fload, Real, Decimal,Numeric.
* Monetary : Kiểu dữ liệu tiền tệ, gồm : Money, SmallMoney.
* Bit : Kiểu Logic: Có (True) hoặc Không (False).
* variant : là kiểu dữ liệu xác định theo kiểu dữ liệu khác, một cột dữ liệu được định nghĩa dữ liệu kiểu này có thể lưu trữ nhiều dữ liệu có kiểu khác nhau trong cùng một bảng.
* Timestamp: Là kiểu dữ liệu có kích thước 8 byte, lưu trữ dạng số nhị phân do hệ thống tự sinh ra, mỗi giá trị Timestamp trong CSDL là duy nhất.
* XML: Là kiểu dùng để lưu trữ dữ liệu dưới dạng Xml
* Thủ tục lưu trữ (Store Procedure) là một đối tượng trong cơ sở dữ liệu bao gồm một tập nhiều câu lệnh SQL được nhóm lại với nhau thành một nhóm với những khả năng: Các cấu trúc điều khiển (IF, WHILE, FOR) có thể được sử dụng trong thủ tục, bên trong thủ tục lưu trữ có thể sử dụng các biến như trong ngôn ngữ lập trình nhằm lưu giữ các giá trị tính toán được, các giá trị được truy xuất được từ cơ sở dữ liệu hoặc một tập các câu lệnh SQL được kết hợp lại với nhau thành một khối lệnh bên trong một thủ tục. Một thủ tục có thể nhận các tham số truyền vào cũng như có thể trả về các giá trị thông qua các tham số (như trong các ngôn ngữ lập trình). Khi một thủ tục lưu trữ đã được định nghĩa, nó có thể được gọi thông qua tên thủ tục, nhận các tham số truyền vào, thực thi các câu lệnh SQL bên trong thủ tục và có thể trả về các giá trị sau khi thực hiện xong.
* Hàm (Functions) : Hàm là một đối tượng trong cơ sở dữ liệu bao gồm một tập nhiều câu lệnh SQL được nhóm lại với nhau thành một nhóm. Điểm khác biệt giữa hàm và thủ tục là hàm trả về một giá trị thông qua tên hàm.
* Bẫy lỗi (Trigger) : một Trigger là một đối tượng chứa một tập các câu lệnh SQL và tập các câu lệnh này sẽ được thực thi khi Trigger được gọi. Điểm khác biệt giữa thủ tục lưu trữ và Trigger là: Các thủ tục lưu trữ được thực thi khi người sử dụng có lời gọi đến chúng còn các Trigger lại được “gọi” tự động khi xảy ra những giao tác làm thay đổi dữ liệu trong các bảng.
* Các ràng buộc (Constraints): Là các cơ chế được hệ thống cài đặt dựa trên máy chủ nhằm bảo vệ tính toàn vẹn của dữ liệu.
* Các khung nhìn – hay bảng ảo (Views): Là các truy vấn lưu trữ trong CSDL để có thể tham chiếu tới một hoặc nhiều bảng. Thường thể hiện hoặc không thể hiện các cột nào đó trong bảng hoặc là liên kết hai hoặc nhiều bảng với nhau.

### Phần mềm hỗ trợ phân tích thiết kế IBM LucidChart

LucidChart là một công cụ lập mô hình trực quan mạnh trợ giúp phân tích và thiết kế các hệ thống phần mềm hướng đối tượng. Nó được dùng để lập mô hình hệ thống trước khi viết mã (code). Dùng mô hình, ta có thể bắt kịp những thiếu sót về thiết kế, trong khi việc chỉnh sửa chúng vẫn chưa tốn kém.

LucidChart bao gồm tất cả các sơ đồ UML, các actor, các use case, các đối tượng, các lớp, các thành phần... Nó mô tả chỉ tiết nội dung mà hệ thống sẽ gộp và cách nó sẽ làm việc.

Giống như một căn nhà có nhiều bản thiết kế mẫu cho phép các thành viên trong đội xây dựng xem xét nó từ nhiều góc nhìn khác nhau như : hệ thống ống nước, hệ thống điện, hệ thống nền ... LucidChart chứa đựng các sơ đồ khác nhau cho phép các thành viên trong nhóm đề án xem hệ thống từ các góc nhìn khác nhau như : Khách hàng, nhà thiết kế, quản trị đề án, v.v.

Khi đã có được bản thiết kế thì sẽ giảm bớt một số vấn đề phiền phức như: lập trình theo truyền thống thì khi hoàn tất đề án, sau một thời gian sử dụng khách hàng yêu cầu thêm một vài chức năng nào đó vì có cập nhật mới thì người lập trình phải xem lại toàn bộ hệ thống rồi sau đó mới cập nhật. Điều này tốn rất nhiều thời gian. Nay nhờ có bản thiết kế thì chỉ cần xem cập nhật đó nằm ở phần nào và chỉnh sửa, nâng cấp hệ thông. Điều đó sẽ linh hoạt và giảm rất nhiều thời gian, v.v.

### Framework Laravel

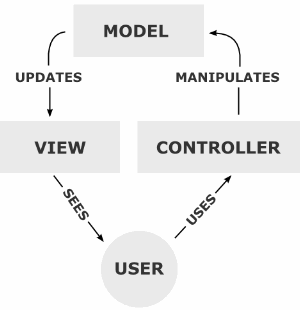
Laravel là một nền tảng ứng dụng web (web application framework) nguồn mở dựa trên nền tảng MVC (Model - View - Contrller) được dùng để xây dựng các ứng dụng web động tương tác với PHP. Nó cho phép các nhà phát triển xây dựng một ứng dụng web nhanh hơn - so với việc viết mã hỗn tạp - bằng cách cung cấp một bộ thư viện đầy đủ cho các tác vụ thông thường, cũng như cung cấp một mô hình tương tác đơn giản và dễ hiểu cho việc kết nối tới những bộ thư viện đó. Những tính năng nổi bật của Laravel bao gồm cú pháp dễ hiểu - rõ ràng, một hệ thống đóng gói Modular và quản lý gói phụ thuộc, nhiều cách khác nhau để truy cập vào các cơ sở dữ liệu quan hệ, nhiều tiện ích khác nhau hỗ trợ việc triển khai vào bảo trì ứng dụng [8].

**MVC:** Mô hình MVC là một kiến trúc phần mềm, được tạo ra với mục đích quản lý và xây dựng dự án phần mềm có hệ thống hơn. Mô hình MVC được dùng khá rộng rãi và đặc biệt là trong các ngôn ngữ lập trình Web. Mô hình MVC chia một ứng dụng ra ba thành phần khác nhau: Model, View và Controller. Với việc sử dụng mô hình MVC giúp cho mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần còn lại.

***Model:*** Đây là thành phần chứa tất cả các nghiệp vụ Logic, phương thức xử lý, truy xuất Database, đối tượng mô tả dữ liệu như các Class, hàm xử lý, v.v.

***View:*** Đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như Textbox, Images, v.v. Hiểu một cách đơn giản, nó là tập hợp các Form hoặc các file HTML.

***Controller:*** Giữ nhiệm vụ tiếp nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng.



**Hình 2. 3.** Mô hình MVC

# CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

Trong chương này em đã khảo sát một số sinh viên trường Đại học Công nghệ giao thông vận tải, link đăng ký đồ án tốt nghiệp trên goole dóc để rút ra các quy trình xử lý dữ liệu, các yêu cầu cần thiết khi xây dựng hệ thống website của mình, từ đó xây dựng biểu đồ ca sử dụng (Usecase Diagram), các bản đặc tả chi tiết, biểu đồ hoạt động, trình tự, trạng thái và biểu đồ lớp theo các yêu cầu chức năng hệ thống đã xác định.

## 3.1. Khảo sát hệ thống

### 3.1.1. Các thông tin cơ bản

Để tìm hiểu và khảo sát cụ thể cho website, em đã trực tiếp khảo sát tại khoa công nghệ thông tin, trường Đại học Công nghệ giao thông vận tải.

## 3.2. Yêu cầu xây dựng hệ thống

* **Trang chủ**
  + Trình bày thân thiện.
  + Hiển thị các chức năng nổi bật của website.
  + Hiển thị thông tin của trường Đại học Công nghệ giao thông vận tải (Số điện thoại, Địa chỉ liên hệ).
  + Các chức năng của Website (Tìm kiếm, Xác nhận đăng ký làm đề tài với giảng viên, vv..).
  + Đơn giản, định hướng, giúp người dùng dễ sử dụng.
* **Trang mô tả đề tài**
  + Trình bày thân thiện.
  + Trình bày chi tiết đề tài (Tên, mô tả chi tiết đề tài).
  + Sắp xếp, phân loại đề tài theo chuyên ngành.
* **Trang chi tiết đề tài**
* Trình bày thân thiện, dễ sử dụng.
* Thông tin đầy đủ, tất cả các thông tin về đề tài.
* Chứa chức năng đăng ký đồ án.
* **Trang quản trị**
* Quản lý tài khoản giáo viên, sinh viên.
* Quản lý đề tài.
* Quản lý lĩnh vực.
* Quản lý tin tức.
* Tổng hợp danh sách đề tài từ giảng viên.
* Phân quyền cho giáo viên, sinh viên.
* Thống kê, báo cáo.
* Xuất danh sách sau khi kết thúc đăng ký đề tài

## 3.3. Phân tích thiết kế

### 3.3.1. Xác định các tác nhân (Actor) và chức năng (UserCase)

Bảng 3.1: Xác định Actor và Usecase

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tác nhân** | **Mô tả tác nhân** | **Chức năng** |
| 1 | SinhVien | Sinh viên khoa công nghệ thông tin | - Đăng nhập  - Đăng ký tài khoản  - Đăng ký đề tài  - Xác nhận đăng ký làm đề tài với giảng viên.  - Nhận đề tài (Sau khi kết thúc phần đăng ký của sinh viên) |
| 2 | QuanTriVien | Giáo viên khoa công nghệ thông tin | - Tạo tài khoản  - Phân quyền cho GV, SV  - Tổng hợp danh sách đề tài từ giảng viên  - Post đề tài  - Phân đề tài  - Thống kê, báo cáo  - Xuất danh sách |
| 3 | GiaoVien | Giáo viên khoa công nghệ thông tin | - Đăng nhập  - Đăng ký tài khoản  - Post đề tài lên hệ thống  - Nhận danh sách hướng dẫn sinh viên từ bộ môn (Sau khi kết thúc phần đăng ký của sinh viên) |

### 3.3.2. Biểu đồ Usecase

#### 3.3.2.1. Biểu đồ Usecase tổng quát