

## LỜI CAM ĐOAN

Chúng em xin cam đoan dự án “ **Thanh toán online** ” là công trình nghiên cứu của chúng em trong thời gian qua, không sao chép lại từ cá nhân hoặc tổ chức nào khác. Những kết quả đạt được do chúng em đã cố gắng nghiên cứu, tìm hiểu, các số liệu, hình ảnh kết quả phù hợp với thực tế. Các kết quả này chưa từng công bố trong bất kỳ tài liệu nào khác. Chúng em xin hoàn toàn chịu trách nhiệm trước nhà trường về lời cam đoan này.

Tác giả Dự án

**Phạm Văn Huỳnh**

**Lê Thị Thương**

## LỜI CẢM ƠN

Chúng em xin gửi lời cảm ơn chân thành và sự tri ân sâu sắc đối với các thầy cô trường Cao Đẳng Viễn Đông, đặc biệt là các thầy cô khoa Công nghệ thông tin đã giúp chúng em hoàn thành chương trình học và đồ án tốt nghiệp này, và chúng em cũng xin chân thành cảm ơn thầy Phan Lâm Hoàng Ngân đã nhiệt tình hướng dẫn chúng em hoàn thành tốt đồ án tốt nghiệp.

Trong quá trình thực hiện đề tài, vì kiến thức bản thân còn hạn hẹp, cũng như kinh nghiệm còn chưa nhiều nên khi thực hiện đồ án này chúng em không tránh khỏi những sai sót chúng em rất mong được sự góp ý kiến của các thầy cô và các bạn để bài báo cáo của chúng em được hoàn thiện hơn.

Cuối cùng em kính chúc quý thầy cô và các bạn dồi dào sức khỏe và thành công trong sự nghiệp cao quý.

Em xin chân thành cảm ơn.

TP.Hồ Chí Minh, ngày .... tháng ... năm 2020

Sinh viên thực hiện

**Nhận xét của đơn vị thực tập**

**Nhận xét của giáo viên hướng dẫn**

## Mục lục

### LỜI MỞ ĐẦU

1. **Lý do chọn đề tài** ..... 10
2. **Ý nghĩa của đề tài** .....Error! Bookmark not defined.
3. **Mục đích** .....Error! Bookmark not defined.
4. **Đối tượng và phạm vi nghiên cứu của đề tài ...**Error! Bookmark not defined.
5. **Kết quả** .....Error! Bookmark not defined.

## PHẦN I. TỔNG QUAN VỀ CÔNG NGHỆ THIẾT KẾ

### CHƯƠNG 1: GIỚI THIỆU VỀ NGÔN NGỮ LẬP TRÌNH

- 1.1. **Giới thiệu lập trình PHP LARAVEL**Error! Bookmark not defined.
- 1.2. **Giới thiệu lập trình ANDROID** .....Error! Bookmark not defined.

### CHƯƠNG 2: GIỚI THIỆU VỀ CƠ SỞ DỮ LIỆU MYSQL

- 2.1 **Giới thiệu về MYSQL** .....Error! Bookmark not defined.
- 2.2 **Xây dựng cấu trúc dữ liệu** .....Error! Bookmark not defined.

### CHƯƠNG 3 GIỚI THIỆU VỀ THANH TOÁN ONLINE

**Thanh toán qua PAYPAL** .....Error! Bookmark not defined.

## PHẦN II: THIẾT KẾ VÀ XÂY DỰNG WEBSITE

### **CHƯƠNG 1: HỆ THỐNG QUẢN TRỊ TRANG WEB** .....Error!

Bookmark not defined.

**CHƯƠNG 2: HỆ THỐNG GIỚI THIỆU SẢN PHẨM**Error! Bookmark not defined.

## PHẦN III: THIẾT KẾ VÀ XÂY DỰNG APP ANDROID

### **CHƯƠNG 1: PHÂN TÍCH CÁC CHỨC NĂNG CỦA ỨNG DỤNG**

Error! Bookmark not defined.

**CHƯƠNG II: THANH TOÁN ONLINE QUA PAYPAL**Error! Bookmark not defined.

**PHẦN KẾT LUẬN:** .....Error! Bookmark not defined.

**HƯỚNG PHÁT TRIỂN CỦA ĐỀ TÀI:**Error! Bookmark not defined.

**TÀI LIỆU THAM KHẢO:**.....Error! Bookmark not defined.

# DANH MỤC CÁC TỪ VIẾT TẮT

STT	Từ viết tắt	Tên đầy đủ	Giải thích
1	API	Application Programming Interface	Giao diện lập trình ứng dụng.
2	DOM	Document Object Model	Các đối tượng thao tác văn bản
3	XHTML	Extensible HyperText Markup Language	Ngôn ngữ Đánh dấu Siêu văn bản Mở rộng
4	XML	eXtensible Markup Language	Ngôn ngữ đánh dấu mở rộng do W3C đề nghị với mục đích tạo ra các ngôn ngữ đánh dấu khác
5	SVG	Scalable Vector Graphics	Ngôn ngữ đánh dấu (markup language) XML và dùng để miêu tả các hình ảnh đồ họa véc tơ hai chiều, tĩnh và hoạt hình, thường dành cho ứng dụng trên các trang mạng.
6	UML	Unified Modeling Language	Ngôn ngữ mô hình gồm các ký hiệu đồ họa mà các phương pháp hướng đối tượng sử dụng để thiết kế các hệ thống thông tin một cách nhanh chóng.
7	CSDL	Cơ sở dữ liệu	

## PHẦN MỞ ĐẦU

### 1. Lý do chọn đề tài

Công việc kinh doanh buôn bán đã xuất hiện từ rất lâu, trải qua mỗi giai đoạn lịch sử nó mang một hình thức đặc thù riêng. Trước kia, khi các công cụ hỗ trợ công việc mua bán chưa phát triển mạnh, thì người kinh doanh mua bán chỉ diễn ra dưới hình thức mua bán trực tiếp. Từ khi khoa học công nghệ phát triển, nó đã tạo ra phát triển các loại hình thức mua bán mới, điển hình là mua bán trực tuyến. Hình thức mua bán trực tuyến hỗ trợ đắc lực cho người kinh doanh tiếp xúc với được nhiều khách hàng, còn khách hàng thì công việc mua bán được diễn ra nhanh chóng, thuận lợi, tiết kiệm được thời gian ...

Hiện nay, Thương Mại Điện Tử (TMĐT) đang được nhiều quốc gia quan tâm, coi là một trong những động lực phát triển chủ yếu của nền kinh tế. TMĐT đem lại những lợi ích tiềm tàng, giúp doanh nghiệp thu được thông tin phong phú về thị trường và đối tác, giảm chi phí tiếp thị và giao dịch, tạo dựng và củng cố quan hệ bạn hàng.

Việc mua hàng qua mạng chỉ với thủ tục đăng ký mua sắm đơn giản nhưng đem lại nhiều lợi ích: tiết kiệm và chủ động về thời gian, tránh khỏi những phiền phức khó chịu. Vậy nên việc mua bán hàng qua mạng đang rất được mọi người quan tâm.

Trên cơ sở các kiến thức được học trong nhà trường và quá trình tìm hiểu các website và apps bán hàng trong thực tế, chúng em đã quyết định chọn đề tài “*Thanh toán Online*” đặc biệt là thanh toán qua Paypal.

Phục vụ tốt hơn nhu cầu của khách hàng và quản lý của cửa hàng trong hoạt động kinh doanh. Khách hàng chỉ cần các thao tác đơn giản trên website và điện thoại di động là có thể đặt hàng được ngay.

## **2. Ý nghĩa của đề tài**

Trước hết, đề tài “*Thanh toán online*” là một ứng dụng được xây dựng như một đề tài thể hiện việc áp dụng những kiến thức quý báu đã được các thầy cô của trường Cao đẳng Viễn Đông nhiệt tình truyền thụ lại cho chúng em. Và đặc biệt là sự theo dõi và



quan tâm giúp đỡ của giảng viên hướng dẫn Phan Lâm Hoàng Ngân trong suốt thời gian chúng em thực hiện đề tài này.

Thứ hai là những kinh nghiệm quý báu mà chúng em có được trong quá trình thực hiện đề tài sẽ là hành trang tuyệt vời giúp ích rất nhiều cho công việc sau này của mình.

Thứ ba, trong thời đại công nghệ thông tin phát triển như vũ bão ngày nay, công nghệ không ngừng được cải tiến, những công nghệ mới ra đời đòi hỏi người lập trình viên phải cố gắng nghiên cứu và tìm hiểu để áp dụng vào thực tiễn. Điều này cũng được thể hiện qua sự cố gắng chúng em khi quyết định thực hiện đề tài này với các công nghệ được coi là tiềm năng và được dự báo sẽ phát triển mạnh mẽ trong tương lai.

Do vậy, chúng em rất hy vọng sẽ tạo nên một sản phẩm hoàn chỉnh nhất có thể để đáp lại sự nhiệt tình giúp đỡ từ thầy cô và cả tâm huyết của em khi quyết định thực hiện đề tài này.

### **3. Mục đích**

Sản phẩm của chúng em được thực hiện dựa trên hai mục đích chính sau :

- Thứ nhất là nghiên cứu các công nghệ lập trình như : lập trình ứng dụng với Android và lập trình PHP-Laravel.
- Thứ hai là phát triển ứng dụng có tính thực tiễn cao, có khả năng triển khai ứng dụng vào thực tế, giúp những người có nhu cầu mua bán hàng hóa có thể thao tác dễ dàng và tiện lợi.

### **4. Đối tượng và phạm vi nghiên cứu của đề tài**

#### **3.1 Đối tượng nghiên cứu**

- Đối tượng nghiên cứu của luận án là thanh toán Online, chủ yếu là thanh toán qua Paypal

#### **3.2 Phạm vi nghiên cứu**

- Phạm vi nghiên cứu là website bán hàng và apps bán hàng.

### **5. Kết quả**

- Xây dựng thành công website bán hàng và apps bán hàng.
- Tìm hiểu nắm bắt được các kiến thức về HTML, CSS, JQUERY, PHP, MYSQL, ANDROID...
- Ứng dụng , phát triển được công nghệ thiết kế web bằng Laravel.

- Ứng dụng , phát triển được công nghệ thiết kế APP ANDROID
- Với quy mô ứng dụng vừa và nhỏ, cộng thêm thời gian không cho phép nên đề tài “Thanh toán Online” có thể chưa có đầy đủ các chức năng hoàn thiện của một ứng dụng bán hàng online. Trong quá trình làm còn nhiều sai sót, do kiến thức có hạn lên website và apps còn ít nhiều thiếu sót về việc hiển thị cũng như tối ưu thuật toán, thời gian thực hiện và hình thức thanh toán. Trong tương lai em sẽ cố gắng khắc phục những sai sót này.
- Em hy vọng dựa trên nền tảng lý thuyết đã được thầy cô truyền thụ lại,kinh nghiệm thực tập, kết hợp với sự tìm hiểu công nghệ của cá nhân, luận văn tốt nghiệp của em sẽ đạt được mục đích mong đợi

## **PHẦN I: TỔNG QUAN VỀ CÔNG NGHỆ THIẾT KẾ**

### **CHƯƠNG 1. GIỚI THIỆU VỀ NGÔN NGỮ LẬP TRÌNH**

#### **1. GIỚI THIỆU LẬP TRÌNH PHP – LARAVEL**

##### **1.1 Lập trình Font-End**

### 1.1.1 Ngôn ngữ lập trình HTML

HTML (HyperText Markup Language) – Ngôn ngữ đánh dấu siêu văn bản được sử dụng để tạo các tài liệu có thể truy cập trên mạng. Tài liệu HTML được tạo nhờ dùng các thẻ và các phần tử của HTML. File được lưu trên máy chủ dịch vụ web với phần mở rộng “.htm” hoặc “.html”. Các trình duyệt sẽ đọc tập tin HTML và hiển thị chúng dưới dạng trang web. Các thẻ HTML sẽ được ẩn đi, chỉ hiển thị nội dung văn bản và các đối tượng khác: hình ảnh, media. Với các trình duyệt khác nhau đều hiển thị một tập HTML với một kết quả nhất định. Các trang HTML được gửi đi qua mạng internet theo giao thức HTTP. HTML không những cho phép nhúng thêm các đối tượng hình ảnh, âm thanh mà còn cho phép nhúng các kịch bản vào trong đó như các ngôn ngữ kịch bản như Javascript để tạo hiệu ứng động cho trang web. Để trình bày trang web hiệu quả hơn thì HTML cho phép sử dụng kết hợp với CSS. HTML không những cho phép nhúng thêm các đối tượng hình ảnh, âm thanh mà còn cho phép nhúng các kịch bản vào trong đó như các ngôn ngữ kịch bản như Javascript để tạo hiệu ứng động cho trang web.

HTML là một chuẩn ngôn ngữ internet được tạo ra và phát triển bởi tổ chức World Wide Web Consortium còn được viết tắt là W3C. Trước đó thì HTML xuất bản theo chuẩn của RFC. HTML được tương thích với mọi hệ điều hành cùng các trình duyệt của nó. Khả năng dễ học, dễ viết là một ưu điểm của HTML không những vậy việc soạn thảo đòi hỏi hết sức đơn giản, chúng ta có thể dùng word, notepad hay bất cứ một trình soạn thảo văn bản nào để viết và chỉ cần lưu với định dạng “.html “ hoặc “.htm” là đã có thể tạo ra một file chứa HTML. Hiện nay, phiên bản mới nhất của HTML là HTML5 với nhiều tính năng ưu việt so với các phiên bản cũ HTML cải tiến khá nhiều đặc biệt hỗ trợ mạnh mẽ các phần tử multimedia mà không cần các plugin. Một tập tin HTML bao gồm trong đó là các đoạn văn bản HTML, được tạo lên bởi các thẻ HTML. HTML5 nói chung mạnh mẽ hơn nhiều không chỉ về tốc độ và độ thích ứng cao mà chính là khả năng hỗ trợ API (Application Programming Interface - giao diện lập trình ứng dụng) và DOM (Document Object Model – các đối tượng thao tác văn bản).

### 1.1.2 Ngôn ngữ CSS

CSS (Cascading Style Sheets) là một ngôn ngữ quy định cách trình bày cho các tài liệu viết bằng HTML, XHTML, XML, SVG, hay UML,...CSS quy định cách hiển thị của các thẻ HTML bằng cách quy định các thuộc tính của các thẻ đó (font chữ, kích thước, màu sắc...).

CSS có cấu trúc đơn giản và sử dụng các từ tiếng anh để đặt tên cho các thuộc tính. CSS khi sử dụng có thể viết trực tiếp xen lẫn vào mã HTML hoặc tham chiếu từ một file css riêng biệt. Hiện nay CSS thường được viết riêng thành một tập tin với mở rộng là “.css”. Chính vì vậy mà các trang web có sử dụng CSS thì mã HTML sẽ trở nên ngắn gọn hơn. Ngoài ra có thể sử dụng một tập tin CSS đó cho nhiều website tiết kiệm rất nhiều thời gian và công sức. Một đặc điểm quan trọng đó là tính kế thừa của CSS do đó sẽ giảm được số lượng dòng code mà vẫn đạt được yêu cầu.

Tuy nhiên, đối với CSS thì các trình duyệt hiểu theo kiểu riêng của nó. Do vậy, việc trình bày một nội dung trên các trình duyệt khác nhau là không thống nhất. CSS cung cấp hàng trăm thuộc tính trình bày dành cho các đối tượng với sự sáng tạo cao trong kết hợp các thuộc tính giúp mang lại hiệu quả.

### **1.1.3 Ngôn ngữ lập trình PHP**

PHP (Hypertext Preprocessor) là một ngôn ngữ lập trình kịch bản hay một loại mã lệnh chủ yếu được dùng để phát triển các ứng dụng viết cho máy chủ, mã nguồn mở, dùng cho mục đích tổng quát. Nó rất thích hợp với web và có thể dễ dàng nhúng vào trang HTML. Do được tối ưu hóa cho các ứng dụng web, tốc độ nhanh, nhỏ gọn, cú pháp giống ngôn ngữ lập trình C và ngôn ngữ lập trình Java, dễ học và thời gian xây dựng sản phẩm tương đối ngắn hơn so với các ngôn ngữ khác nên PHP đã nhanh chóng trở thành một ngôn ngữ lập trình web phổ biến nhất thế giới. Một tập tin PHP có phần mở rộng \*.php, nó có thể chứa các văn bản, mã nguồn HTML, CSS, Javascript, JQuery... và đương nhiên có thể chứa mã nguồn PHP.

Đối với các trang PHP khi có yêu cầu xem trang web thì Server sẽ tiến hành phát sinh trang web đó từ mã nguồn PHP sang mã nguồn HTML, sau đó mới chuyển mã nguồn đó về trình duyệt web để người dùng xem. Vì các trình duyệt web không thể đọc được các mã nguồn PHP mà chỉ đọc được các mã nguồn HTML.

### **1.1.4 Framework Laravel**

Laravel là một PHP framework mã nguồn mở và miễn phí, được phát triển bởi Taylor Otwell và nhắm vào mục tiêu hỗ trợ phát triển các ứng dụng web theo kiến trúc model-view-controller (MVC). 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.

Vào khoảng Tháng 3 năm 2015, các lập trình viên đã có một cuộc bình chọn PHP framework phổ biến nhất, Laravel đã giành vị trí quán quân cho PHP framework phổ biến nhất năm 2015, theo sau lần lượt là Symfony2, Nette, CodeIgniter, Yii2 vào một số khác. Trước đó, Tháng 8 2014, Laravel đã trở thành project PHP phổ biến nhất và được theo dõi nhiều nhất trên Github.

Laravel được phát hành theo giấy phép MIT, với source code được lưu trữ tại [Github](#).

Laravel là ứng dụng MVC hoàn chỉnh tương đối dễ tiếp cận phù hợp với người mới lập trình PHP.

Dưới đây là một số tính năng nổi bật của Laravel:

- Viết trên mô hình MVC PHP.
- Đầy đủ các lớp Database giúp hỗ trợ tốt cho các nền tảng khác nhau.
- Dễ dàng đọc hiểu, sử dụng: Các tính năng dựng sẵn: Cung cấp cho người dùng rất nhiều các nhóm tính năng giúp quá trình phát triển trở nên nhanh chóng hơn rất nhiều.
- Các tính năng bảo mật.
- Cung cấp Blade Templade
- Thư viện hỗ trợ lớn.
- Cộng đồng mạnh mẽ

Trong dự án này, chúng em sử dụng phiên bản Laravel 5

### **1.1.5 Mô hình MVC**

MVC (viết tắt của Model-View-Controller) là một mẫu kiến trúc phần mềm để tạo lập giao diện người dùng trên máy tính. MVC chia một ứng dụng thành ba phần tương tác được với nhau để tách biệt giữa cách thức mà thông tin được xử lý nội hàm và phần thông tin được trình bày và tiếp nhận từ phía người dùng.

Khi sử dụng đúng cách, mẫu MVC giúp cho người phát triển phần mềm cô lập các nguyên tắc nghiệp vụ và giao diện người dùng một cách rõ ràng hơn. Phần mềm phát triển theo mẫu MVC tạo nhiều thuận lợi cho việc bảo trì vì các nguyên tắc nghề nghiệp và giao diện ít liên quan với nhau.

Trong mẫu Model-View-Controller, mô hình (model) tượng trưng cho dữ liệu của chương trình phần mềm. Tầm nhìn hay khung nhìn (view) bao gồm các thành phần của giao diện người dùng. Bộ kiểm tra hay bộ điều chỉnh (controller) quản lý sự trao đổi giữa dữ liệu và các nguyên tắc nghề nghiệp trong các thao tác liên quan đến mô hình.

### 1.1.6 Một số thư viện hỗ trợ

#### 1.1.6.1 Thư viện Javascript

JavaScript là một ngôn ngữ dạng script thường được sử dụng cho việc lập trình web ở phía client, nó tuân theo chuẩn ECMAScript. Là một ngôn ngữ linh động, cú pháp dễ sử dụng như các ngôn ngữ khác và dễ dàng lập trình. JavaScript không hề liên quan tới ngôn ngữ lập trình java, được hầu hết các trình duyệt ngày nay hỗ trợ. Với javascript, ứng dụng web của bạn sẽ trở nên vô cùng sinh động, mang tính trực quan và tương tác cao. JavaScript theo phiên bản hiện hành là một ngôn ngữ lập trình kịch bản dựa trên đối tượng được phát triển từ các ý niệm nguyên mẫu. Ngôn ngữ này được dùng rộng rãi cho các trang web, nhưng cũng được dùng để tạo khả năng viết script sử dụng các đối tượng nằm sẵn trong các ứng dụng. Giống Java, JavaScript có cú pháp tương tự ngôn ngữ lập trình C. “.js” là phần mở rộng thường được dùng cho tập tin mã nguồn JavaScript.

#### 1.1.6.2. Thư viện Ajax

AJAX (Asynchronous JavaScript and XML - JavaScript và XML) là một nhóm các công nghệ phát triển web được sử dụng để tạo các ứng dụng web động hay các ứng dụng giàu tính Internet. Ajax là một kỹ thuật phát triển web có tính tương tác cao bằng cách kết hợp các ngôn ngữ: HTML (hoặc XHTML) với CSS trong việc hiển thị thông tin.

Mô hình DOM (Document Object Model) được thực hiện thông qua JavaScript, nhằm hiển thị thông tin động và tương tác với những thông tin được hiển thị.

Đối tượng XMLHttpRequest để trao đổi dữ liệu một cách không đồng bộ với máy chủ web.

XML thường là định dạng cho dữ liệu truyền, mặc dầu bất cứ định dạng nào cũng có thể dùng, bao gồm HTML định dạng trước, văn bản thuần (plain text), JSON và ngay cả EBML.

#### 1.1.6.3. Thư viện Bootstrap

Bootstrap là một Framework có chứa HTML, CSS, JAVASCRIPT, Framework trong tiếng việt có nghĩa là “khuôn khổ” giúp tiết kiệm được thời gian, công sức hơn nữa việc xây dựng hai template cho giao diện Desktop và Mobile đã lỗi thời thay vào đó là Responsive. Responsive sẽ giúp website của bạn hiển thị tương thích với mọi kích thước màn hình nhờ đó bạn sẽ tùy chỉnh hiển thị được nhiều hơn trên các loại màn hình khác nhau.

Ưu điểm của Bootstrap.

- Tiết kiệm thời gian: Bootstrap giúp người thiết kế giao diện website tiết kiệm rất nhiều thời gian. Các thư viện Bootstrap có những đoạn mã sẵn sàng cho bạn áp dụng vào website của mình. Bạn không phải tốn quá nhiều thời gian để tự viết code cho giao diện của mình.
- Tùy biến cao: Bạn hoàn toàn có thể dựa vào Bootstrap và phát triển nền tảng giao diện của chính mình. Bootstrap cung cấp cho bạn hệ thống Grid System mặc định bao gồm 12 cột và độ rộng 940px. Bạn có thể thay đổi, nâng cấp và phát triển dựa trên nền tảng này.
- Responsive Web Design: Với Bootstrap, việc phát triển giao diện website để phù hợp với đa thiết bị trở nên dễ dàng hơn bao giờ hết. Đây là xu hướng phát triển giao diện website đang rất được ưu chuộng trên thế giới.

#### 1.1.6.4.Thư viện JQUERY

JQuery là một thư viện kiểu mới của JavaScript, được tạo bởi John Resig vào năm 2006 với một phương châm tuyệt vời: Write less, do more - Viết ít hơn, làm nhiều hơn. JQuery làm đơn giản hóa việc truyền tải HTML, xử lý sự kiện, tạo hiệu ứng động và tương tác Ajax. Với jQuery, khái niệm Rapid Web Development đã không còn quá xa lạ.

JQuery là một bộ công cụ tiện ích JavaScript làm đơn giản hóa các tác vụ đa dạng với việc viết ít code hơn. Dưới đây liệt kê một số tính năng tối quan trọng được hỗ trợ bởi jQuery:

Thao tác DOM – jQuery giúp dễ dàng lựa chọn các phần tử DOM để *traverse* (*duyệt*) một cách dễ dàng như sử dụng CSS, và chỉnh sửa nội dung của chúng bởi sử dụng phương tiện Selector mã nguồn mở, mà được gọi là Sizzle.

- Xử lý sự kiện – jQuery giúp tương tác với người dùng tốt hơn bằng việc xử lý các sự kiện đa dạng mà không làm cho HTML code rối tung lên với các Event Handler.
- Hỗ trợ AJAX – jQuery giúp bạn rất nhiều để phát triển một site giàu tính năng và phản hồi tốt bởi sử dụng công nghệ AJAX.
- Hiệu ứng – jQuery đi kèm với rất nhiều các hiệu ứng đa dạng và đẹp mắt mà bạn có thể sử dụng trong các Website của mình.

- Gọn nhẹ – jQuery là thư viện gọn nhẹ - nó chỉ có kích cỡ khoảng 19KB (gzipped).
- Được hỗ trợ hầu hết bởi các trình duyệt hiện đại – jQuery được hỗ trợ hầu hết bởi các trình duyệt hiện đại, và làm việc tốt trên IE 6.0+, FF 2.0+, Safari 3.0+, Chrome và Opera 9.0+.
- Cập nhật và hỗ trợ các công nghệ mới nhất – jQuery hỗ trợ CSS3 Selector và cú pháp XPath cơ bản.

## 2. Các thành phần cơ bản của LARAVEL

```
lar-53/
|--- app/
|   |--- Console
|   |--- Exceptions
|   |--- Http
|   |   |--- Controllers/
|   |       |---Auth
|   |       |---Controller.php
|   |   |--- Middleware
|   |       |--- Kernel.php
|   |--- Providers
|   |--- User.php
|--- bootstrap/
|--- config/
|--- database/
|--- public/
|   |--- ...
|   |--- index.php
|--- resources/
|   |--- assets
|   |--- lang
|   |--- views
|--- routes/
|   |--- api.php
|   |--- console.php
|   |--- web.php
|--- storage/
|--- tests/
|--- vendor/
|--- .env
|--- .env.example
|--- .gitattributes
|--- .gitignore
|--- artisan
|--- composer.json
|--- composer.lock
|--- gulpfile.js
|--- package.json
|--- phpunit.xml
|--- readme.md
|--- server.php
```



## Hình ảnh 2

### **Chú thích:**

app: Chứa các thư mục, các tập tin php, thư viện, models.

Console: Chứa các tập tin định nghĩa các câu lệnh trên artisan.

Excerpt: Chứa các tập tin quản lý, điều hướng lỗi.

Http

Controllers : Chứa các controllers của project.

Middleware: Chứa các tập tin lọc và ngăn chặn các requests.

Kernel.php: Cấu hình, định nghĩa Middleware.

Providers: Chứa các providers thực hiện việc binding vào service container(ở phần nâng cao mình sẽ nói sau).

User.php: Là model User mà Laravel tự định sẵn cho chúng ta.

bootstrap: Chứa tập tin điều hướng hệ thống.

config: Chứa mọi tập tin cấu hình của Laravel.

database: Chứa các thư mục tập tin về database.

migrations: Chứa các tập tin định nghĩa, khởi tạo và sửa bảng.

seeds: Chứa các tập tin định nghĩa dữ liệu insert(thêm) vào trong database.

factories: Chứa các tập tin định nghĩa các cột bảng dữ liệu để tạo ra các dữ liệu ảo.

public: Chứa các tập tin css, js, image.

index.php: Đây là tập tin root của Laravel

resources: Chứa các views, ngôn ngữ(language) của project.

routes: Chứa các tập tin định nghĩa các router, xử lý điều hướng router bao gồm : web, api và console(Mình sẽ nói thêm ở phần sau).

storage: Chứa các tập tin hệ thống cache, session, ...

tests: ...

vendor: Chứa các thư viện của composer.

.env: Là tập tin cấu hình chính của laravel như key app,database.

.env.example: Tập tin cấu hình mẫu của laravel.

composer.json: tập tin của composer.

composer.lock: tập tin của composer.

package.js: Tập tin cấu hình của nodejs (chứa các package cần dùng cho projects).

gulpfile.js: Là tập tin gulp builder.

phpunit.xml: Là tập tin xml của phpunit dùng để testing project.

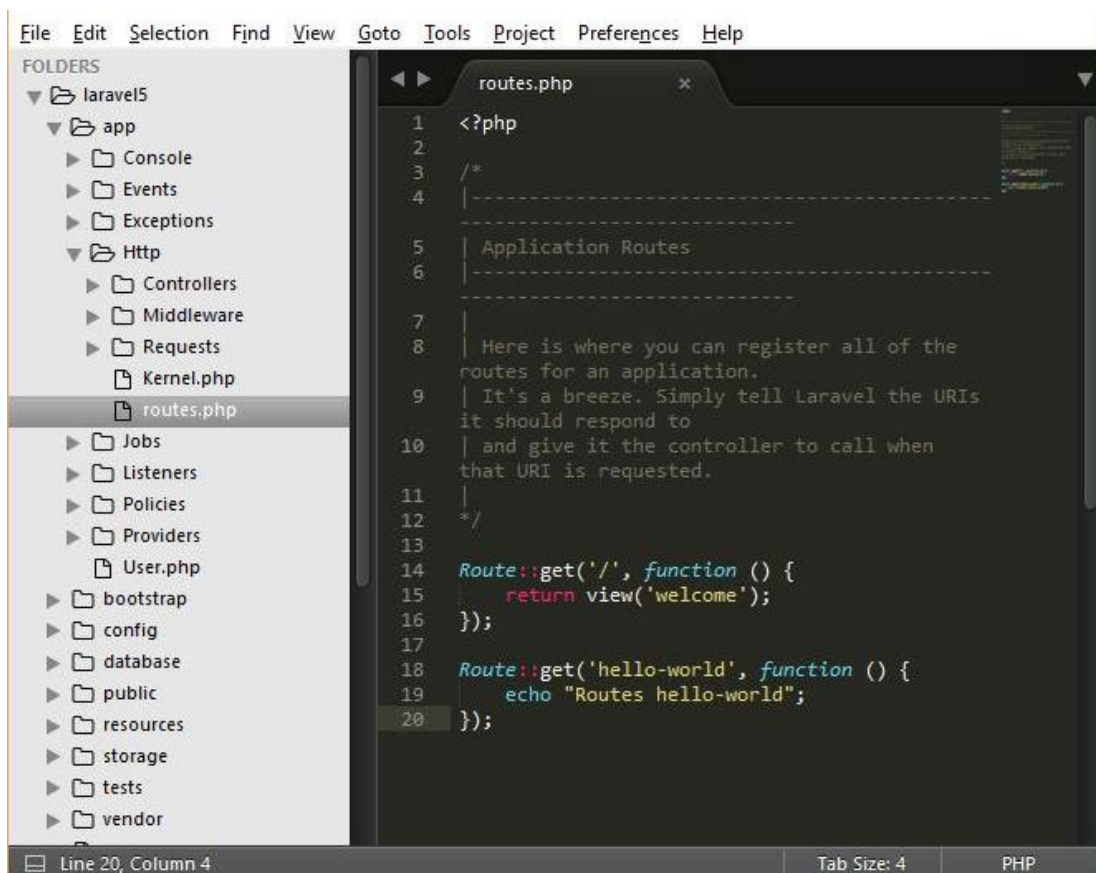
server.php: Là tập tin để artisan trở đến tạo server khi gõ lệnh php artisan serve .

artisan: Tập tin thực thi lệnh của Laravel.

## 2.1 Route

### 2.1.1 Chức năng: điều hướng các request của người dùng.

Mở file routers.php để viết điều hướng



Hình ảnh 1 :

### 2.1.2 các loại route trong Larvel

-Laravel là một framework hỗ trợ chúng ta rất đa dạng về route, và cũng rất dễ sử dụng. Nó gồm có các loại như sau:

Route::get -> nhận resquest với phương thức GET.

Route::post -> nhận resquest với phương thức POST.

Route::put -> nhận resquest với phương thức PUT.

Route::delete -> nhận request với phương thức DELETE.

Route::match -> kết hợp nhiều phương thức như POST,GET,PUT,..

Route::any -> nhận tất cả các phương thức.

Route::group -> tạo ra các nhóm route.

Route::controller -> gọi đến controller tương ứng mà chúng ta tự định.

Route::resource -> sử dụng với resource controller.

### 2.1.3 Sử dụng route phổ biến trong Laravel.

#### 2.1.3.1 Route::get

-Đây là route sẽ tiếp nhận các request với phương thức GET.

-Cú Pháp:

```
Route::get($url, $action);
```

Trong đó: \$url là đường dẫn của route, \$action là các hành động thực hiện trong router.

#### 2.1.3.2 Route::post

-Đối với loại route này thì nó không khác gì với Route::get() ngoài : Route::get() nhận phương thức GET còn Route::post() nhận phương thức POST.

-Cú Pháp:

```
Route::post($url, $action);
```

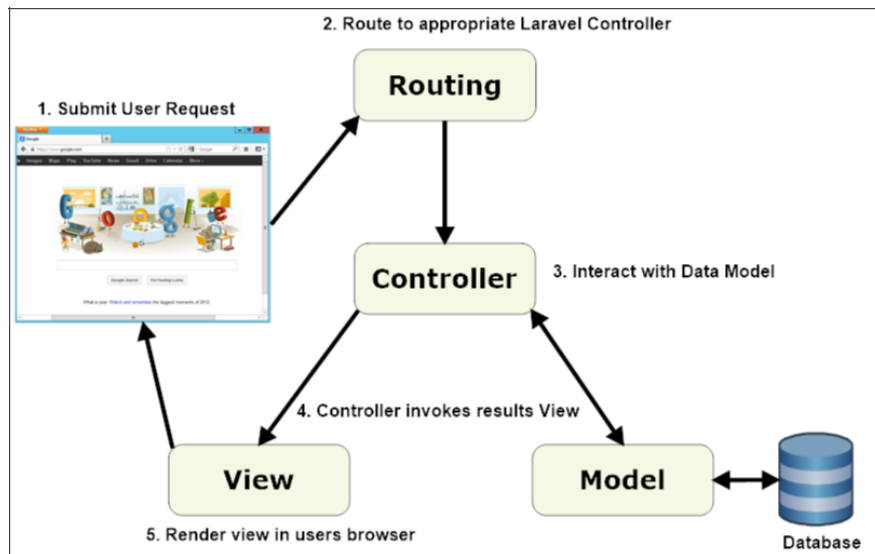
#### 2.1.3.3 Route::match

-Nếu muốn sử dụng hai phương thức GET,POST mà không muốn dùng phương thức PUT,PATCH,DELETE,... Laravel đã cho chúng ta thêm một route nữa là match route.

-Cú Pháp:

```
Route::match('phuong thuc', 'url', 'action');
```

## 2.2 Cấu trúc MVC trong Laravel



**Hình ảnh 3:**

Sau **route** thì một phần rất quan trọng khác được sử dụng rất thường xuyên trong Laravel Framework là mô hình tổ chức source code MVC. Mô hình MVC là một kiểu kiến trúc giúp cho việc tổ chức source code trở lên có hệ thống. Ở đây, mọi **Request** từ phía người dùng đều phải qua **Route**, dữ liệu được gửi xuống **Controller** để xử lý, cần dữ liệu sẽ lấy từ **Model** lên hoặc cập nhật dữ liệu xuống **Model**, kết quả gửi ra **View** cho người sử dụng.

## 2.3 Cacs command line sử dụng trong laravel 5

Lệnh	Chức năng	Ghi chú
composer create-project laravel/laravel <Tên project> --prefer-dist	Khởi tạo project web với Laravel 5.0	Cần cd tới thư mục chứa project trước khi chạy lệnh tạo project.
php artisan make:model <Tên Bảng> --migration	Tạo model và migrate	
php artisan migrate	Cập nhật migrate để thêm bảng vào database sau khi tạo migrate và thêm cột trong file	

				create_teenbang_table	
php	artisan	make:controller	<Tên Controller>	Tạo Controller để xử lý	

## 2. GIỚI THIỆU VỀ LẬP TRÌNH ANDROID

### 2.1. Khái quát về android

#### 2.1.1 Lịch sử Android

Ban đầu, Android là hệ điều hành cho các thiết bị cầm tay dựa trên lõi Linux do công ty Android Inc. (California, Mỹ) thiết kế. Công ty này sau đó được Google mua lại vào năm 2005 và bắt đầu xây dựng Android Platform. Các thành viên chủ chốt ở Android Inc. gồm có: Andy Rubin, Rich Miner, Nick Sears, and Chris White.

Và sau tiếp, vào cuối năm 2007, thuộc về Liên minh Thiết bị Cầm tay Mã Nguồn mở (Open Handset Alliance) gồm các thành viên nổi bật trong ngành viễn thông và thiết bị cầm tay như:

Texas Instruments, Broadcom Corporation, Google, HTC, Intel, LG, Marvell Technology Group, Motorola, Nvidia, Qualcomm, Samsung Electronics, Sprint Nextel, T-Mobile, ARM Holdings, Atheros Communications, Asustek Computer Inc, Garmin Ltd, Softbank, Sony Ericsson, Toshiba Corp, and Vodafone Group,...

Mục tiêu của Liên minh này là nhanh chóng đổi mới để đáp ứng tốt hơn cho nhu cầu người tiêu dùng và kết quả đầu tiên của nó chính là nền tảng Android. Android được thiết kế để phục vụ nhu cầu của các nhà sản xuất thiết, các nhà khai thác và các lập trình viên thiết bị cầm tay.

Phiên bản SDK lần đầu tiên phát hành vào tháng 11 năm 2007, hãng T-Mobile cũng công bố chiếc điện thoại Android đầu tiên đó là chiếc T-Mobile G1, chiếc smartphone đầu tiên dựa trên nền tảng Android. Một vài ngày sau đó, Google lại tiếp

tục công bố sự ra mắt phiên bản Android SDK release Candidate 1.0. Trong tháng 10 năm 2008, Google được cấp giấy phép mã nguồn mở cho Android Platform.

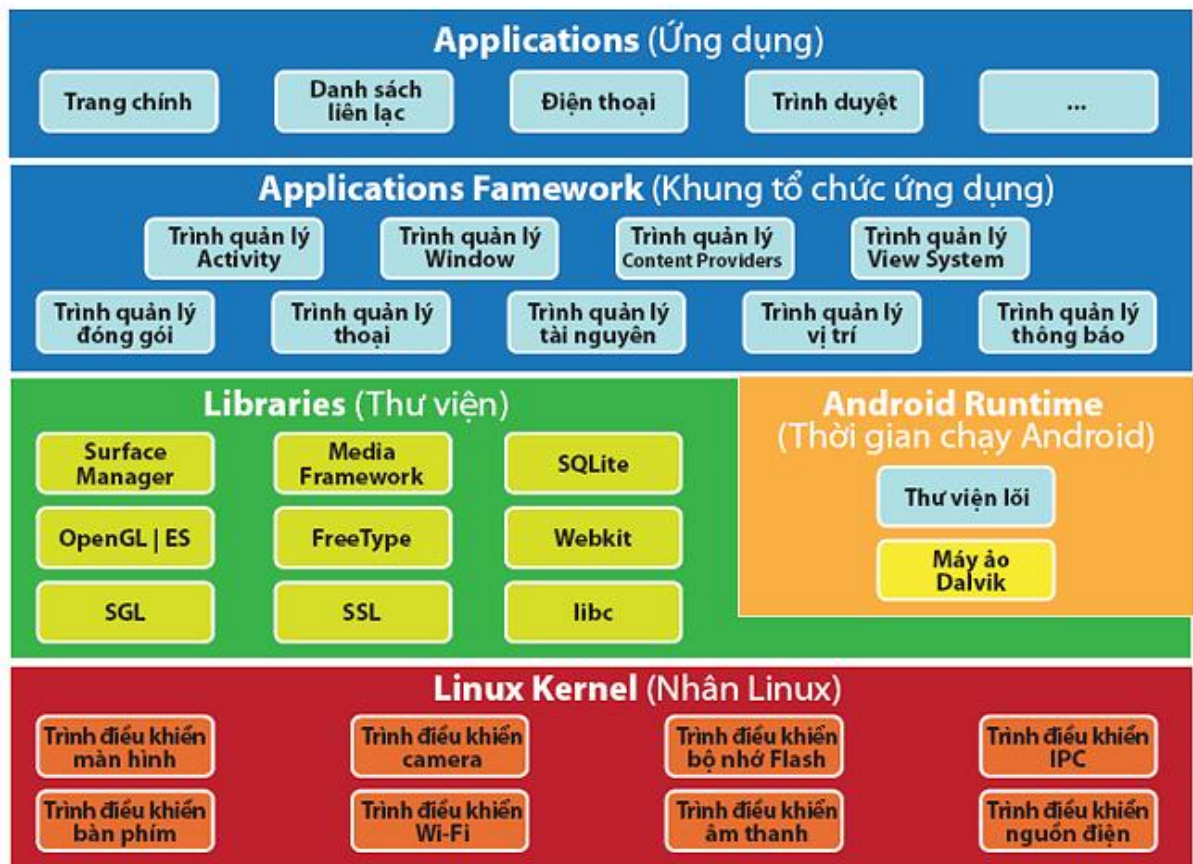
Khi Android được phát hành thì một trong số các mục tiêu trong kiến trúc của nó là cho phép các ứng dụng có thể tương tác được với nhau và có thể sử dụng lại các thành phần từ những ứng dụng khác. Việc tái sử dụng không chỉ được áp dụng cho các dịch vụ mà nó còn được áp dụng cho cả các thành phần dữ liệu và giao diện người dùng.

Vào cuối năm 2008, Google cho phát hành một thiết bị cầm tay được gọi là Android Dev Phone 1 có thể chạy được các ứng dụng Android mà không bị ràng buộc vào các nhà cung cấp mạng điện thoại di động. Mục tiêu của thiết bị này là cho phép các nhà phát triển thực hiện các cuộc thí nghiệm trên một thiết bị thực có thể chạy hệ điều hành Android mà không phải ký một bản hợp đồng nào. Vào khoảng cùng thời gian đó thì Google cũng cho phát hành một phiên bản vá lỗi 1.1 của hệ điều hành này. Ở cả hai phiên bản 1.0 và 1.1 Android chưa hỗ trợ soft-keyboard mà đòi hỏi các thiết bị phải sử dụng bàn phím vật lý. Android cố định vấn đề này bằng cách phát hành SDK 1.5 vào tháng Tư năm 2009, cùng với một số tính năng khác. Chẳng hạn như nâng cao khả năng ghi âm truyền thông, vật dụng, và các live folder..

Android đã đi một chặng đường dài từ khởi đầu khiêm tốn để trở thành hệ điều hành di động hàng đầu thế giới. Nhiều tin đồn cho rằng Google đang ở giai đoạn rất sớm để phát triển một hệ điều hành hoàn toàn mới, được gọi là Fuchsia, có thể hỗ trợ mọi thứ từ điện thoại thông minh đến máy tính bảng, thậm chí cả máy tính xách tay và máy tính để bàn. Tháng 5/2017, tại Google I / O, công ty cho biết hiện có hơn 2 tỷ thiết bị đang hoạt động chạy một số phiên bản của hệ điều hành Android. Tính linh hoạt, kết hợp với các cập nhật hàng năm, sẽ đảm bảo Android sẽ vẫn là nhà lãnh đạo trong ngành công nghiệp này trong nhiều năm tới.

## 2.1.2 KIẾN TRÚC CỦA ANDROID

Mô hình sau thể hiện một cách tổng quát các thành phần của hệ điều hành Android. Mỗi một phần sẽ được đặc tả một cách chi tiết dưới đây.



Bảng gì đó sẽ sửa sau

- Linux kernel

Ở dưới cùng của các tầng trên là Linux - Linux 2.6 với khoảng 115 bản vá lỗi. Tầng này cung cấp chức năng hệ thống cơ bản như quản lý các tiến trình, quản lý bộ nhớ, quản lý thiết bị như máy ảnh, bàn phím, màn hình hiển thị, v.v... Ngoài ra, nhân Linux xử lý tất cả các vấn đề về Mạng và một loạt các trình điều khiển thiết bị giao tiếp với phần cứng ngoại vi.

- Libraries

Tầng này là một tập hợp các thư viện bao gồm trình duyệt web mã nguồn mở sử dụng WebKit engine, thư viện libc, cơ sở dữ liệu SQLite là một kho lưu trữ hữu ích cho việc lưu trữ và chia sẻ dữ liệu của ứng dụng, có các thư viện để chơi/ghi lại âm thanh và video, các thư viện SSL chịu trách nhiệm về bảo mật Internet, v.v...

- Android Runtime

Đây là phần thứ ba của kiến trúc Android. Phần này cung cấp một thành phần quan trọng được gọi là Dalvik Virtual Machine - một loại máy ảo Java được thiết kế đặc biệt và tối ưu hóa cho Android.

Dalvik VM sử dụng các tính năng cốt lõi của Linux như quản lý bộ nhớ và đa luồng, những tính năng này đều có trong ngôn ngữ Java. Dalvik VM cho phép tất cả các ứng dụng Android chạy trong tiến trình riêng của nó.

Android Runtime cũng cung cấp một tập các thư viện lõi cho phép các nhà phát triển ứng dụng Android viết các ứng dụng Android sử dụng ngôn ngữ lập trình Java (J2SE).

- Application Framework

Tầng Application Framework cung cấp nhiều dịch vụ cấp cao hơn cho các ứng dụng trong ở dạng các class trong Java. Các nhà phát triển ứng dụng được phép sử dụng các dịch vụ này trong các ứng dụng của họ. (Ví dụ: dịch vụ kết nối Internet, dịch vụ SMS,...)

- Applications

Lập trình viết ứng dụng và các ứng dụng được cài đặt ở tầng này. Ví dụ: Mario, Facebook, Zalo, v.v...

### 2.1.3 Application framework

Bằng cách cung cấp một nền tảng phát triển mở, Android cung cấp cho các nhà phát triển khả năng xây dựng các ứng dụng cực kỳ phong phú và sáng tạo. Nhà phát triển được tự do tận dụng các thiết bị phần cứng, thông tin địa điểm truy cập, các dịch vụ chạy nền, thiết lập hệ thống báo động, thêm các thông báo để các thanh trạng thái, và nhiều, nhiều hơn nữa.

Nhà phát triển có thể truy cập vào các API cùng một khuôn khổ được sử dụng bởi các ứng dụng lõi. Các kiến trúc ứng dụng được thiết kế để đơn giản hóa việc sử dụng lại các thành phần; bất kỳ ứng dụng có thể xuất bản khả năng của mình và ứng dụng nào khác sau đó có thể sử dụng những khả năng (có thể hạn chế bảo mật được thực thi bởi khuôn khổ). Cơ chế này cho phép các thành phần tương tự sẽ được thay thế bởi người sử dụng.

Cơ bản tất cả các ứng dụng là một bộ các dịch vụ và các hệ thống, bao gồm:



Một tập hợp rất nhiều các View có khả năng kế thừa lẫn nhau dùng để thiết kế phân giao diện ứng dụng như: gridview, tableview, linearlayout,...

Một “Content Provider” cho phép các ứng dụng có thể truy xuất dữ liệu từ các ứng dụng khác (chẳng hạn như Contacts) hoặc là chia sẻ dữ liệu giữa các ứng dụng đó.

Một “Resource Manager” cung cấp truy xuất tới các tài nguyên không phải là mã nguồn, chẳng hạn như: localized strings, graphics, and layout files.

Một “Notifcation Manager” cho phép tất cả các ứng dụng hiển thị các custom alerts trong status bar.

Activity Manager được dùng để quản lý chu trình sống của ứng dụng và điều hướng các activity.

#### 2.1.4 Library

Android bao gồm một tập hợp các thư viện C/C++ được sử dụng bởi nhiều thành phần khác nhau trong hệ thống Android. Điều này được thể hiện thông qua nền tảng ứng dụng Android. Một số các thư viện cơ bản được liệt kê dưới đây:

- **System C library**: a BSD-derived implementation of the standard C system library (libc), tuned for embedded Linux-based devices.
- **Media Libraries** - based on PacketVideo's OpenCORE; the libraries support playback and recording of many popular audio and video formats, as well as static image files, including MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG.
- **Surface Manager** – Quản lý việc truy xuất vào hệ thống hiển thị.
- **LibWebCore** - a modern web browser engine which powers both the Android browser and an embeddable web view.
- **SGL** - the underlying 2D graphics engine.
- **3D libraries** - an implementation based on OpenGL ES 1.0 APIs; the libraries use either hardware 3D acceleration (where available) or the included, highly optimized 3D software rasterizer.
- **FreeType** - bitmap and vector font rendering.

#### 2.1.5 ANDROID EMULATOR

Android SDK và Plugin Eclipse được gọi là một Android Developer Tool (ADT). Các Android coder sẽ cần phải sử dụng công cụ IDE (Integrated Development Environment) này để phát triển, debugging và testing cho ứng dụng. Tuy nhiên, các

coder cũng có thể không cần phải sử dụng IDE mà thay vào đó là sử dụng command line để biên dịch và tất nhiên là vẫn có Emulator như thường.

Android Emulator được trang bị đầy đủ hầu hết các tính năng của một thiết bị thật. Tuy nhiên, một số đã bị giới hạn như là kết nối qua cổng USB, camera và video, nghe phone, nguồn điện giả lập và bluetooth.

Android Emulator thực hiện các công việc thông qua một bộ xử lý mã nguồn mở, công nghệ này được gọi là QEMU (<http://bellard.org/qemu/>) được phát triển bởi Fabrice Bellard.

## **2.2 Mô hình thiết kế Model View Controller (MVC)**

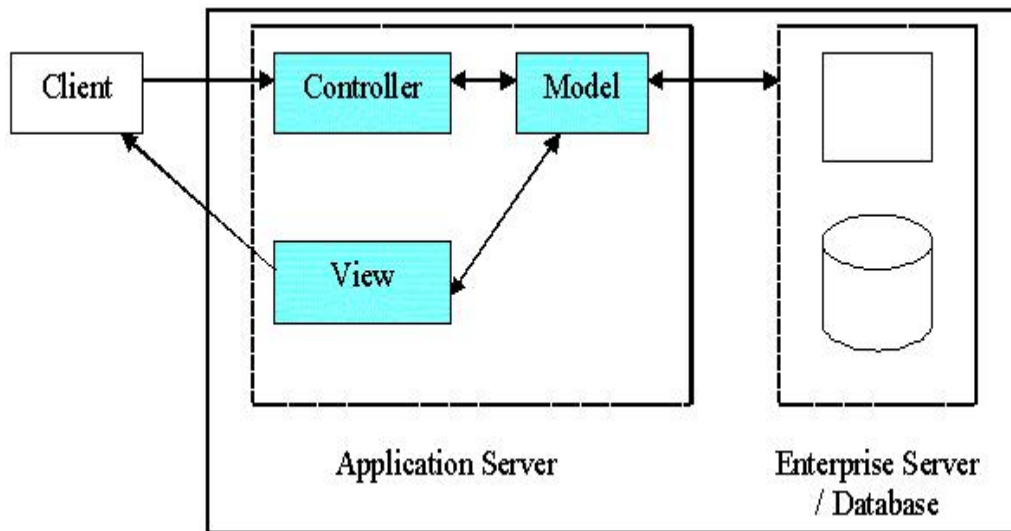
### **2.2.1 Model View Controller (MVC) là gì?**

MVC là tên một phương pháp chia nhỏ một ứng dụng thành ba thành phần để cài đặt, mỗi thành phần đóng một vai trò khác nhau và ảnh hưởng lẫn nhau, đó là models, views, và controllers.

- Model: Models trong các ứng dụng dựa trên MVC là những thành phần có nhiệm vụ lưu trữ thông tin, trạng thái của các đối tượng, thông thường nó là một lớp được ánh xạ từ một bảng trong CSDL. Lấy ví dụ, chúng ta có lớp Product được sử dụng để mô tả dữ liệu từ bảng Products trong SQL, bao gồm ProductID, OrderDate...

- Views: Views, nó chính là các thành phần chịu trách nhiệm hiển thị các thông tin lên cho người dùng thông qua giao diện. Thông thường, các thông tin cần hiển thị được lấy từ thành phần Models.

- Controllers : Controllers trong các ứng dụng kiểu MVC chịu trách nhiệm xử lý các tác động về mặt giao diện, các thao tác đối với models, và cuối cùng là chọn một view thích hợp để hiển thị ra màn hình. Trong kiến trúc MVC, view chỉ có tác dụng hiển thị giao diện mà thôi, còn điều khiển dòng nhập xuất của người dùng vẫn do Controllers đảm trách



Hình 5

- Khi người sử dụng ra một lệnh, lệnh này sẽ được gửi tới phần Controller.
- Phần Điều khiển sẽ khởi tạo phần Model (nếu cần thiết), gửi các yêu cầu tới phần Model để thực hiện.
- Căn cứ trên các lệnh và thông tin nhận được từ lệnh, phần Model sẽ đảm nhận việc lấy thông tin và cập nhật thông tin trong các hệ thống khác, ví dụ trong các Enterprise Server như Application Server, Mail Server, Database Server, từ hệ thống .

Sau khi hoàn thành việc thu thập, cập nhật thông tin, Model sẽ truyền những thông tin cần thiết về phần Controller.

Lúc này, phần Controller sẽ quyết định chọn thành phần nào trong phần View để hiện dữ liệu ra cho người dùng.

Phần View khi làm nhiệm vụ hiện thông tin cho người dùng cũng có thể truy cập các thông tin hiển thị từ Model , hoặc gửi thông tin hiển thị tới Model. Trường hợp này xảy ra khi Model chứa các thông tin có thể dùng để hiện trực tiếp, ví dụ một danh sách khách hàng, hoặc một danh sách các e-mail trong mailbox. Khi phần View hiện thông tin, nó có thể báo cho phần Model biết nó đang hiện phần nào của thông tin