

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----



ISO 9001:2015

ĐỒ ÁN TỐT NGHIỆP

NGÀNH CÔNG NGHỆ THÔNG TIN

Hải phòng

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----

**XÂY DỰNG HỆ THỐNG QUẢN LÝ BẢO HÀNH
VÀ SỬA CHỮA VẬT TƯ**

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

-----o0o-----

**XÂY DỰNG HỆ THỐNG QUẢN LÝ BẢO HÀNH
VÀ SỬA CHỮA VẬT TƯ**

ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY

Ngành: Công Nghệ Thông Tin

BỘ GIÁO DỤC VÀ ĐÀO TẠO

Cộng hòa xã hội chủ nghĩa Việt Nam

TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG

Độc lập - Tự do - Hạnh phúc

-----oOo-----

NHIỆM VỤ THIẾT KẾ TỐT NGHIỆP

Sinh viên: Trần Phú Thế

Mã số: 1412101014

Lớp: CT1802

Ngành: Công Nghệ Thông Tin

Tên đề tài: Xây dựng hệ thống quản lý bảo hành và sửa chữa Vật Tư.

NHIỆM VỤ ĐỀ TÀI

1. Nội dung và các yêu cầu cần giải quyết trong nhiệm vụ đề tài tốt nghiệp

a. Nội dung:

b. Các yêu cầu cần giải quyết

2. Các số liệu cần thiết để thiết kế, tính toán

3. Địa điểm thực tập

CÁN BỘ HƯỚNG DẪN ĐỀ TÀI TỐT NGHIỆP

Người hướng dẫn thứ nhất :

Họ và tên : Đỗ Văn Chiêu

Học hàm, học vị: Tiến sĩ

Cơ quan công tác: Khoa công nghệ thông tin Trường đại học dân lập Hải Phòng

Nội dung hướng dẫn:

.....
.....
.....

Người hướng dẫn thứ hai :

Họ và tên :.....

Học hàm, học vị:.....

Cơ quan công tác:

Nội dung hướng dẫn:

.....
.....
.....

Đề tài tốt nghiệp được giao ngày 11 tháng 6 năm 2018

Yêu cầu hoàn thành trước ngày 31 tháng 8 năm 2018

Đã nhận nhiệm vụ : Đ.T.T.N

Sinh viên

Đã nhận nhiệm vụ : Đ.T.T.N

Cán bộ hướng dẫn : Đ.T.T.N

Hải phòng, ngàytháng.....năm 2018

HIỆU TRƯỞNG

GS.TS Trần Hữu Nghị

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN TỐT NGHIỆP

Họ và tên giảng viên:

Đơn vị công tác:

Họ và tên sinh viên: Ngành:

Nội dung hướng dẫn:

.....

1. Tinh thần thái độ của sinh viên trong quá trình làm đề tài tốt nghiệp

.....
.....
.....
.....

2. Đánh giá chất lượng của đề án/khóa luận (so với nội dung yêu cầu đã đề ra trong nhiệm vụ Đ.T. T.N trên các mặt lý luận, thực tiễn, tính toán số liệu...)

.....
.....
.....
.....

3. Ý kiến của giảng viên hướng dẫn tốt nghiệp

Đạt Không đạt Điểm:.....

Hải Phòng, ngày tháng 8 năm 2018

Giảng viên hướng dẫn

(Ký và ghi rõ họ tên)

CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM

Độc lập - Tự do - Hạnh phúc

PHIẾU NHẬN XÉT CỦA GIẢNG VIÊN CHẤM PHẢN BIỆN

Họ và tên giảng viên:

Đơn vị công tác:

Họ và tên sinh viên: Ngành:

Đề tài tốt nghiệp:

.....

1. Phần nhận xét của giảng viên chấm phản biện

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

....

2. Những mặt còn hạn chế

.....
.....
.....
.....
.....
.....
.....

3. Ý kiến của giảng viên chấm phản biện

Được bảo vệ Không được bảo vệ Điểm:.....

Hải Phòng, ngày tháng 8 năm 2018

Giảng viên chấm phản biện

(Ký và ghi rõ họ tên)

MỤC LỤC

GIỚI THIỆU	4
1.1 Giới thiệu về công ty CETI	4
<i>Giới thiệu về bảo hành sửa chữa</i>	4
<i>Giới thiệu về hệ thống bảo hành sửa chữa điện tử</i>	5
1. Dịch vụ web	6
2. Các công nghệ xây dựng dịch vụ Web	6
1.2 Giới thiệu về ngôn ngữ MVC5 và SQL	17
1. Ngôn ngữ MVC5	17
2. Hệ quản trị CSDL SQL	17
CHƯƠNG 2: CỞ SỞ LÝ THUYẾT	18
2.1 Web service và chức năng	18
2.2 Xây dựng 1 số lớp giao diện	18
CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG	23
3.1 Phân Tích Hệ Thông về dữ liệu	Error! Bookmark not defined.
3.2 Phân Tích Thiết Kế Hệ Thông Website	23
CHƯƠNG 4: CHƯƠNG TRÌNH THỰC NGHIỆM	31
4.1 Thiết kế cơ sở dữ liệu	31
.....	31
4.2 Một số giao diện	33
KẾT LUẬN	40
TÀI LIỆU THAM KHẢO	41

GIỚI THIỆU

1.1 Giới thiệu về công ty CETI

Tiền thân là những sinh viên năng động, luôn khao khát mang lại lợi ích và những giá trị cao nhất cho Sinh viên mới ra trường, mình đã khởi nghiệp bán điện thoại từ khi là sinh viên đại học năm thứ 4. Khó khăn, gian nan rất nhiều với sinh viên còn non nớt kinh nghiệm cả về tuổi đời lẫn xã hội.

CETI chính thức được thành lập từ năm 2012, chuyên kinh doanh điện thoại di động, đồ chơi công nghệ, phụ kiện điện thoại, sửa chữa điện thoại di động tại thành phố Hải Phòng.

Với về dày kinh nghiệm và uy tín được đã tạo dựng được trong những năm vừa qua, Cửa hàng điện thoại Di Động Sinh Viên là một trong những địa chỉ tin cậy của người yêu Công Nghệ tại Hải Phòng nói riêng và trên toàn quốc nói chung. Mình xin cam kết 100% đem lại cho khách hàng sự hài lòng về chất lượng sản phẩm và dịch vụ của Shop.

Giới thiệu về bảo hành sửa chữa

Hằng ngày khách hàng luôn phải làm việc với các thiết bị di động của mình, đôi khi thiết bị của họ sẽ bị gặp một số các trục trặc không mong muốn và đến cửa hàng với nhu cầu sửa chữa khắc phục lỗi của các thiết bị trên. Công ty sẽ nhận thông tin sản phẩm của khách hàng. Sau khi nhận được hàng sẽ tiến kiểm tra thông tin thiết bị xem có còn bảo hành hay không và tiến hành sửa chữa cho khách hàng.

Lợi ích của việc sử dụng quản lý trên hệ thống với quản lý trên giấy tờ :

Giảm thiểu rủi ro khi lưu trữ thông tin : Không còn nỗi lo mất, cháy, hỏng, bị mờ hoặc bay mất thông tin và bảo quản thông tin cho doanh nghiệp.

Khởi tạo, nhập thông tin nhanh chóng: Không cần viết tay, hóa đơn điện tử giúp kế toán viên tránh tình trạng viết nhầm, viết sai. Không phải xử lý các trường hợp viết sai hóa đơn.

An toàn thông tin, dễ dàng tra cứu và tìm kiếm : kế toán không cần phải mất hàng giờ vào kho lưu trữ để tìm kiếm hóa đơn gốc. Thông tin hóa đơn được bảo mật.

Giảm 50% công việc của kế toán: bằng cách tích hợp dữ liệu đầu vào với các phần mềm kế toán như: MÍA, FAST.....nhanh chóng lên được báo cáo sổ sách thay vì phải ngồi nhập dữ liệu như trước đây.

Không cần phải lập báo cáo tình hình sử dụng hóa đơn: bởi tất cả thông tin hóa đơn đã được gửi lên và lưu trữ trên Tổng cục Thuế khi xác thực. Giảm chi phí nhân công giải quyết các công việc báo cáo, thực hiện các thủ tục hành chính liên quan tới hóa đơn, thuế.

Giới thiệu về hệ thống bảo hành sửa chữa điện tử

Các điều kiện Công ty đã đảm bảo khi triển khai :

- Hạ tầng về công nghệ thông tin để lưu trữ và sử dụng chứng từ điện tử và phải có chữ ký điện tử.
- Sử dụng phần mềm hoá đơn điện tử tự xây dựng.
- Ban hành Quyết định áp dụng hoá đơn điện tử và gửi Thông báo phát hành hoá đơn điện tử tới cơ quan thuế.
- Thông báo cho khách hàng về định dạng hoá đơn và cách thức truyền nhận hoá đơn điện tử.

Ưu điểm :

- Giảm thiểu các thủ tục hành chính, lưu trữ. Tiết kiệm thời gian thực hiện phân tích, báo cáo. Có thể gửi hóa đơn cho Khách hàng ngay sau khi phát hành qua nhiều hình thức : Email, SMS, Website. Tiết kiệm thời gian tìm kiếm.
- Giảm chi phí in ấn, gửi nhận hóa đơn cho khách hàng. Không mất chi phí lưu trữ, bảo quản, không thất lạc, rách, hỏng, cháy. Không thể làm giả hóa đơn.
- Dịch vụ hiện đại, nhanh chóng nhận hóa đơn, thuận tiện tra cứu

CHƯƠNG I : GIỚI THIỆU VỀ NGÔN NGỮ MVC5

1. Dịch vụ web

Khái niệm

Dịch vụ web (Web Service) là sự kết hợp các ứng dụng trên máy tính cá nhân, thiết bị di động với ứng dụng trên các thiết bị khác, các cơ sở dữ liệu và các mạng máy tính để tạo thành một cơ cấu tính toán hoàn hảo mà người sử dụng có thể làm việc, yêu cầu, phân tích, khai thác, cập nhật,... thông tin với nó thông qua mạng Internet hoặc Wifi. Đồng thời nó cũng xuất bản các chức năng của mình để mọi người dùng internet trên thế giới đều có thể sử dụng thông qua nền tảng Web.

Web Service truyền thông bằng cách sử dụng các giao thức mở, tài nguyên phần mềm có thể xác định bằng địa chỉ URL, thực hiện các chức năng và đưa ra các thông tin người dùng yêu cầu, các ứng dụng độc lập và tự mô tả chính nó. Nó bao gồm các modul độc lập cho hoạt động của khách hàng và doanh nghiệp và bản thân nó được thực thi trên server. Nền tảng cơ bản của Web Service . Bất cứ một ứng dụng nào cũng đều có thể có một thành phần Web Service. Web Service có thể được tạo ra bằng bất kỳ một ngôn ngữ lập trình nào.

Dịch vụ Web (Web Service) được coi là một công nghệ mang đến cuộc cách mạng trong cách thức hoạt động của các dịch vụ B2B (Business to Business) và B2C (Business to Customer). Giá trị cơ bản của dịch vụ Web dựa trên việc cung cấp các phương thức theo chuẩn trong việc truy nhập đối với hệ thống đóng gói và hệ thống kế thừa. Các phần mềm được viết bởi những ngôn ngữ lập trình khác nhau và chạy trên những nền tảng khác nhau có thể sử dụng dịch vụ Web để chuyển đổi dữ liệu thông qua mạng Internet theo cách giao tiếp tương tự bên trong một máy tính. Tuy nhiên, công nghệ xây dựng dịch vụ Web không nhất thiết phải là các công nghệ mới, nó có thể kết hợp với các công nghệ đã có như XML, SOAP, WSDL, UDDI... Với sự phát triển và lớn mạnh của Internet, dịch vụ Web thật sự là một công nghệ đáng được quan tâm để giảm chi phí và độ phức tạp trong tích hợp và phát triển hệ thống. Chúng ta sẽ xem xét các dịch vụ Web từ mức khái niệm đến cách thức xây dựng.

2. Các công nghệ xây dựng dịch vụ Web

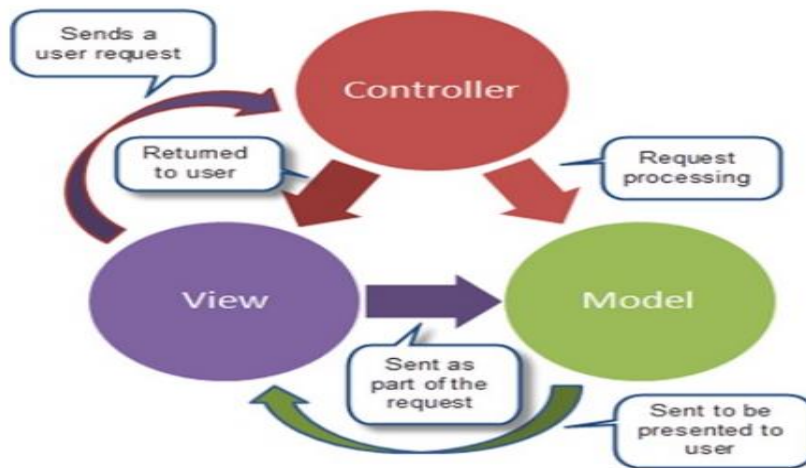
a. Đặc điểm của dịch vụ Web

- Cho phép client và server tương tác ngay cả trong môi trường khác nhau.
- Phần lớn được xây dựng dựa trên mã nguồn mở và phát triển các chuẩn đã được công nhận.
- Nó có thể triển khai bởi 1 phần mềm ứng dụng phía server (vd: PHP, Oracle Application server)
 - Ưu điểm

- Cung cấp khả năng hoạt động rộng lớn với các phần mềm khác nhau chạy trên tầng khác nhau.
- Sử dụng các giao thức chuẩn mở.
- Nâng cao khả năng tái sử dụng.
- Thúc đẩy đầu tư các hệ thống phần mềm đã tồn tại.
- Tạo mối quan hệ tương tác lẫn nhau và mềm dẻo giữa các thành phần trong hệ thống, dễ dàng cho việc phát triển các ứng dụng phân tán.
- Thúc đẩy hệ thống tích hợp, giảm sự phức tạp của hệ thống, hạ giá thành hoạt động, phát triển hệ thống nhanh và tương tác hiệu quả với hệ thống của các doanh nghiệp khác.
- Nhược điểm
 - Vào những khoảng thời gian chết của Web Service sẽ dẫn đến những thiệt hại lớn
- Giao diện không thay đổi.
- Có thể lỗi nếu một máy khách không được nâng cấp.
- Thiếu các giao thức cho việc vận hành.
 - Có quá nhiều chuẩn cho dịch vụ web khiến người dùng khó nắm bắt.
 - Phải quan tâm nhiều hơn tới vấn đề an toàn bảo mật.

b. Cách thức hoạt động

- Nền tảng cơ bản là MVC5.
- Asp.Net MVC là một Framework sử dụng .Net Framework cho việc phát triển ứng dụng web động, trước khi Asp.Net MVC ra đời, lập trình viên sử dụng công nghệ Asp.Net Web Form trên nền tảng .Net Framework để phát triển ứng dụng Web động. Asp.Net MVC phát triển trên mẫu thiết kế chuẩn MVC, cho phép người sử dụng phát triển các ứng dụng phần mềm. MVC là tên một mẫu phát triển ứng dụng, phương pháp này 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.



- Models:** Các đối tượng Models là một phần của ứng dụng, các đối tượng này thiết lập logic của phần dữ liệu của ứng dụng. Thông thường, các đối tượng model lấy và lưu trạng thái của model trong CSDL. Ví dụ như, một đối tượng Employee (nhân viên) sẽ lấy dữ liệu từ CSDL, thao tác trên dữ liệu và sẽ cập nhật dữ liệu trở lại vào bảng Employees ở SQL Server. Trong các ứng dụng nhỏ, model thường là chỉ là một khái niệm nhằm phân biệt hơn là được cài đặt thực thụ, ví dụ, nếu ứng dụng chỉ đọc dữ liệu từ CSDL và gửi chúng đến view, ứng dụng không cần phải có tầng model và các lớp liên quan. Trong trường hợp này, dữ liệu được lấy như là một đối tượng model.

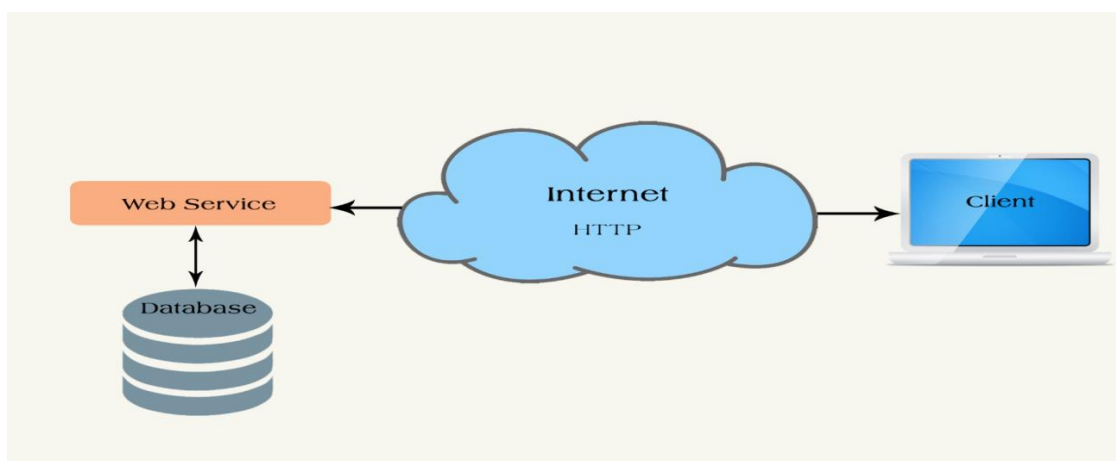
Views: Views là các thành phần dùng để hiển thị giao diện người dùng (UI). Thông thường, view được tạo dựa vào thông tin dữ liệu model. Ví dụ như, view dùng để cập nhật bảng Employees sẽ hiển thị các hộp văn bản, drop-down list, và các check box

Controllers: Controller là các thành phần dùng để quản lý tương tác người dùng, làm việc với model và chọn view để hiển thị giao diện người dùng. Trong một ứng dụng MVC, view chỉ được dùng để hiển thị thông tin, controller chịu trách nhiệm quản lý và đáp trả nội dung người dùng nhập và tương tác với người dùng. Ví dụ, controller sẽ quản lý các dữ liệu người dùng gửi lên (query-string values) và gửi các giá trị đó đến model, model sẽ lấy dữ liệu từ CSDL nhờ vào các giá trị này.
- Nền tảng ASP.NET MVC mang lại những lợi ích sau:

 - Dễ dàng quản lý sự phức tạp của ứng dụng bằng cách chia ứng dụng thành ba thành phần model, view, controller
 - Hỗ trợ tốt hơn cho mô hình phát triển ứng dụng hướng kiểm thử (TDD)
 - Hỗ trợ tốt cho các ứng dụng được xây dựng bởi những đội có nhiều lập trình viên và thiết kế mà vẫn quản lý được tính năng của ứng dụng

c. Cách thức hoạt động

- Mô hình hoạt động

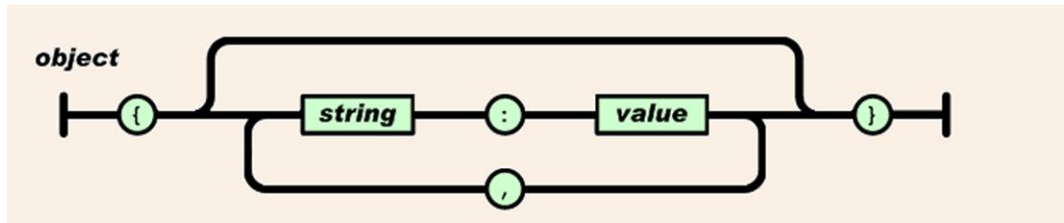


Hình 1.1 Mô hình chung của Web Service

- Giai đoạn triển khai công bố định nghĩa dịch vụ, xây dựng WSDL và triển khai mã thực thi của dịch vụ Web.
 - Giai đoạn tiến hành tìm kiếm và gọi thực thi dịch vụ Web bởi những người sử dụng dịch vụ.
 - Giai đoạn quản lý quản lý và quản trị dịch vụ, duy trì sự ổn định của dịch vụ, cập nhật thông tin mới, sửa lỗi khi nó xảy ra.
- Quy trình xây dựng một dịch vụ Web bao gồm các bước sau:
 - Định nghĩa và xây dựng các chức năng, các dịch vụ mà dịch vụ sẽ cung cấp.
 - Tạo WSDL cho dịch vụ.
 - Xây dựng SOAP hoặc REST server.
 - Đăng ký WSDL với UDDI registry để cho phép các client có thể tìm thấy và truy xuất.
 - Client nhận tập tin WSDL và từ đó xây dựng SOAP hoặc REST client để có thể kết nối với SOAP hoặc REST server.
 - Xây dựng ứng dụng phía client và sau đó gọi thực hiện dịch vụ thông qua việc kết nối tới SOAP hoặc REST server.
- XML – eXtensible Markup Language
 - Là ngôn ngữ đánh dấu với mục đích chung do W3C đề nghị.
 - Là một dạng chuẩn cho phép lưu các thông tin hướng cấu trúc, được tổ chức dưới dạng thẻ (tag) tương ứng.

- Các thẻ (tag) của XML thường không được định nghĩa trước mà chúng được tạo ra theo quy ước của người, (hoặc Chương trình) tạo ra XML theo những quy ước của chính người tạo.
 - Giúp đơn giản hóa việc chia sẻ dữ liệu giữa các hệ thống khác nhau, đặc biệt là các hệ thống được kết nối với Internet.
 - Sử dụng các khai báo kiểu dữ liệu DTD (Document Type Definition) hay lược đồ Schema để mô tả dữ liệu.
- JSON – JavaScript Object Notation
 - Định nghĩa dữ liệu theo ngôn ngữ JavaScript, tiêu chuẩn ECMA-262 năm 1999.
 - Là một định dạng văn bản đơn giản với các trường dữ liệu được lồng vào nhau.
 - Dùng để trao đổi dữ liệu giữa các thành phần của một hệ thống tương thích với hầu hết các ngôn ngữ C, C++, C#, Java, JavaScript, Perl, Python, v.v.
 - Vì sao nên sử dụng JSON ?
 - Có thể đọc hiểu và dễ dàng tiếp cận (human-readability).
 - Dữ liệu truyền tải ngắn gọn so với những định dạng dữ liệu khác như: XML, HTML,... → Tiết kiệm dung lượng hơn XML, HTML,...
 - Dễ dàng chuyển đổi (parse) dữ liệu từ dạng chuỗi (nhận từ server) sang dữ liệu có thể sử dụng được (Object, Number, Array).
 - Dễ truy cập nội dung.
 - Với những ứng dụng AJAX lấy và xử lý dữ liệu từ 1 web service nào đó khác domain. Nếu nội dung trả về có dạng JSON thì javascript từ trang web của chúng ta có thể trực tiếp truy cập.
 - Các kiểu dữ liệu JSON
 - JSON có 5 kiểu dữ liệu chính:
 1. Kiểu số (Number).
 2. Kiểu chuỗi (String).
 3. Kiểu mảng (Array).
 4. Kiểu đối tượng (Object).
 5. Giá trị (Value).
 - Kiểu đối tượng (Object)
 - 1 đối tượng là 1 hỗn độn của các cặp tên và giá trị.
 - 1 đối tượng bắt đầu bởi dấu ngoặc đơn trái { và kết thúc với dấu ngoặc đơn phải }.
 - Từng tên được theo sau bởi dấu 2 chấm (:) và các cặp tên/giá trị được tách ra bởi dấu phẩy (,).

- 1 đối tượng có thể chứa chiều cặp tên/giá trị (Name/values).



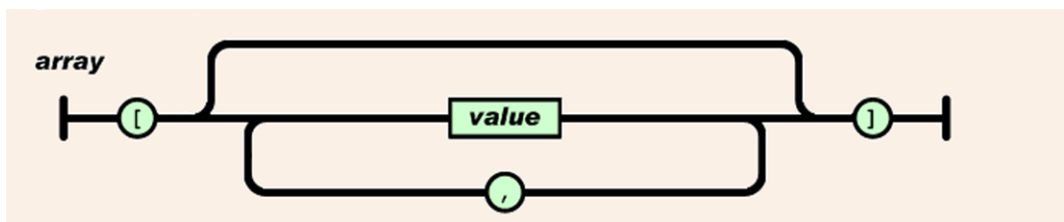
Hình 1.1

- Ví dụ:

```
{"Ten":"Cuong", "Ho":"Nguyen"}
```

- **Kiểu mảng (Array)**

- 1 mảng là 1 tập hợp các giá trị đã được sắp xếp.
- 1 mảng bắt đầu bởi dấu mở ngoặc vuông trái [và kết thúc với dấu ngoặc vuông phải].
- Các giá trị được cách nhau bởi dấu phẩy (,).
- 1 mảng có thể chứa nhiều đối tượng.



Hình 1.2

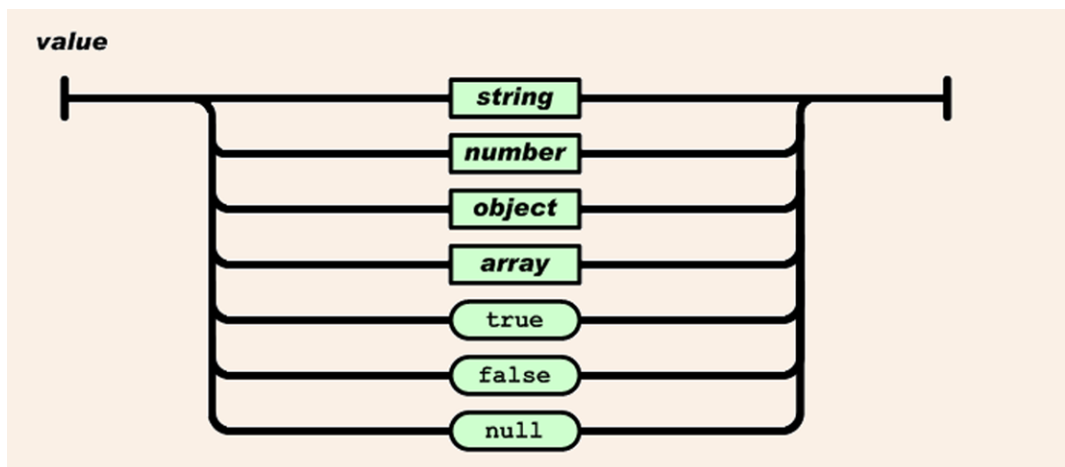
Ví dụ:

```
{
  "NhanVien":
  [
    {"tenNV":"Truong", "Ho":"Nguyen"},
    {"tenNV":"Dai", "Ho":"Dinh"},
    {"tenNV":"Hoc", "Ho":"Pham"}
  ]
}
```

}

- **Kiểu giá trị (Value)**

- Giá trị JSON có thể là:
 - 1 số nguyên (integer) hay số thực (floating point).
 - 1 chuỗi (nằm trong dấu nháy đôi "").
 - 1 luận lý (true hoặc false).
 - 1 mảng (nằm trong ngoặc vuông []).
 - 1 đối tượng (nằm trong dấu ngoặc nhọn {}).
 - Kiểu null.
- Những cấu trúc này có thể đã được lồng vào nhau.



Hình 1.3

- **Kiểu chuỗi (String)**

- 1 chuỗi (string) là 1 tập hợp của các số hay mẫu tự Unicode, được bao bọc trong các dấu trích dẫn kép (").
- Dùng dấu chéo (/) để thoát khỏi 1 chuỗi.
- 1 ký tự đã được hiển thị như là 1 chuỗi ký tự đơn đọc.

1.4.2.4 Kiến trúc của Dịch vụ Web

Dịch vụ Web gồm có 3 chuẩn chính: SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language) và UDDI (Universal Description, Discovery, and Integration). Hình 1 mô tả chồng giao thức của dịch vụ Web, trong đó UDDI được sử dụng để đăng ký và khám phá dịch vụ Web đã được miêu tả cụ thể trong WSDL. Giao tác UDDI sử dụng SOAP để nói chuyện với UDDI server, sau đó các ứng dụng SOAP yêu cầu một dịch vụ Web. Các thông điệp SOAP được gửi đi chính xác bởi HTTP và TCP/IP.

Chồng giao thức dịch vụ Web là tập hợp các giao thức mạng máy tính được sử dụng để định nghĩa, xác định vị trí, thi hành và tạo nên dịch vụ Web tương tác với những ứng dụng hay dịch vụ khác. Chồng giao thức này có 4 thành phần chính:

Dịch vụ vận chuyển (Service Transport): có nhiệm vụ truyền thông điệp giữa các ứng dụng mạng, bao gồm những giao thức như HTTP, SMTP, FTP, JSM và gần đây nhất là giao thức thay đổi khối mở rộng (Blocks Extensible Exchange Protocol- BEEP). Thông điệp XML: có nhiệm vụ giải mã các thông điệp theo định dạng XML để có thể hiểu được ở mức ứng dụng tương tác với người dùng. Hiện tại, những giao thức thực hiện nhiệm vụ này là XML-RPC, SOAP và REST. Mô tả dịch vụ: được sử dụng để miêu tả các giao diện chung cho một dịch vụ Web cụ thể. WSDL thường được sử dụng cho mục đích này, nó là một ngôn ngữ mô tả giao tiếp và thực thi dựa trên XML. Dịch vụ Web sẽ sử dụng ngôn ngữ này để truyền tham số và các loại dữ liệu cho các thao tác và chức năng mà dịch vụ Web cung cấp.

Khám phá dịch vụ: tập trung dịch vụ vào trong một nơi được đăng ký, từ đó giúp một dịch vụ Web có thể dễ dàng khám phá ra những dịch vụ nào đã có trên mạng, tốt hơn trong việc tìm kiếm những dịch vụ khác để tương tác. Một dịch vụ Web cũng phải tiến hành đăng ký để các dịch vụ khác có thể truy cập và giao tiếp. Hiện tại, UDDI API thường được sử dụng để thực hiện công việc này.

Trong đó, tầng giao thức tương tác dịch vụ (Service Communication Protocol) với công nghệ chuẩn là SOAP. SOAP là giao thức nằm giữa tầng vận chuyển và tầng mô tả thông tin về dịch vụ, cho phép người dùng triệu gọi một dịch vụ từ xa thông qua một thông điệp XML. Ngoài ra, để các dịch vụ có tính an toàn, toàn vẹn và bảo mật thông tin, trong kiến trúc dịch vụ Web, chúng ta có thêm các tầng Policy, Security, Transaction, Management.

1.4.2.5 An toàn cho dịch vụ Web

Dịch vụ Web liên kết và tương tác với các ứng dụng qua Internet, chính vì vậy bảo mật là một vấn đề được quan tâm khi các công ty tiến tới kết hợp ứng dụng với một dịch vụ Web. Việc đảm bảo an toàn cho dịch vụ Web là một vấn đề quan trọng, đặc biệt đối với những dịch vụ liên quan đến trao đổi tiền tệ, thông tin từ thị trường chứng khoán hay dịch vụ bán hàng qua mạng (liên quan đến trả tiền bằng tài khoản và có yêu cầu thông tin cá nhân của người dùng).

Trước khi có WS-Security (bảo mật cho dịch vụ Web) thì ý nghĩa thông thường của an toàn dịch vụ Web là bảo mật kênh truyền dữ liệu. Hiện nay, nó được thực hiện cho những SOAP/HTTP dựa trên cơ chế truyền thông điệp bằng cách sử dụng giao thức HTTPS. Không chỉ là an toàn ở mức truyền thông điệp, HTTPS còn cung cấp sự an toàn tới toàn bộ gói dữ liệu HTTP.

Mặc dù HTTPS không bao gồm tất cả các khía cạnh trong chuẩn an toàn chung cho dịch vụ Web nhưng nó đã cung cấp một lớp bảo mật khá đầy đủ với định danh, chứng thực, tính toàn vẹn thông điệp hay độ tin cậy.

Đảm bảo an toàn cho dịch vụ Web:

Khái niệm về WS-Security: đây là một chuẩn an toàn bao trùm cho SOAP, nó được dùng khi muốn xây dựng những dịch vụ Web toàn vẹn và tin cậy. Toàn vẹn có nghĩa là khi có một giao dịch hay khi truyền thông tin, hệ thống và thông tin sẽ không bị chặn, giao dịch sẽ không bị mất cũng như không thể có người lấy cắp được dữ liệu trên đường truyền. WS-security được thiết kế mang tính mở nhằm hướng tới những mô hình an toàn khác bao gồm PKI, Kerberos và SSL. Nó cũng đưa ra nhiều hỗ trợ cho các cơ chế an toàn khác, nhiều khuôn dạng chữ ký và công nghệ mã hóa, đảm bảo sự an toàn, toàn vẹn thông điệp và tính tin cậy của thông điệp. Tuy nhiên, WS-security cũng chưa thể đảm bảo được tất cả yêu cầu về bảo mật và an toàn thông tin, nó chỉ là một trong những lớp của giải pháp an toàn cho dịch vụ Web.

Tính toàn vẹn tạo ra một chữ ký số hóa XML dựa trên nội dung của thông điệp. Nếu dữ liệu bị thay đổi bất hợp pháp, nó sẽ không còn thích hợp với chữ ký số hóa XML đó. Chữ ký này được tạo ra dựa trên khóa mà người gửi thông điệp tạo ra, do đó người nhận chỉ nhận thông điệp khi có chữ ký sử dụng và nội dung phù hợp. Ngược lại sẽ có một thông báo lỗi. Việc chứng thực được thực hiện giữa client và server là cách chứng thực rất cơ bản (sử dụng định danh người dùng và mật khẩu).

WS-security chỉ là một trong những lớp an toàn và bảo mật cho dịch vụ Web, vì vậy cần một mô hình an toàn chung lớn hơn để có thể bao quát được các khía cạnh khác. Các thành phần được thêm có thể là WS-Secure Conversation Describes, WS-Authentication Describes, WS-Policy Describes hay WS-Trust Describes. Chúng sẽ thực hiện việc đảm bảo an toàn hơn cho hệ thống khi trao đổi dữ liệu, mở và đóng các phiên làm việc cũng như quản lý dữ liệu cần chứng thực và chính sách chứng thực.

1.4.2.6 Xây dựng một dịch vụ Web

Có 4 giai đoạn chính để xây dựng một dịch vụ Web là xây dựng, triển khai, tiến hành và quản lý, trong đó:

Giai đoạn xây dựng bao gồm phát triển và chạy thử ứng dụng dịch vụ Web, xây dựng các chức năng và định nghĩa dịch vụ. Có hai cách khác nhau để tiến hành trong giai đoạn này, đó là Red-path- solid và Blue-path-dashed. Với Red-path-solid, chúng ta sẽ xây dựng một dịch vụ Web mới từ trạng thái ban đầu hoặc với một dịch vụ đã có sẵn. Từ đó, xây dựng định nghĩa service (WSDL) với các đối tượng, hàm chức năng mà chúng ta mong muốn. Nếu theo

cách Blue-path-dashed, dịch vụ Web sẽ được xây dựng từ đầu hoặc từ một định nghĩa dịch vụ WSDL. Sử dụng WSDL này, xây dựng hoặc sửa đổi lại mã để thực hiện các yêu cầu mong muốn trong dịch vụ Web.

Giai đoạn triển khai: công bố định nghĩa dịch vụ, xây dựng WSDL và triển khai mã thực thi của dịch vụ Web. Triển khai dịch vụ Web tới một ứng dụng phía server, sau đó sẽ công bố dịch vụ Web trên mạng Internet để các client có thể nhìn thấy. Sử dụng UDDI registry để công bố lên mạng.

Giai đoạn tiên hành: tìm kiếm và gọi thực thi dịch vụ Web bởi những người dùng muốn sử dụng dịch vụ.

Quản lý: Quản lý và quản trị dịch vụ, duy trì sự ổn định của dịch vụ, cập nhật thông tin mới, sửa lỗi khi nó xảy ra...

Để xây dựng một dịch vụ Web, chúng ta cần hiểu được những việc phải làm và nên bắt đầu từ đâu. Có 3 cách tiếp cận chủ yếu để xây dựng nên một dịch vụ Web, có thể từ một ứng dụng đã có (bottom-up); từ một định nghĩa dịch vụ, WSDL để phát sinh một ứng dụng mới (top-down) hoặc có thể từ một nhóm các dịch vụ Web hiện có, kết hợp lại với nhau để tạo nên các chức năng mới hoặc mở rộng thêm chức năng. Những hướng tiếp cận này dựa trên những gì mà chúng ta đã có, tùy thuộc vào yêu cầu của hệ thống, trong đó tối đa việc sử dụng lại các chức năng, các thành phần, môđun đã được xây dựng.

Quy trình xây dựng một dịch vụ Web bao gồm các bước sau:

Định nghĩa và xây dựng các chức năng, các dịch vụ mà dịch vụ sẽ cung cấp (sử dụng ngôn ngữ Java chẳng hạn)

Tạo WSDL cho dịch vụ

Xây dựng SOAP server

Đăng ký WSDL với UDDI registry để cho phép các client có thể tìm thấy và truy xuất.

Client nhận file WSDL và từ đó xây dựng SOAP client để có thể kết nối với SOAP server

Xây dựng ứng dụng phía client (chẳng hạn sử dụng Java) và sau đó gọi thực hiện dịch vụ thông qua việc kết nối tới SOAP server.

Lựa chọn một ngôn ngữ, xây dựng các tiến trình nghiệp vụ và chúng ta bắt đầu tạo nên một dịch vụ Web như ý muốn. Sau đó là cung cấp dịch vụ Web này trên Internet.

1.4.2.7. Tích hợp dịch vụ Web theo chuẩn

Để có thể thành công với dịch vụ Web chúng ta phải quan tâm đến khá nhiều vấn đề, bao gồm việc triển khai, giám sát và tích hợp hệ thống. Doanh nghiệp không những phải phát triển một ứng dụng dịch vụ Web mới mà còn phải tích hợp các ứng dụng nghiệp vụ phụ trợ của họ trong kiến trúc Dịch vụ Web. Cùng với việc triển khai và tích hợp, những nhà kinh doanh và những người sử dụng kỹ thuật cũng cần có khả năng giám sát, triển khai toàn diện để đảm bảo hoạt động kinh doanh hiệu quả và tin cậy.

Giám sát (monitoring): Cần hỗ trợ ở cả mức công cụ và cơ sở hạ tầng để giám sát các dịch vụ Web chạy như thế nào qua toàn bộ mạng, từ một chi nhánh con của một công ty trên mạng tới các chi nhánh khác trong công ty hay giao tiếp với doanh nghiệp khác. Kết hợp thông báo theo sự kiện với các lỗi trong luồng nghiệp vụ cho những người dùng không có kinh nghiệm giám sát dịch vụ Web và các dịch vụ kế thừa khác.

Xác định đường đi dữ liệu (Data routing): Việc thiết lập đường đi của dữ liệu giữa những thành phần của dịch vụ Web hướng tới tối đa hóa khả năng sử dụng lại. Nếu coi một thành phần (component) là một đối tượng thì mỗi thể hiện (instance) của nó sẽ không quan tâm đến các thể hiện khác của cùng thành phần đó. Những thể hiện của cùng một thành phần có thể dễ dàng được sử dụng lại trong các ứng dụng phân tán khác bởi vì chúng hoàn toàn độc lập và không phụ thuộc lẫn nhau.

Triển khai (Deployment): Triển khai các dịch vụ Web có khả năng nâng cấp, điều khiển và cấu hình các thành phần từ xa thông qua mạng phân tán.

Quản lý (Management): Có thể xây dựng theo kiến trúc P2P (Peer-to-Peer). Các hoạt động chính như thực thi các thành phần, định tuyến dữ liệu, xử lý luồng công việc và chuyển đổi dữ liệu được thực hiện tại các điểm cuối của mạng. Server sẽ tập trung giải quyết các hoạt động khác như quản lý, điều khiển sự kiện, chứng thực bảo mật và quản trị.

Cấu hình và quản lý phiên bản (Configuration and version management): Sử dụng các công cụ linh hoạt để quản lý các phiên bản khác nhau của dịch vụ Web, cho phép các phiên bản được nâng cấp và điều khiển từ một công cụ quản lý tập trung. Kết hợp giữa ứng dụng và mạng giúp các kỹ sư triển khai có thể điều khiển các thành phần chạy trên nền tảng hệ thống phần cứng cụ thể bên trong mạng.

Bảo mật (Security): các chuẩn mở như HTTP, XML, SOAP, WSDL và chuẩn bảo mật JSM được sử dụng rộng rãi khiến chúng trở thành lý tưởng để xây dựng các ứng dụng web. Đầu tiên, dịch vụ Web sử dụng những công nghệ này giống như firewall, SSL và các chứng nhận số. Dịch vụ Web thế hệ sau này sẽ kết hợp với những công nghệ có khả năng bảo mật cao hơn, giống như mã hóa XML và chứng nhận số XML.

Như vậy, với một dịch vụ Web, việc giao tiếp và truyền nhận dữ liệu trở nên dễ dàng và hiệu quả hơn, đồng thời đem lại chi phí thấp hơn và tăng cường những khả năng giao tiếp thời gian thực, kết nối với mọi người trên khắp thế giới. Bản chất của nền tảng công nghệ này là kiến trúc hướng dịch vụ và sự phát triển của dịch vụ Web có tương lai rất khả quan.

1.2 Giới thiệu về ngôn ngữ MVC5 và SQL

1. Ngôn ngữ MVC5

Các ứng dụng web là các chương trình thực thi trên máy chủ web (Web server) và truy cập và tương tác với người sử dụng thông qua trình duyệt web. Khái niệm này được người sử dụng thừa nhận rộng rãi, tuy nhiên ngày nay CNTT phát triển mạnh mẽ, một loạt các nền tảng công nghệ ra đời, ứng dụng web hoàn toàn có thể không cần phải tương tác hoặc thực thi trên máy chủ web, có nghĩa là ứng dụng web này chạy thuần túy trên trình duyệt web. Điều đó có nghĩa ứng dụng web không cần kết nối Internet vẫn có thể chạy được. Ứng dụng web cho phép người sử dụng có thể chia sẻ và truy cập thông tin thông qua Internet. Lập trình viên có thể tạo các ứng dụng web để phục vụ mục tiêu thương

Ứng dụng web có thể được nhìn theo nhiều hướng khác nhau: Hướng nhìn của các nhà phát triển ứng dụng, thường chia ứng dụng web thành nhiều các phân tầng khác nhau, mỗi phân tầng đảm nhiệm một số các nhiệm vụ mang tính logic của ứng dụng. Hướng nhìn triển khai ứng dụng, thường phân chia ứng dụng thành các thành phần vật lý khác nhau, các thành phần vật lý này được cài đặt tại các máy tính (các nút mạng) khác nhau trong hệ thống.

2. Hệ quản trị CSDL SQL

SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS)) sử dụng câu lệnh SQL (**Transact-SQL**) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.

SQL Server được tối ưu để có thể chạy trên môi trường cơ sở dữ liệu rất lớn (Very Large Database Environment) lên đến Tera-Byte và có thể phục vụ cùng lúc cho hàng ngàn user. SQL Server có thể kết hợp “ăn ý” với các server khác như Microsoft Internet Information Server (IIS), E-Commerce Server, Proxy Server

SQL được sử dụng cho việc hỗ trợ NodeJs, Visua C# và nhiều ngôn ngữ khác, làm nơi lưu trữ những thông tin trên các trang web viết bằng MVC5

CHƯƠNG 2: CỞ SỞ LÝ THUYẾT

2.1 Web service và chức năng

2.1.1 Tổng quan

- Loại web là hoạt động về dịch vụ.
- Sản phẩm, lĩnh vực hoạt động của web là cung cấp giải pháp quản lý sửa chữa trên nền web cho doanh nghiệp. Thông tin bảo hành được khởi tạo, lập, gửi, nhận, lưu trữ và quản lý bằng phương tiện điện tử. Được đăng ký bằng chữ ký điện tử (Ký số), có giá trị về mặt pháp lý như hóa đơn giấy thông thường. có thể chuyển đổi thành hóa đơn giấy khi có nhu cầu.

2.1.2 Yêu cầu kỹ thuật giao tiếp

- **Webservice bảo hành sửa chữa**

- **Đầu vào:**

- Webservice dùng chung trong các trường hợp lập hóa đơn gốc, lập hóa đơn điều chỉnh tiền, lập hóa đơn điều chỉnh thông tin, lập hóa đơn thay thế
- Data: Định dạng JSON hoặc XML

- Sản phẩm, lĩnh vực hoạt động của web là cung cấp giải pháp quản lý bảo hành sửa chữa trên nền web cho doanh nghiệp. Thông tin được khởi tạo, lập, gửi, nhận, lưu trữ và quản lý bằng phương tiện điện tử. Được đăng ký bằng chữ ký điện tử (Ký số), có giá trị về mặt pháp lý như hóa đơn giấy thông thường. có thể chuyển đổi thành hóa đơn giấy khi có nhu cầu.

2.1.3 Các chức năng

- Lập hóa đơn.
- Lấy file hóa đơn.

2.2 Xây dựng 1 số lớp giao diện

2.2.1 Dịch vụ lưu trữ trực tuyến (Hosting)

- Dịch vụ lưu trữ trực tuyến ở đây em lựa chọn sử dụng hosting free của trang 000webhost.com vì lý do chi phí và hiện tại cũng chưa đưa vào thực tiễn nên sẽ sử dụng hosting trả phí sau.

2.2.2 Mô tả nghiệp vụ Dịch vụ web (Webservice)

- Kỹ thuật thực hiện
 - Ủy quyền: Tài khoản và mật khẩu của khách hàng phải được dùng phương thức xác thực Basic Authentication.
 - Giao thức giao tiếp: chuẩn RESTful.
 - Method: Tùy thuộc đường link lựa chọn POST hoặc GET.
 - Dữ liệu gửi lên server: có thể là JSON hoặc FormParam.
 - Gửi dữ liệu dạng JSON cần khai báo kiểu: Content-Type: application/json.
 - Gửi dữ liệu dạng FormParam cần khai báo kiểu: Content-Type: application/x-www-form-urlencoded.
 - Nhận dữ liệu cần khai báo kiểu: Accept: application/json.
- Xác thực Basic Authentication
 - Là một phương thức xác thực phổ thông có trên nền tảng ứng dụng Web. Nó sẽ xuất hiện ra khi Client yêu cầu những thông tin phải được xác thực.
 - Để truy cập vào các nguồn dữ liệu (Resource) được bảo mật bởi Basic Authentication, người dùng phải gửi một request và trong request đó có chứa thông tin username/password được đính kèm trên Header. Chuỗi đầu vào này sẽ được mã hóa.
- Ví dụ: MDEwNDgzMTAzMDoxMjM=
 - **REST** viết tắt cho REpresentational State Transfer. Khái niệm về REST lần đầu tiên được giới thiệu vào năm 2000 trong luận văn Tiến sĩ của Roy Fielding (đồng sáng lập giao thức HTTP). Là 1 kiểu kiến trúc lập trình, định nghĩa các quy tắc để thiết kế web service chú trọng vào tài nguyên. Mọi thứ trong REST đều được coi là tài nguyên và được định danh thông qua URI, và có thể được biểu diễn thông qua dạng văn bản, XML, JSON RESTful là những ứng dụng mà có sử dụng kiến trúc REST.
- **REST** là một bộ quy tắc để tạo ra một ứng dụng Web Service, mà nó tuân thủ 4 nguyên tắc thiết kế cơ bản sau:
 1. Sử dụng các phương thức HTTP một cách rõ ràng
 2. Phi trạng thái
 3. Hiện thị cấu trúc thư mục như các URIs
 4. Truyền tải JavaScript Object Notation (JSON), XML hoặc cả hai.

- **RESTful Web Service** là các Web Service được viết dựa trên kiến trúc REST. REST đã được sử dụng rộng rãi thay thế cho các Web Service dựa trên SOAP và WSDL. RESTful Web Service nhẹ (lightweigh), dễ dàng mở rộng và bảo trì.

Ràng buộc trong REST

- **Client - Server:** Hoạt động theo mô hình Client - Server, việc tách biệt này nhằm đơn giản hóa việc thực hiện các thành phần, giảm sự phức tạp của ngữ nghĩa kết nối, nâng cao hiệu quả của việc điều chỉnh hiệu năng, tăng khả năng mở rộng của máy chủ.
- **Stateless:** Hiểu đơn giản là server và client không lưu trạng thái của nhau. Với mỗi một request được gửi đi đều phải được đóng gói đầy đủ thông tin để server có thể nhận và hiểu được. Điều này giúp hệ thống dễ phát triển, bảo trì, mở rộng vì không tốn công CRUD trạng thái của Client. Tuy nhiên có mặt hạn chế là làm tăng lưu lượng thông tin cần truyền tải giữa client và server.
- **Khả năng caching:** Các response có thể lấy từ cache. Bằng cách cache các response, server giảm tải việc xử lý request, client cũng nhận được thông tin nhanh hơn.
- **Chuẩn hóa Interface:** Nhằm đơn giản hóa và tách biệt kiến trúc, cho phép từng phần phát triển độc lập,, người phát triển đã tạo ra API cơ bản để thiết kế bất kỳ dịch vụ REST nào (dù là web hay mobile thì đều có thể kết nối vào được). Tuy nhiên khi chuẩn hóa thì ta không thể tối ưu từng kết nối được.
- **Phân lớp hệ thống:** giảm mức độ phức tạp của hệ thống, giúp các thành phần tách biệt nhau từ đó dễ dàng mở rộng. Với mỗi một lớp chỉ trao đổi trực tiếp với lớp ngay dưới và trên nó.

Tại sao nên dùng REST?

- Thiết kế web trước đây sử dụng SOAP (Simple Object Access Protocol) và WSDL (Web Service Definition Language), tuy nhiên bây giờ REST tối ưu hơn so với 2 phương pháp này.
- Rõ ràng về URL (REST URL đại diện cho resource xác định chứ không phải hành động)
- Trả về nhiều định dạng khác nhau như: html, xml, ...
- Code ngắn gọn dễ hiểu
- Hiệu suất tốt, tin cậy, dễ phát triển.

Resources

Do REST tập trung vào tài nguyên, nên ta sẽ tìm hiểu qua một chút về tài nguyên trong REST Các đặc tính của Resource:

- Hiện thị dữ liệu: Dữ liệu có thể được thể hiện ở nhiều dạng khác nhau (binary, JSON, XML .v.v)
- Nhận diện rõ ràng: Mỗi URL tại một thời điểm chỉ trả về 1 tài nguyên xác định.
- Dữ liệu điều khiển : cache-control.
- Dữ liệu mô tả (metadata) : Kiểu nội dung, lần cập nhật mới .v.v

a) Method

REST đặt ra một quy tắc đòi hỏi lập trình viên xác định rõ ý định của mình thông qua các phương thức của HTTP. Thông thường ý định đó bao gồm lấy dữ liệu, trên dữ liệu, cập nhật dữ liệu hoặc xóa dữ liệu. Vậy khi bạn muốn thực hiện một trong các ý định trên hãy lưu ý các quy tắc sau:

- Để tạo một tài nguyên trên máy chủ, bạn cần sử dụng phương thức POST.
- Để truy xuất một tài nguyên, sử dụng GET.
- Để thay đổi trạng thái một tài nguyên hoặc để cập nhật nó, sử dụng PUT.
- Để huỷ bỏ hoặc xoá một tài nguyên, sử dụng DELETE.

Chú ý rằng các nguyên tắc ở trên là không bắt buộc, thực tế bạn có thể sử dụng phương thức GET để yêu cầu lấy dữ liệu, trên, sửa hoặc xóa dữ liệu trên Server. Tuy nhiên REST đưa ra các nguyên tắc ở trên mục đích đưa mọi thứ trở lên rõ ràng và dễ hiểu.

b) Định dạng dữ liệu gửi lên server (Đầu vào Web Service)

- Kiểu dữ liệu như đã được giới thiệu ở mục 1.3.6 em chọn sử dụng 1 loại là JSON.
- Lý do là JSON dễ học và chỉ cần sử dụng 1 hàm chuyển sang JSON và hàm chuyển về là xong.

c) Gửi dữ liệu dạng JSON

- Với những link Web Service yêu cầu dữ liệu đầu vào là JSON thì ta cần thêm cú pháp Content-Type: application/json vào phần Header.

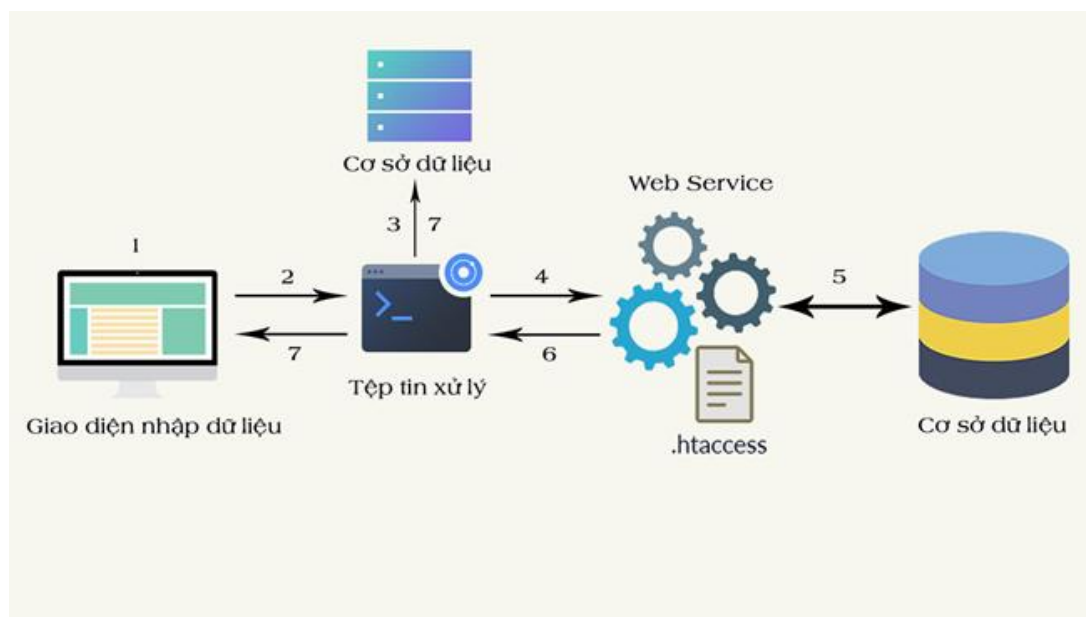
d) Gửi dữ liệu dạng FormParam

- Với những link Web Service yêu cầu dữ liệu đầu vào là FormParam thì ta cần thêm cú pháp Content-Type: application/x-www-form-urlencoded vào phần Header.

e) Nhận dữ liệu phản hồi

- Tất cả dữ liệu trả về từ server đều là JSON. Ta cần thêm cú pháp Accept: application/json vào phần Header.

2.2.3 Mô hình hoạt động



Hình 2.8 Mô hình hoạt động

Quy trình hoạt động

Bước 1. Người dùng nhập dữ liệu vào form chức năng.

Bước 2. Khi ấn nút thao tác (submit) sẽ được chuyển dữ liệu tới tập tin xử lý tương ứng.

Bước 3. Tập tin xử lý nhận được dữ liệu truyền tới. Sẽ lưu vào csdl rồi gán vào 1 mảng định dạng JSON hoặc sử dụng hàm `json_encode` đúng theo yêu cầu đầu vào của web service Viettel.

Bước 4. Tập tin xử lý gửi tiếp mảng dữ liệu JSON kèm theo đó là các thông tin kỹ thuật ở phần 3.4.2 tới đường link web service Viettel tương ứng chức năng.

Bước 5. Web service Viettel sẽ nhận dữ liệu và xử lý dữ liệu với cơ sở dữ liệu của họ.

Bước 6. Web service gửi lại phản hồi là một chuỗi dữ liệu ở định dạng JSON.

Bước 7. Tập tin xử lý nhận được phản hồi sẽ báo cho người dùng các dữ liệu để họ lưu lại và đồng thời lưu thêm thông tin này vào csdl.

CHƯƠNG 3: PHÂN TÍCH THIẾT KẾ HỆ THỐNG

3.1 Phân Tích Thiết Kế Hệ Thống Website

a. Bài toán.

Công ty cổ phần CETI chuyên buôn bán, sửa chữa và bảo hành các thiết bị di động thông minh. Công ty cần một hệ thống phần mềm cho phép quản trị viên có thể theo dõi các hoạt động bảo hành sửa chữa của cửa hàng đưa ra yêu cầu cụ thể về vật tư, thông tin khách hàng, thời hạn bảo hành của vật tư. Nhà quản lý có thể thấy ngay được danh sách các chi tiết bảo hành sửa chữa của các sản phẩm họ cung cấp, thời hạn bảo hành. Nhà quản lý thông qua đó có thể biết được các sản phẩm của khách hàng đem đến có còn bảo hành miễn phí hay không và sau đó quyết định sẽ bảo hành thu phí đối với khách hàng hay không.

Nhiệm vụ cần thực hiện là xây dựng hệ thống quản lý bảo hành và sửa chữa vật tư như bài toán trên.

b. Khó khăn phát sinh

Hằng ngày công việc bảo hành và sửa chữa của công ty quá bận rộn, một phần là do khách hàng đến bảo hành và sửa chữa khá đông và mọi thông tin về bảo hành và sửa chữa còn khá lộn xộn do mọi thông tin của khách hàng và các sản phẩm điều lưu trên giấy tờ và sổ sách sẽ dẫn đến công ty không thể sắp xếp được số lượng nhân viên hoạt động

c. Hướng giải quyết

- Theo như hiện tại Công ty cần một phần mềm quản lý tiện dụng và hiệu quả hơn, tiết kiệm thời gian cho khách hàng cũng như nhân viên. Các vấn đề cần thực hiện:
- Khách hàng: Sau khi khách hàng mua sản phẩm sẽ để lại thông tin.
- Chi tiết sửa chữa : Nhân viên sẽ tiến hành sửa chữa cho khách hàng và sau đó báo lại với quản trị viên
- Quản lý: Kiểm soát toàn bộ quá trình, thông tin khách hàng, sản phẩm được sửa chữa.

Bước 2: Lập kế hoạch

Sử dụng các thông tin thu thập được từ giai đoạn một, đó là thời gian để đặt cùng một kế hoạch cho trang web của bạn. Đây là điểm mà một bản đồ trang web được phát triển.

Bản đồ trang web là một danh sách của tất cả các lĩnh vực chủ đề chính của trang web, cũng như chủ đề phụ, nếu có. Nó phục vụ như một hướng dẫn như những gì nội dung sẽ được trên trang web, và là điều cần thiết để phát triển một, dễ hiểu hệ thống định vị phù hợp. Người dùng cuối của trang web – hay còn gọi là khách hàng của bạn – phải được lưu ý

khi thiết kế trang web của bạn. Đây là, sau khi tất cả, những người sẽ được tìm hiểu về dịch vụ của bạn hoặc mua sản phẩm của bạn. Một giao diện người dùng tốt tạo ra một cách dễ dàng để điều hướng trang web, và là cơ sở cho việc này.

Trong giai đoạn lập kế hoạch, thiết kế web của bạn cũng sẽ giúp bạn quyết định những gì công nghệ phải được thực hiện. Các yếu tố như hình thức tương tác, thương mại điện tử, flash, vv sẽ được thảo luận khi lập kế hoạch trang web của bạn.

Bước 3: Thiết kế

Từ các thông tin thu thập đến thời điểm này, đó là thời gian để xác định giao diện của trang web của bạn.

Đối tượng mục tiêu là một trong những yếu tố quan trọng được xem xét. Một trang web nhằm vào thanh thiếu niên, ví dụ, sẽ thấy mình khác nhau hơn một có nghĩa là cho một tổ chức tài chính. Là một phần của giai đoạn thiết kế, nó cũng quan trọng để kết hợp các yếu tố như logo của công ty hoặc màu sắc để giúp tăng cường bản sắc của công ty của bạn trên trang web.

Thiết kế web của bạn sẽ tạo ra một hoặc nhiều thiết kế nguyên mẫu cho trang web của bạn. Đây thường là một hình ảnh jpg. Những gì thiết kế cuối cùng sẽ như thế nào. Thông thường bạn sẽ nhận được một email với mock-up cho trang web của bạn, trong khi thiết kế khác mang nó một bước xa hơn bằng cách cho bạn truy cập vào một khu vực an toàn của trang web của họ có nghĩa là cho khách hàng để xem công việc đang tiến.

Dù bằng cách nào, thiết kế của bạn sẽ cho phép bạn xem các dự án của bạn trong thiết kế và phát triển giai đoạn. Lý do quan trọng nhất của việc này là nó mang lại cho bạn cơ hội để thể hiện thích và không thích của bạn trên thiết kế trang web.

Trong giai đoạn này, giao tiếp giữa bạn và thiết kế của bạn là rất quan trọng để đảm bảo rằng các trang web cuối cùng sẽ phù hợp với nhu cầu và khẩu vị của bạn. Điều quan trọng là bạn làm việc chặt chẽ với các nhà thiết kế của bạn, trao đổi ý tưởng, cho đến khi bạn đạt được thiết kế cuối cùng cho trang web của bạn.

Sau đó phát triển có thể bắt đầu ...

Bước 4: Phát triển :

Giai đoạn phát triển là điểm mà các trang web riêng của mình được tạo ra. Tại thời điểm này, nhà thiết kế web của bạn sẽ mất tất cả các yếu tố đồ họa cá nhân từ các mẫu thử nghiệm và sử dụng chúng để tạo ra, trang web chức năng thực tế.

Điều này thường được thực hiện bằng cách đầu tiên phát triển các trang chủ, theo sau là một “vỏ” cho các trang trí nội thất. Vỏ phục vụ như là một khuôn mẫu cho các trang nội dung của trang web của bạn, vì nó có chứa các cấu trúc định hướng chính cho trang web. Một khi lớp vỏ đã được tạo ra, thiết kế của bạn sẽ mất nội dung của bạn và sử dụng nó trong trang web, trong các lĩnh vực thích hợp.

Các yếu tố như hình thức liên lạc tương tác, hình ảnh động flash hoặc xe đẩy mua sắm thương mại điện tử được thực hiện và thực hiện chức năng trong giai đoạn này, là tốt.

Toàn bộ thời gian này, thiết kế của bạn nên tiếp tục làm cho trang web theo tiến độ của bạn có sẵn cho bạn để xem, để bạn có thể đề nghị bất kỳ thay đổi bổ sung hay chỉnh sửa bạn muốn làm.

Về mặt kỹ thuật, một trang web thành công đòi hỏi một sự hiểu biết về front-end phát triển web. Điều này liên quan đến việc viết mã XHTML / CSS tuân thủ các tiêu chuẩn web hiện tại, tối đa hóa chức năng, cũng như khả năng tiếp cận cho là lớn khán giả càng tốt.

Bước 5: Bảo trì :

Sự phát triển của trang web của bạn không nhất thiết phải có. Một cách để mang lại truy cập lặp lại vào trang web của bạn là cung cấp nội dung mới hoặc các sản phẩm một cách thường xuyên. Hầu hết các nhà thiết kế web sẽ được hạnh phúc để tiếp tục làm việc cùng với các bạn hơn, để cập nhật thông tin trên trang web của bạn. Nhiều nhà thiết kế cung cấp các gói bảo trì ở mức giá giảm, dựa trên mức độ thường xuyên bạn dự đoán thay đổi hoặc bổ sung vào trang web của bạn.

Nếu bạn muốn có nhiều bàn tay trên, và cập nhật nội dung của riêng bạn, có cái gì đó gọi là CMS (Content Management System) có thể được thực hiện để trang web của bạn. Đây là cái gì đó sẽ được quyết định trong giai đoạn lập kế hoạch. Với CMS, thiết kế của bạn sẽ sử dụng phần mềm trực tuyến để phát triển một hướng trang web cơ sở dữ liệu cho bạn.

Một trang web do một CMS cung cấp cho bạn khả năng chỉnh sửa các lĩnh vực nội dung của trang web mình. Bạn có quyền truy cập vào một khu vực hành chính phía sau, nơi bạn có thể sử dụng một trình soạn thảo văn bản trực tuyến (tương tự như một phiên bản mini của

Microsoft Word). Bạn sẽ có thể chỉnh sửa nội dung hiện theo cách này, hoặc nếu bạn cảm thấy mạo hiểm hơn, thậm chí bạn có thể thêm các trang mới và nội dung chính mình. Các khả năng là vô tận!

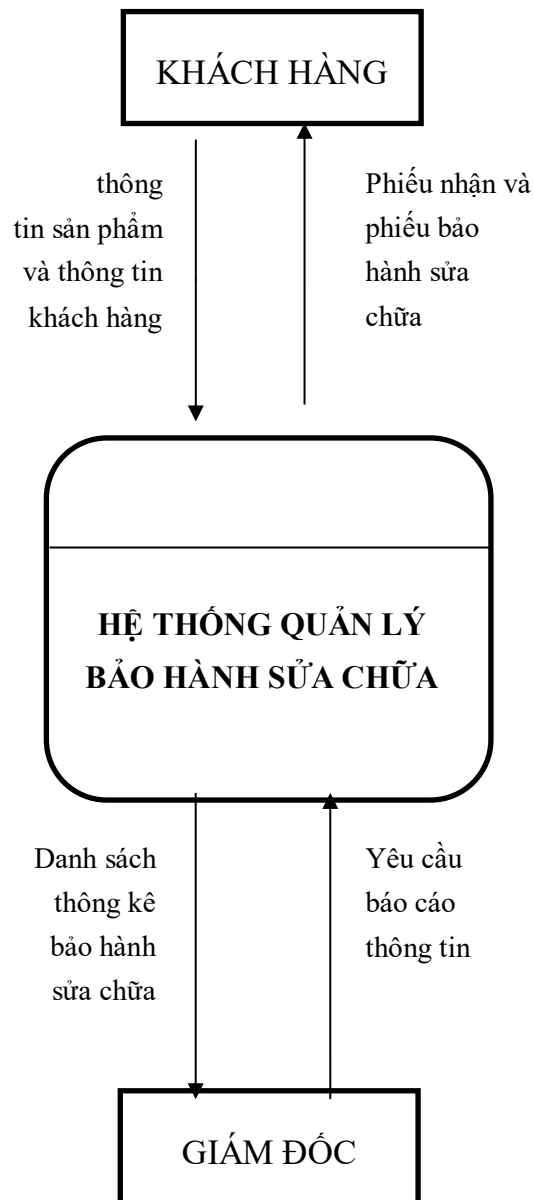
Một số người thích có tất cả các kiểm soát để họ có thể thực hiện cập nhật cho trang web của mình những phút họ quyết định làm như vậy. Những người khác thích để tay khỏi trang web hoàn toàn, vì chúng có đủ các nhiệm vụ trên tay mà quan trọng hơn cho họ để xử lý trực tiếp.

Đó là nơi mà sự giúp đỡ của một nhà thiết kế web của bạn đến, một lần nữa, vì chúng có thể tiến hành bảo trì trang web cho bạn – một trong ít điều để bạn có thể làm luôn luôn là một điều tốt trong những thời điểm bận rộn!

Các mặt hàng loại bảo dưỡng khác bao gồm SEO (Search Engine Optimization), SES (Công cụ Tìm kiếm thông tin). Đây là tối ưu hóa các trang web bạn với các yếu tố như tiêu đề, mô tả và thẻ từ khóa giúp trang web của bạn đạt được thứ hạng cao trong công cụ tìm kiếm. Các mã xác nhận đề cập trước đó là cái gì đó đóng một vai trò quan trọng trong SEO, là tốt.

3.3 Sơ đồ ngữ cảnh

a. Biểu đồ



Biểu đồ 2.1 Biểu đồ ngữ cảnh

b. Mô tả hoạt động

KHÁCH HÀNG:

- Khách hàng đưa sản phẩm và thông tin cá nhân
- Khách hàng nhận về phiếu nhận và hóa đơn bảo hành sửa chữa

HỆ THỐNG:

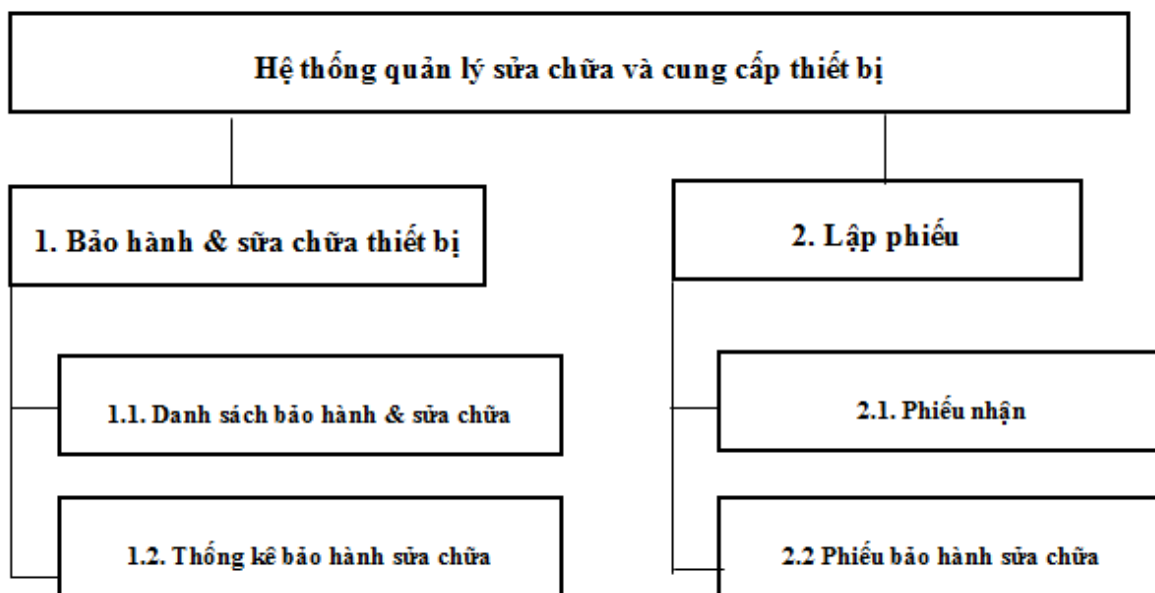
- Hệ thống sẽ cho quản trị viên biết sản phẩm còn bảo hành hay không
- Hệ thống lập phiếu nhận, các hóa đơn

GIÁM ĐỐC:

- Gửi yêu cầu báo cáo danh sách
- Sau khi sửa chữa xong gửi lại thông tin sửa chữa và sản phẩm lại cho quản trị viên

3.4 Sơ đồ phân rã chức năng

a. Sơ đồ



b. Mô hình chức năng lá

1. Bảo hành và sửa chữa thiết bị
 - 1.1. Danh sách bảo hành & sửa chữa
 - 1.2. Thống kê bảo hành & sửa chữa
2. Lập phiếu
 - 2.1. Phiếu nhận
 - 2.2. Phiếu bảo hành và sửa chữa

3.5 Danh sách các hồ sơ dữ liệu

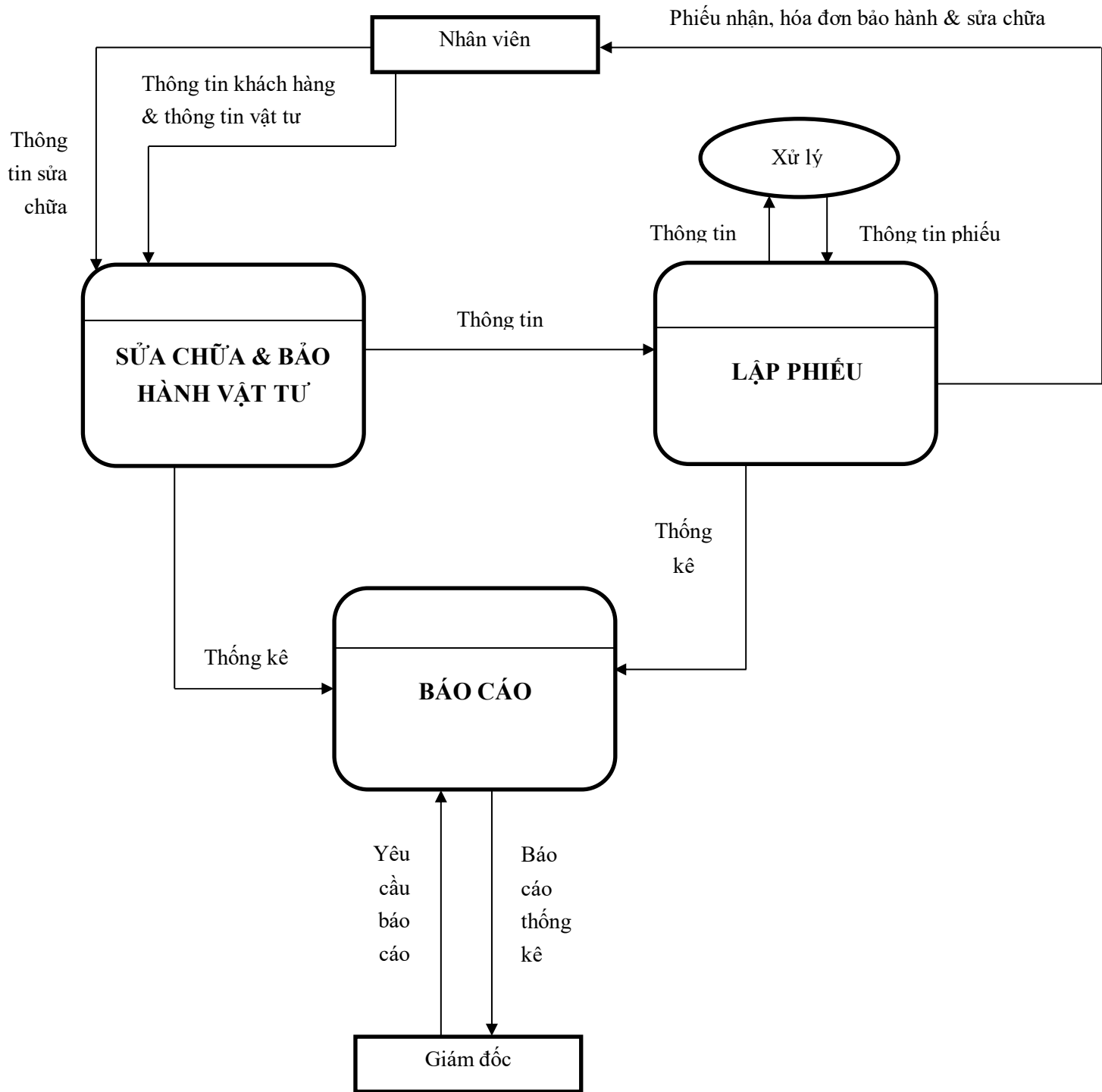
1. Phiếu nhận
2. Phiếu bảo hành và sửa chữa

3.6 Ma trận thực thể dữ liệu

Thực thể dữ liệu		
a. Phiếu nhận		
b. Phiếu bảo hành & sửa chữa		
Chức năng nghiệp vụ	a	b
1. Sửa chữa và bảo hành vật tư		
2. Thống kê bảo hành & sửa chữa		

Hình 3.6 Ma trận thực thể chức năng

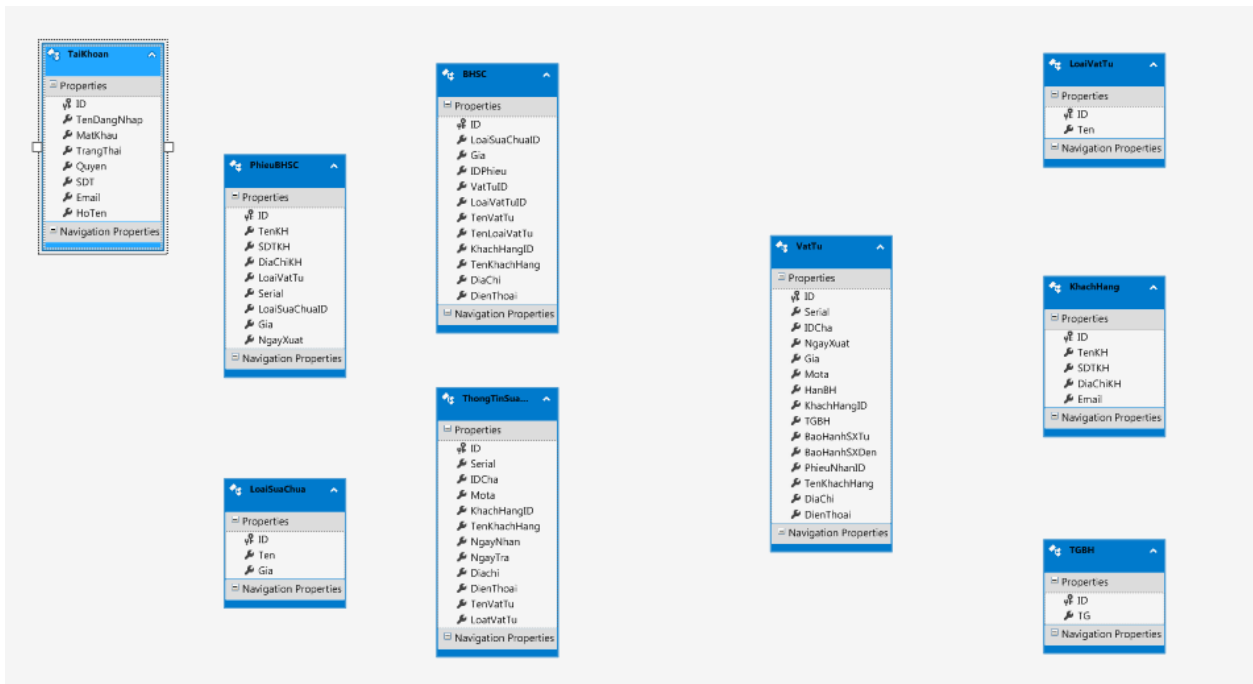
3.7 Sơ đồ luồng dữ liệu



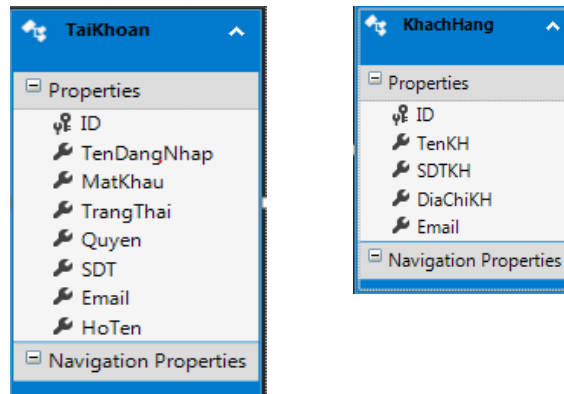
3.7 Sơ đồ luồng dữ liệu

CHƯƠNG 4: CHƯƠNG TRÌNH THỰC NGHIỆM

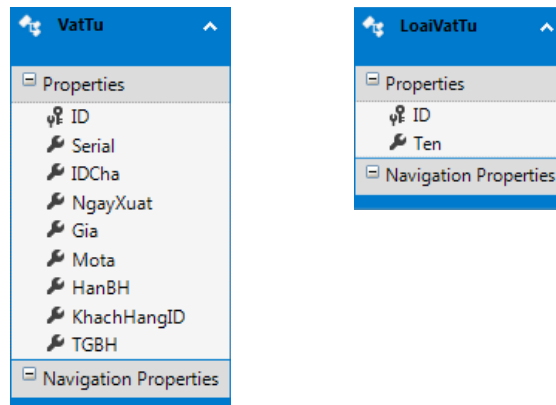
4.1 Thiết kế cơ sở dữ liệu



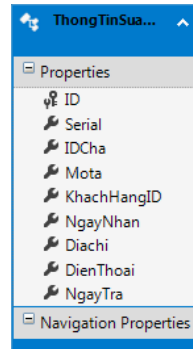
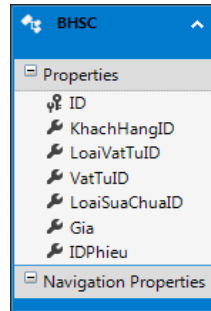
- Bảng Tài Khoản(Quản trị viên) và Bảng thông tin khách hàng :



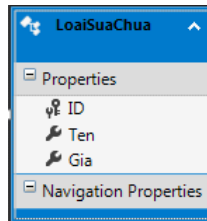
- Bảng vật tư và loại vật tư :



- Bảng Phiếu nhận bảo hành (ThôngTinBaoHanh) và bảng hóa đơn bảo hành, sửa chữa(BHSC).



- Bảng loại sửa chữa :



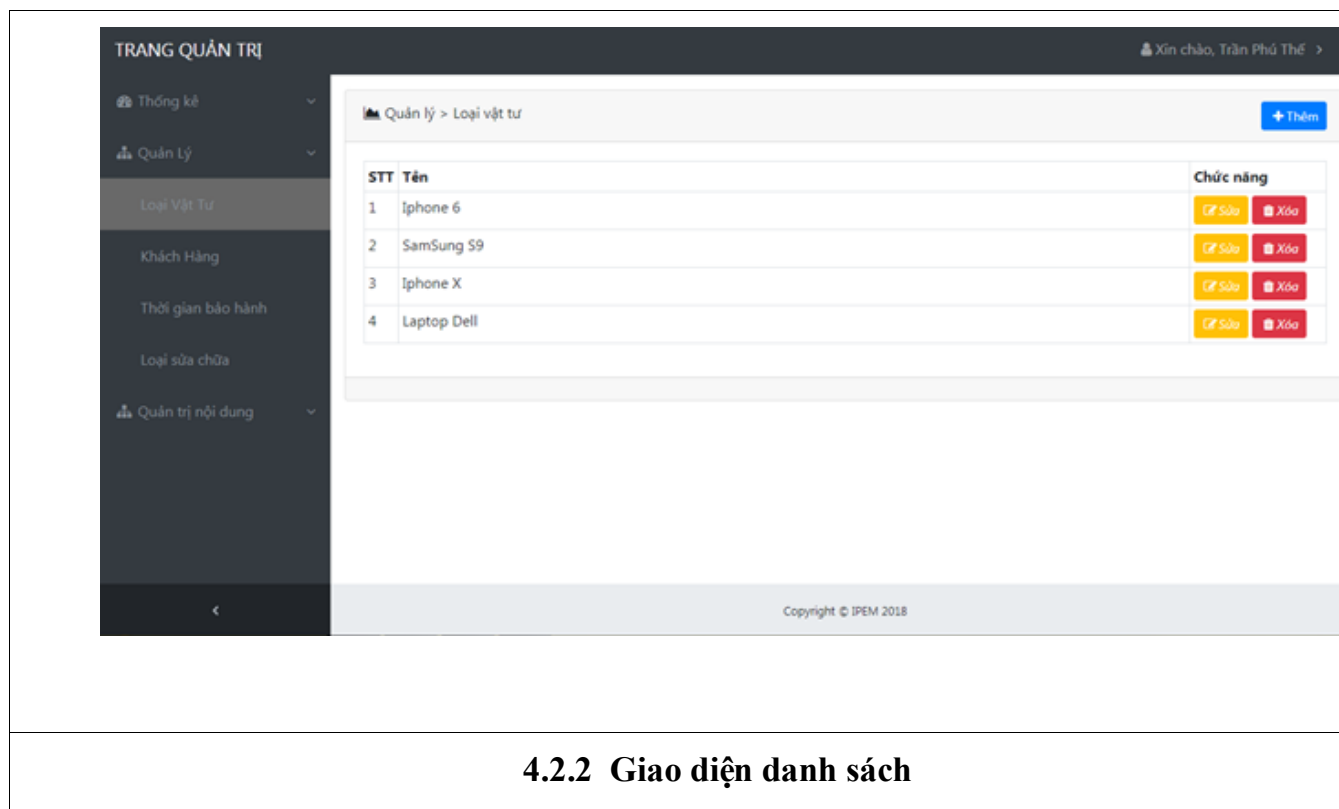
4.2 Một số giao diện

4.2.1 Giao Diện Chính :



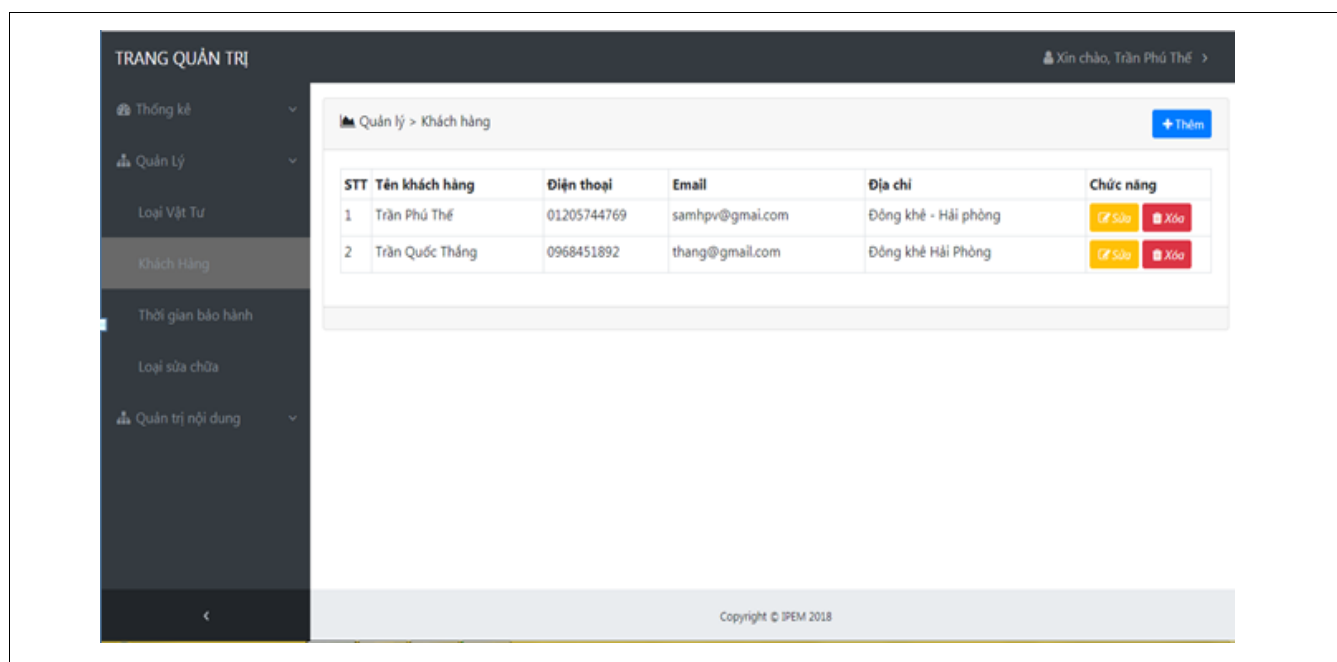
Hình 4.2.1 Giao Diện Trang Chủ

4.2.2 Giao Diện Danh sách :



4.2.2 Giao diện danh sách

4.2.3 Giao Diện thông tin khách hàng đã mua sản phẩm:



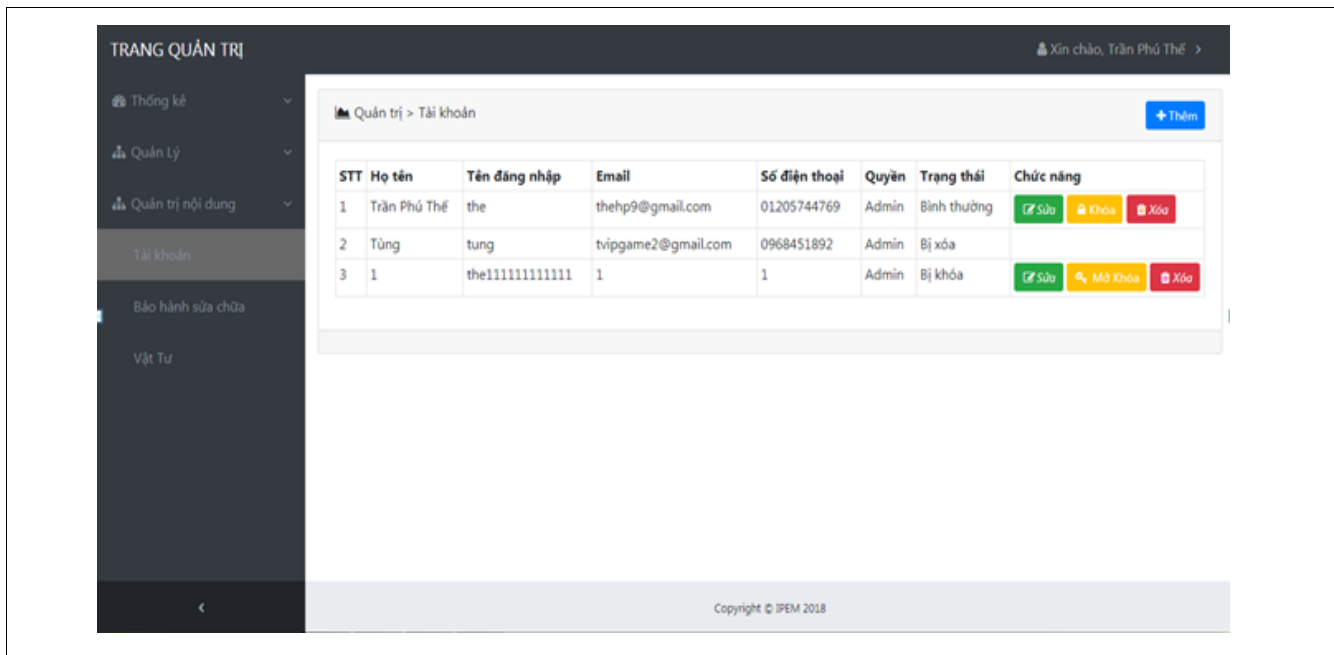
Hình 4.2.3 Giao Diện thông tin khách hàng đã mua sản phẩm

4.2.4 Giao Diện loại sửa chữa :



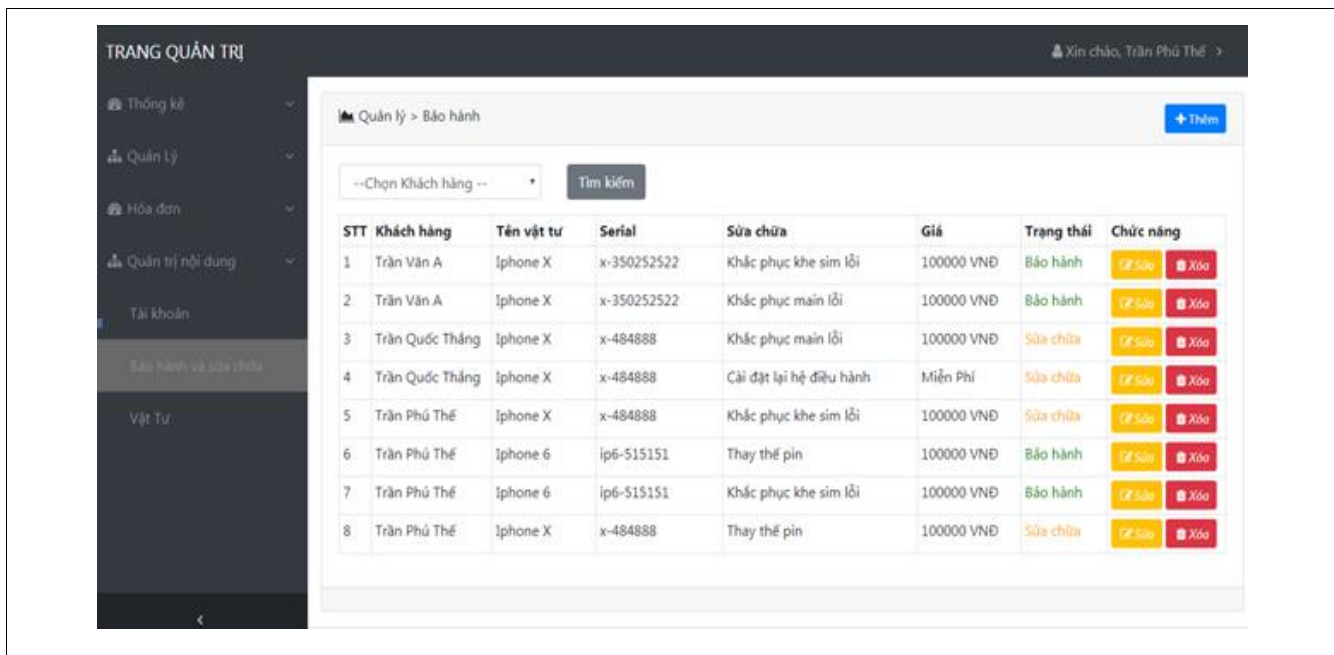
Hình 4.4 Giao Diện loại sửa chữa

4.2.6 Giao Diện Tài Khoản :



Hình 4.6 Giao Diện Tài Khoản

4.2.7 Thông Tin Bảo Hành & Sửa Chữa:



Hình 4.7 Thông Tin Bảo Hành & Sửa Chữa

4.2.6 Giao Diện Hóa Đơn:

TRANG QUẢN TRỊ

- 📊 Thống kê
- 👤 Quản lý
- 📄 Hóa đơn
- 📁 Quản trị nội dung

Xin chào, Trần Phú Thế >

TRUNG TÂM BẢO HÀNH SỬA CHỮA 24H

Địa chỉ: 10/30 Cụm 5 Đông Khê - Ngô Quyền - Hải Phòng
Điện thoại: 01205744769 -- Facebook: Bảo hành 24h Hải Phòng

Ngày xuất: 02/08/2018
Mã số phiếu: 19

PHIẾU NHẬN

Khách hàng:

Địa chỉ: Điện thoại:

Người nhận: Trần Phú Thế

Mô tả chung về lỗi:

Sản phẩm / hàng hóa	Serial	Thời gian nhận
iPhone X	m-215241	02/08/2018 8:55:12 CH

Hạn trả hàng:

TRANG QUẢN TRỊ

- 📊 Thống kê
- 👤 Quản lý
- 📄 Hóa đơn
- 📁 Quản trị nội dung

Xin chào, Trần Phú Thế >

TRUNG TÂM BẢO HÀNH SỬA CHỮA 24H

Địa chỉ: 10/30 Cụm 5 Đông Khê - Ngô Quyền - Hải Phòng
Điện thoại: 01205744769 -- Facebook: Bảo hành 24h Hải Phòng

Ngày xuất: 02/08/2018
Mã số phiếu: 14

PHIẾU NHẬN

Khách hàng: Trần Phú Thế
Địa chỉ: 10/30 Đông Khê - Ngô Quyền - Hải Phòng
Người nhận: Trần Phú Thế

Điện thoại: 0966451892

Mô tả chung về lỗi:
không lên màn hình

Sản phẩm / hàng hóa	Serial	Thời gian nhận
iPhone 5	ip6-61151	15:13 02/08/2018

Hạn trả hàng: 02/08/2018

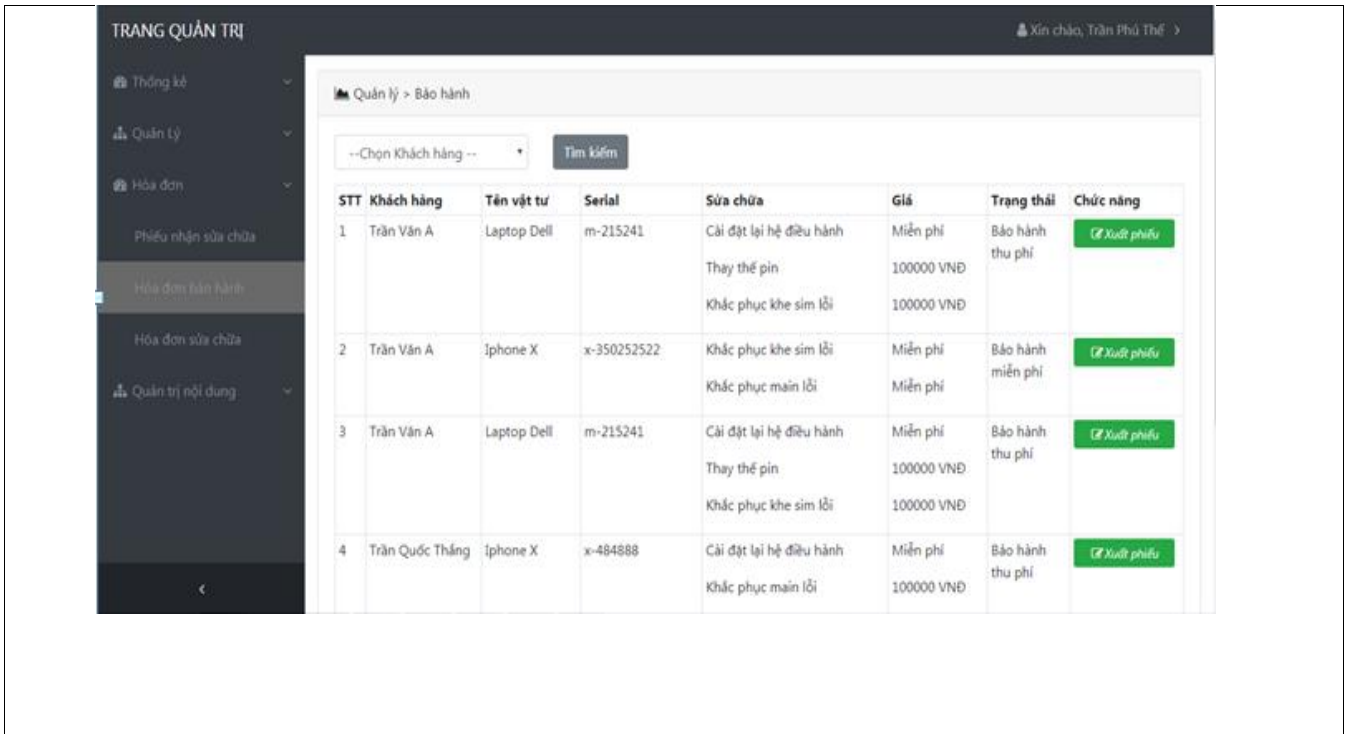
NGƯỜI LẬP PHIẾU

Trần Phú Thế

KỸ THUẬT VIÊN

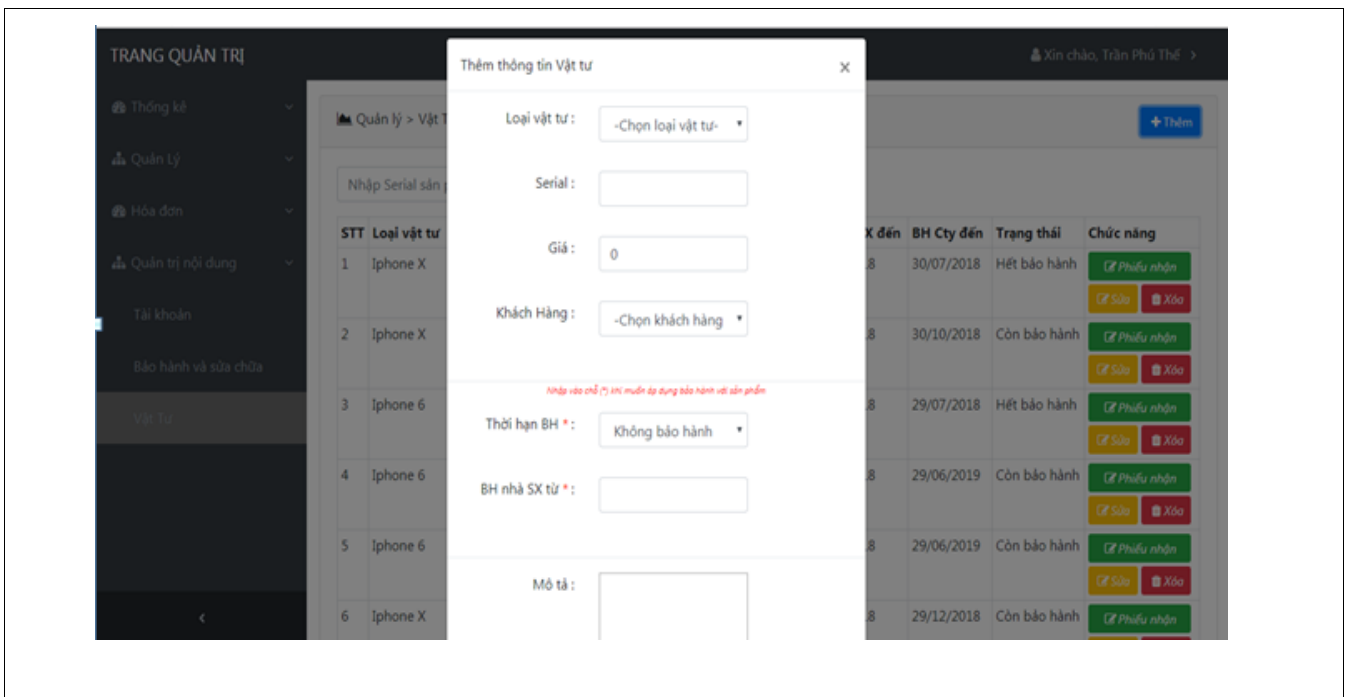
KHÁCH HÀNG

Hải Phòng, Ngày 02, Tháng 08, Năm 2018



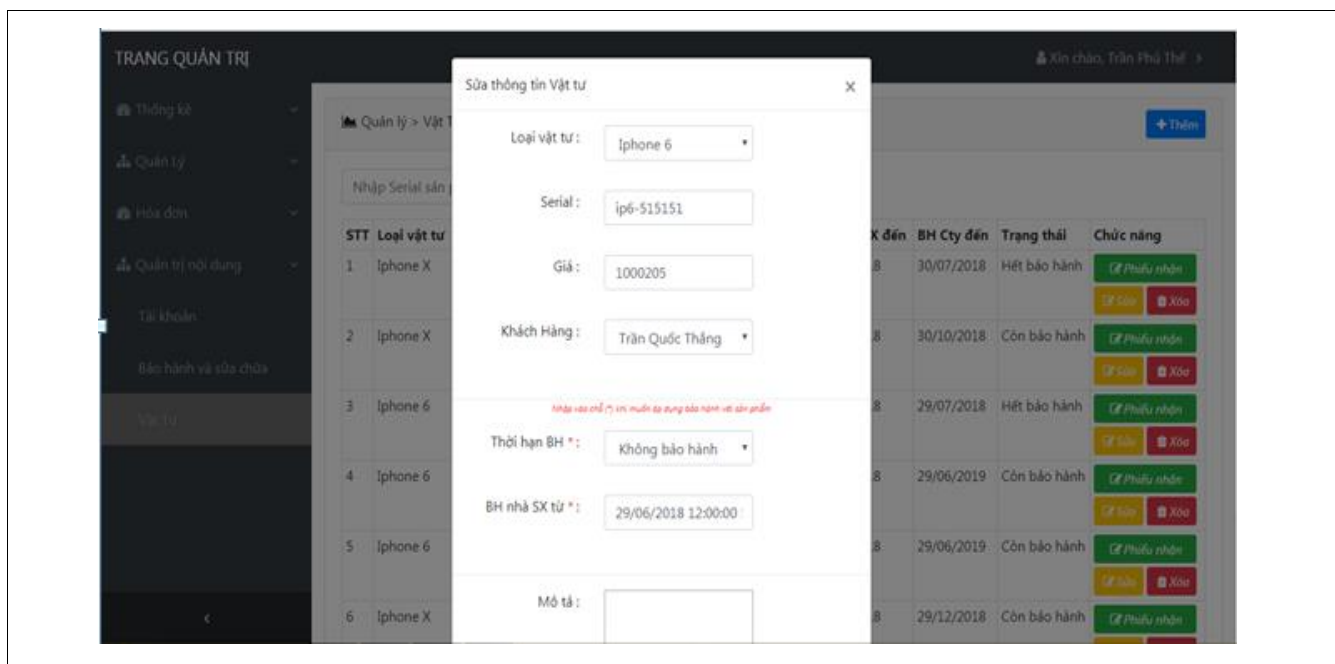
Hình 4.6 Giao Diện Hóa Đơn

4.2.8 Thêm Mới Vật Tư :



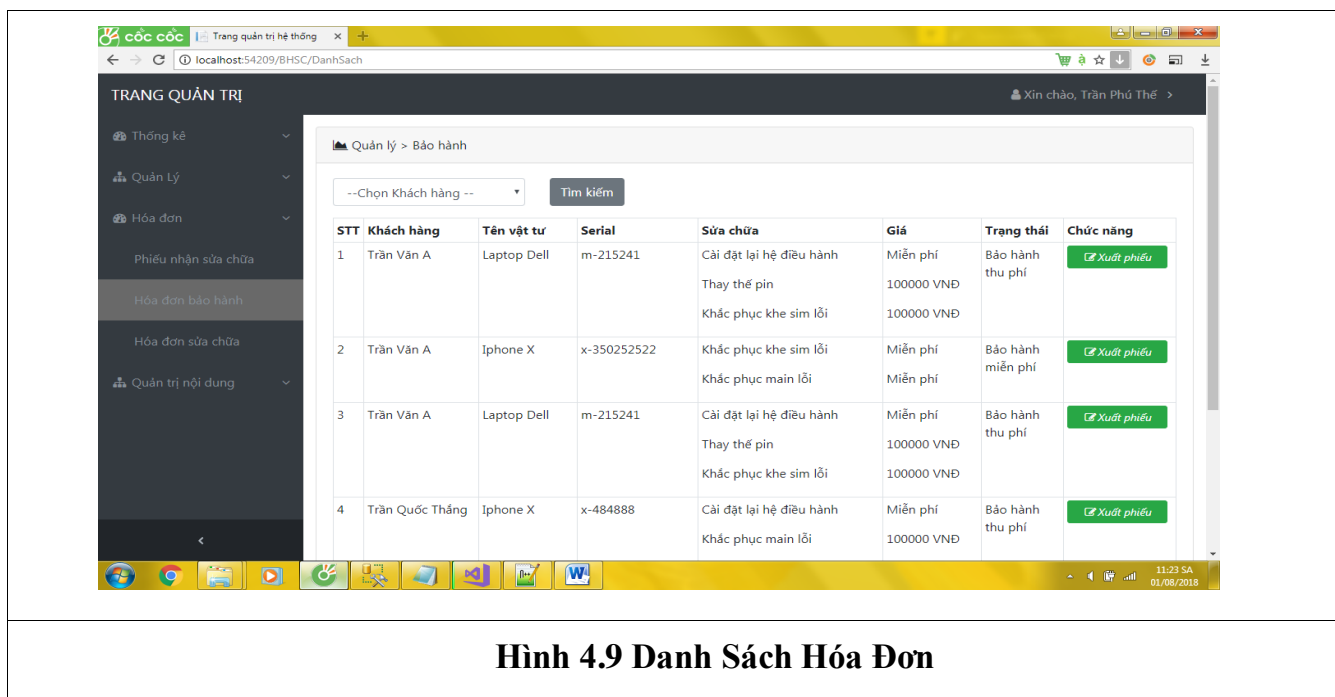
Hình 4.8 Thêm Mới Vật Tư

4.2.8 Chỉnh Sửa Vật Tư :



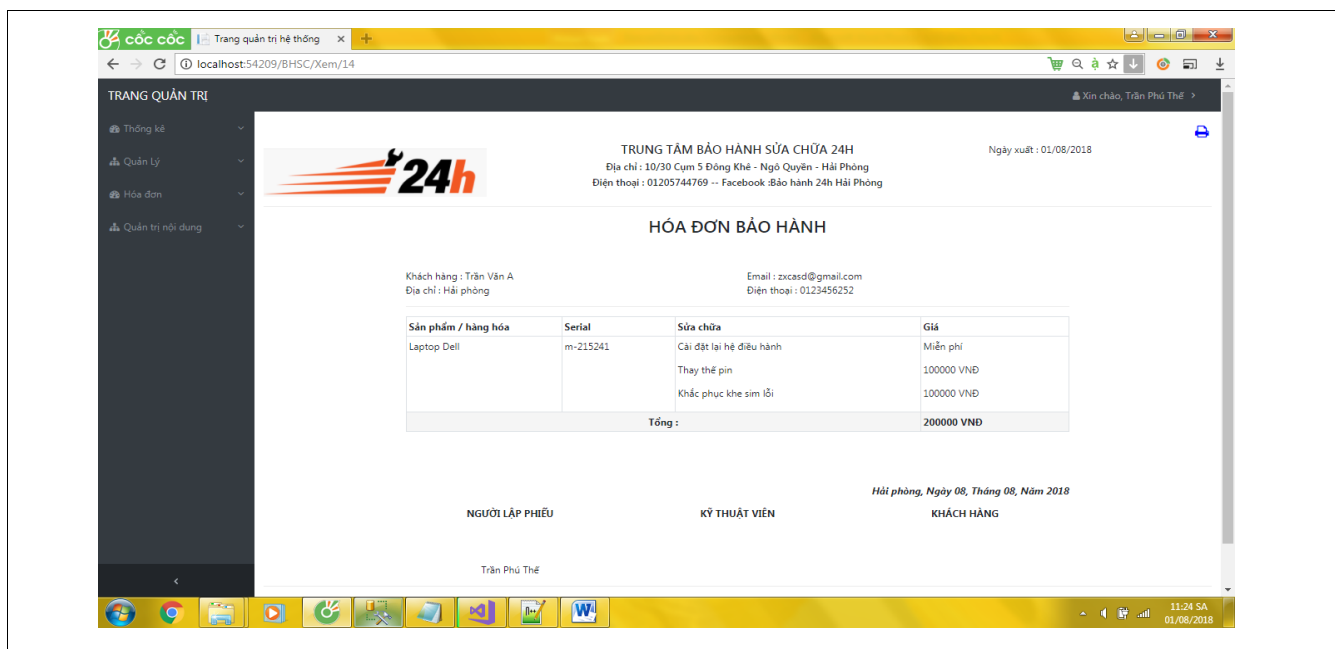
Hình 4.8 Chỉnh Sửa Vật Tư

4.2.9 Danh Sách Hóa Đơn :



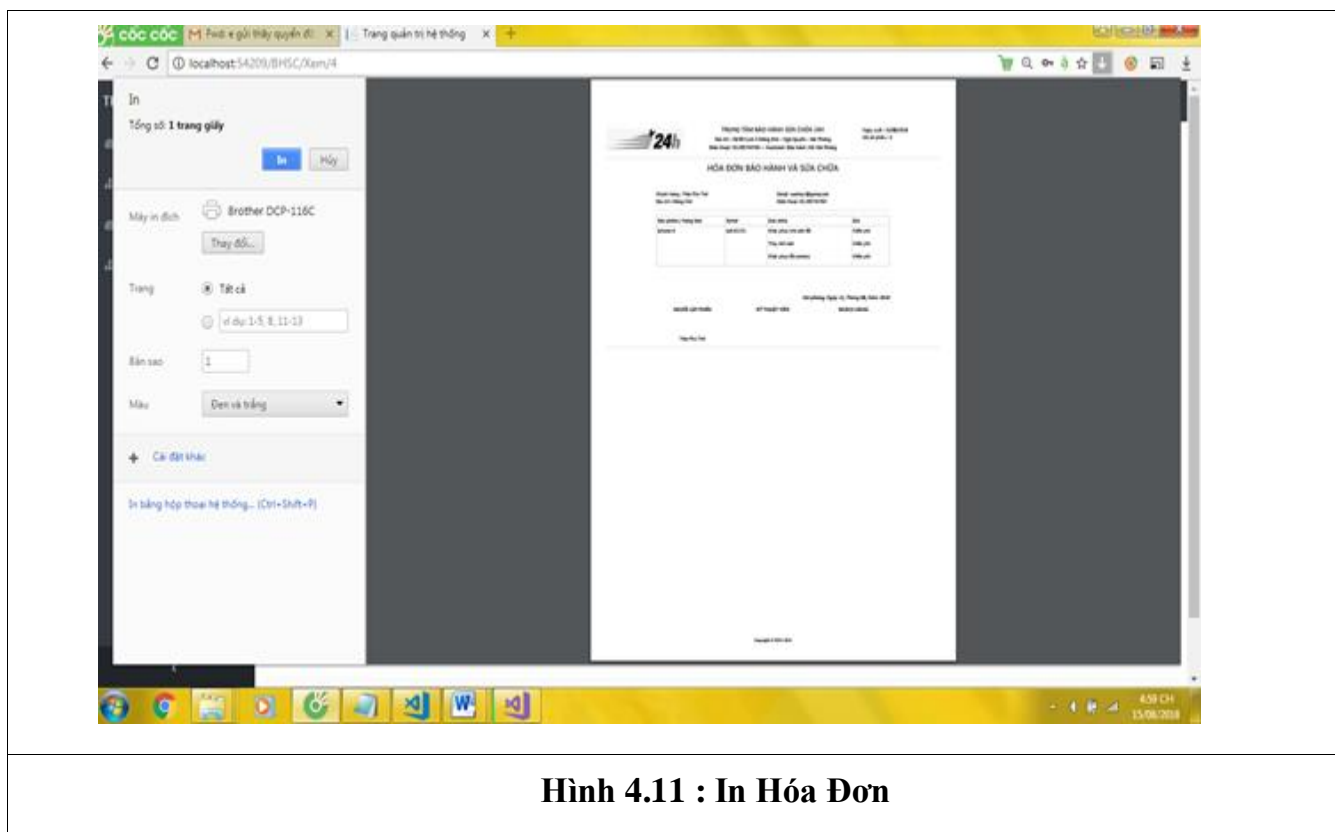
Hình 4.9 Danh Sách Hóa Đơn

4.2.10 Lập Hóa Đơn



Hình 4.10 Lập Hóa Đơn

4.2.11 In Hóa Đơn



Hình 4.11 : In Hóa Đơn

KẾT LUẬN

Trong đồ án này em đã nghiên cứu, tìm hiểu về mô hình webservice, website webservice, ngôn ngữ lập trình MVC5, HTML, kỹ thuật Restful, phần mềm kiểm thử Postman, dữ liệu JSON. Đồng thời em đã nắm bắt được quy trình làm 1 website.

- Lý thuyết:
 - Phát biểu và mô tả được nghiệp vụ bài toán.
 - Biểu diễn các nghiệp vụ bài toán bằng cách các sơ đồ tiến trình nghiệp vụ.
 - Phân tích thiết kế bài toán một cách đầy đủ.
 - Thiết kế được cơ sở dữ liệu quan hệ để lưu trữ dữ liệu.
 - Thiết kế được các lớp xử lý.
- Chương trình:
 - Sử dụng hệ quản trị cơ sở dữ liệu SQL.
 - Sử dụng ngôn ngữ MV5 để xây dựng các lớp.
 - Hệ thống được cài đặt thử nghiệm ban đầu trên localhost, rồi đưa lên hosting cho ra kết quả, đáp ứng được yêu cầu bài toán.

Trong thời gian nghiên cứu, xây dựng chương trình, em đã cố gắng hết sức làm việc với sự giúp đỡ tận tình của thầy giáo hướng dẫn, chương trình đã đạt được kết quả nhất định. Tuy nhiên với thời gian ngắn, trình độ và kinh nghiệm còn hạn chế. Chương trình vẫn còn nhiều thiếu sót. Rất mong các thầy cô giáo và các bạn tận tình giúp đỡ để chương trình ngày càng hoàn thiện hơn.

Nếu được phát triển tiếp đề tài này, em sẽ tiếp tục bổ sung thêm nhiều chức năng khác và phát triển thêm giao diện chương trình để giúp người dùng sử dụng tiện ích hơn.

TÀI LIỆU THAM KHẢO

[1] Liên hệ trực tiếp thầy giáo Đỗ Văn Chiêu

[2] Các trang web w3schools.com, freetuts.net, stackoverflow.com

[3] <https://www.youtube.com/>