**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**



**KIẾN TRÚC PHẦN MỀM**

***Nhóm 04 - Sinh viên thực hiện***

1. Nguyễn Lê Chí Đại – 20065421
2. Lương Trần Gia Bảo – 20068451
3. Nguyễn Công Danh – 20067411

# MỤC LỤC

[CHƯƠNG 1 : GIỚI THIỆU 3](#_Toc14693505)

[1.1 Tổng quan 3](#_Toc14693506)

[1.2 Mục tiêu đề tài 3](#_Toc14693507)

[1.3 Phạm vi đề tài 1](#_Toc14693508)

1.4 Bản phân công công việc 4

[CHƯƠNG 2 : KIẾN TRÚC PHẦN MỀM 6](#_Toc14693510)

[2.1 Xác định yêu cầu của kiến trúc 5](#_Toc14693511)

2.1.1 Mô tả yêu cầu chức năng 6

2.1.2 Yêu cầu phi chức năng 7

2.1.3 Sơ đồ phân cấp chức năng 8

[2.2 Chọn lựa kiến trúc nền tảng 8](#_Toc14693511)

2.2.1 Các mô hình kiến trúc 8

2.2.2 Mô hình MVC 11

2.2.2.1 Tổng quan 11

2.2.2.2 Hiện thực MVC theo Code 13

2.2.2.3 Ưu và nhược điểm 14

2.2.2.4 Tại sao lại sử dụng mô hình MVC cho đề tài 15

[CHƯƠNG 3 : PHÂN TÍCH VÀ THIẾT KẾ 16](#_Toc14693514)

[3.1 Thu thập, làm rõ yêu cầu của hệ thống 16](#_Toc14693515)

[3.2 Phân tích yêu cầu bằng UML 18](#_Toc14693515)

[3.2.1 Usecase tổng quát 18](#_Toc14693516)

[3.2.2 Danh sách tác nhân và mô tả 19](#_Toc14693517)

[3.2.3 Danh sách các tình huống hoạt động (Use cases) 20](#_Toc14693518)

[3.2.4 Tình huống hoạt động 20](#_Toc14693519)

3.2.4.1 Đăng ký thông tin khách hàng 21

3.2.4.2 Đặt lịch 22

3.2.4.3 Nhập thông tin pin mới 25

3.2.4.4 Thực hiện giao dịch thanh toán 27

3.2.4.5 Gửi Pin 29

3.2.4.6 Xác nhận nhận lại lịch đặt 30

3.2.4.7 Kiểm tra pin và lịch đặt 33

3.2.4.8 Báo cáo định kỳ…………………..……….….….…... …... .35

3.2.4.9 Cập nhật pin 37

3.2.4.10 Xóa pin 39

3.2.4.11 Xem giỏ hàng 41

3.2.4.12 Thêm lịch đặt 42

3.2.4.13 Xóa lịch đặt 43

3.2.4.14 Cập nhật lịch đặt 44

[3.3 Class diagram 46](#_Toc14693520)

[3.4 Database diagram 47](#_Toc14693521)

[CHƯƠNG 4 : HIỆN THỰC 48](#_Toc14693522)

[4.1 Cấu hình phần cứng, phần mềm 48](#_Toc14693523)

4.2 Screen Flow: Phân luồng màn hình 49

4.3 Các chức năng chính của các actor 50

[4.4 Giao diện của hệ thống 51](#_Toc14693524)

[4.5 Kế hoạch kiểm thử hệ thống 59](#_Toc14693524)

4.5.1 Những yêu cầu về tài nguyên cho kiểm thử 59

4.5.1.1 Phần cứng 59

4.5.1.2 Phần mềm 59

4.5.2 Danh sách các tình huống kiểm tra 59

[CHƯƠNG 5 : KẾT LUẬN 59](#_Toc14693525)

[5.1 Kết quả đạt được 59](#_Toc14693526)

[5.2 Hạn chế của đồ án 60](#_Toc14693527)

[5.3 Hướng phát triển 60](#_Toc14693528)

# CHƯƠNG I: GIỚI THIỆU

## 1.1 Tổng quan

Môi trường Trái Đất đang phải đối mặt với nhiều vấn đề nghiêm trọng do hoạt động của con người gây ra, bao gồm: Ô nhiễm môi trường, biến đổi khí hậu, suy thoái tài nguyên thiên nhiên, mất đa dạng sinh học.

Bảo vệ môi trường là vấn đề sống còn của con người. Một môi trường ô nhiễm, suy thoái sẽ ảnh hưởng nghiêm trọng đến sức khỏe con người, đời sống kinh tế - xã hội và sự phát triển bền vững của đất nước.

Vì thế thiết kế website thu mua pin cũ để tái chế bảo vệ môi trường là một dự án ý nghĩa, góp phần giải quyết vấn đề môi trường cấp bách hiện nay. Với sự nỗ lực của đội ngũ nhân viên và sự ủng hộ của cộng đồng, chúng tôi tin tưởng rằng website sẽ đạt được những thành công nhất định và góp phần xây dựng một tương lai xanh cho Trái Đất.

## 1.2 Mục tiêu đề tài

Mục tiêu là xây dựng được hệ thống Website giúp thống kê và cho khách hàng có thể đặt chổ online trước khi đem pin đến xử lý. Đối với admin hỗ trợ quản lý thông tin người dùng, thông tin từng loại pin, thống kê thu chi, quản lý danh sách đặt lịch. Đối với người dùng có thể tạo tài khoản quản lý thông tin cá nhân, đặt lịch, xem thông tin và giá các loại pin, xem lịch đặt xử lý pin của mình ...

## 1.3 Phạm vi đề tài

Hệ thống được xây dựng để phục vụ cho 2 đối tượng chính là người dùng và quản trị viên. Người dùng chỉ được sử dụng các chức năng như: xem thông tin hàng, đặt lịch, xem lịch đã đặt,… tất cả người dùng có chức năng giống nhau. Quản trị viên có quyền truy cập, xử lý vào tất cả các tài nguyên của người dùng.

## 1.4 Bản phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
|  | ***Công việc*** | ***Thành viên thực hiện*** | ***Ghi chú*** |
| **Tuần 01**  (Từ: 16/04/2024  Đến: 22/04/2024) | Lập nhóm, làm quen nhóm | Lương Trần Gia Bảo  Nguyễn Lê Chí Đại  Nguyễn Công Danh |  |
| Nhận đề tài | Lương Trần Gia Bảo  Nguyễn Lê Chí Đại  Nguyễn Công Danh |  |
| Phân tích nghiệp vụ của đề tài | Nguyễn Lê Chí Đại |  |
| Liệt kê yêu cầu chức năng và phi chức năng | Lương Trần Gia Bảo  Nguyễn Lê Chí Đại |  |
| Vẽ sơ dồ phân cấp | Nguyễn Công Danh |  |
| **Tuần 02**  (Từ: 23/04/2014  Đến: 29/04/2014) | Vẽ sơ đồ use case  Đặc tả use case | Nguyễn Lê Chí Đại |  |
| Vễ sơ đồ activity | Lương Trần Gia Bảo |  |
| Vẽ sơ dồ sequence | Nguyễn Công Danh |  |
| **Tuần 03**  (Từ: 30/04/2014  Đến: 05/05/2014) | Vẽ sơ đồ class | Nguyễn Lê Chí Đại |  |
| Đặc tả class | Nguyễn Lê Chí Đại |
| Thiết kế CSDL | Lương Trần Gia Bảo |  |
| Vẽ lược đồ CSDL | Nguyễn Công Danh |  |
| Thiết kế sơ lược màn hình | Lương Trần Gia Bảo  Nguyễn Công Danh |  |
| **Tuần 04**  (Từ: 06/05/2014  Đến: 12/05/2014) | Hiện thực ứng dụng | Lương Trần Gia Bảo  Nguyễn Lê Chí Đại |  |
| Kiểm thử ứng dụng | Lương Trần Gia Bảo  Nguyễn Công Danh |  |
| Tìm lỗi | Nguyễn Lê Chí Đại  Nguyễn Công Danh |  |
| Sửa lỗi | Lương Trần Gia Bảo  Nguyễn Lê Chí Đại |  |
| **Tuần 05**  (Từ: 13/05/2014  Đến: 20/05/2014) | Tiếp tục sửa lỗi (nếu có) | Lương Trần Gia Bảo  Nguyễn Lê Chí Đại  Nguyễn Công Danh |  |
| Hoàn thành tài liệu | Nguyễn Lê Chí Đại |  |

# CHƯƠNG II: KIẾN TRÚC PHẦN MỀM

## 2.1 Xác định yêu cầu của kiến trúc

### 2.1.1 Mô tả yêu cầu chức năng

Nhằm hưởng ứng phong trào thu gom pin cũ đã qua xử dụng để tái chế. Công ty DBD có tạo một trang web phục vụ cho nhu cầu đặt lịch xử lý pin như sau:

Người dùng vãng lai có thể lên website để tìm loại pin mà mình muốn đổi để xem chi tiết về giá cả trao đổi pin (vd: pin con ó đỏ 20 viên = 5000 vnd).

Khi người dùng chọn được loại pin rồi có thể đặt lịch để xử lý pin. Người dùng điền các thông tin cần thiết để đặt lịch (vd: tên, số dt, địa chỉ, chi nhánh…..), trước khi đặt lịch khách hàng phải đăng nhập để có thể xem thông tin đặt lịch của mình ở giỏ hàng và nhận thông báo về lịch hẹn xử lý pin.

Với trường hợp đã đặt lịch: khách hàng đem pin đến quầy và đọc tên hoặc số điện thoại đã đặt lịch cho nhân viên. Nhân viên kiểm tra số lượng pin mà người dùng đem đến có đúng với lịch xử lý hay không nếu đúng xẽ xác nhận *(nhân viên đăng nhập admin để có thể xem lịch và xác nhận thông tin).* Sau khi xác nhận số lượng pin trong kho sẽ tự động cập nhật lại (số lượng pin trong kho = pin trong kho + pin khách hàng đem đến).

Với trường hợp khách hàng vãng lai (không đặt lịch), nhận viên sẽ kiểm tra số lượng pin mà khách hàng đem đến. *(đăng nhập admin)* điền thông tin khách hàng( tên, số dt, địa chỉ ….. và các thông tin số lượng pin…) sau đó lưu khách hàng mới này. Sau khi xác nhận số lượng pin trong kho sẽ cập nhật lại (số lượng pin trong kho = pin trong kho + pin khách hàng đem đến).

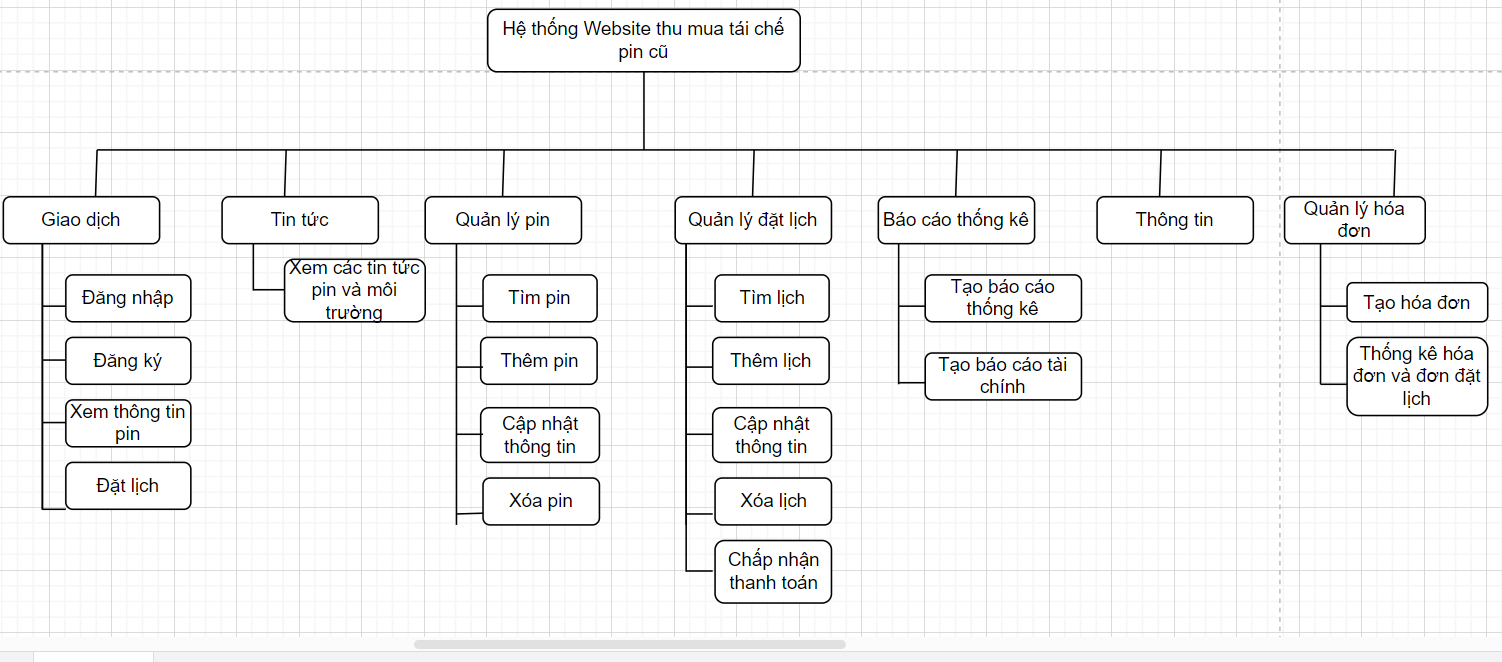
Các chức năng thêm sửa xóa update để quản lý pin, lịch và khách hàng sẽ được thực hiện dưới quyền admin.

Có thể xem lại thống kê số lượng pin trong kho đã thu gom,tổng chi phí đã chi để lập thống kê.

### 2.1.2 Yêu cầu phi chức năng

|  |  |
| --- | --- |
| Hiệu suất | Ứng dụng hoạt động mượt mà, ổn định. |
| Bảo mật | Đảm bảo an toàn thông tin cho người dùng, bao gồm bảo mật tài khoản, bảo mật dữ liệu |
| Khả năng mở rộng | Dễ dàng mở rộng thêm chức năng mới trong tương lai |
| Khả năng tương thích | Hoạt động tốt trên các trình duyệt web và hệ điều hành phổ biến |
| Thiết kế giao diện | Giao diện thân thiện, dễ sử dụng, phù hợp |

### 2.1.3 Sơ dồ phân cấp chức năng

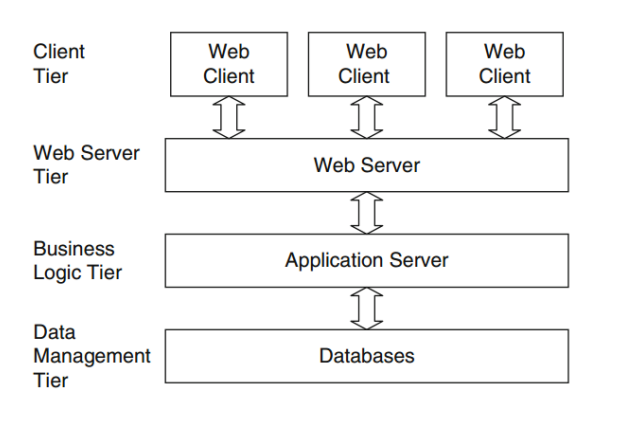


## 2.2 Chọn lựa kiến trúc nền tảng

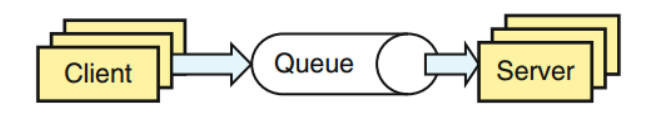
### 2.2.1 Các mô hình kiến trúc

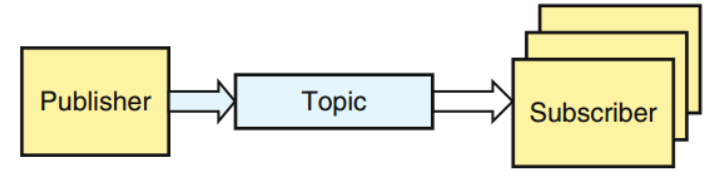
Trong lập trình phần mềm, có rất nhiều mô hình kiến trúc khác nhau được sử dụng để thiết kế và xây dựng các hệ thống phần mềm. Một số mô hình kiến trúc phổ biến nhất bao gồm:

* Multitier Client Server

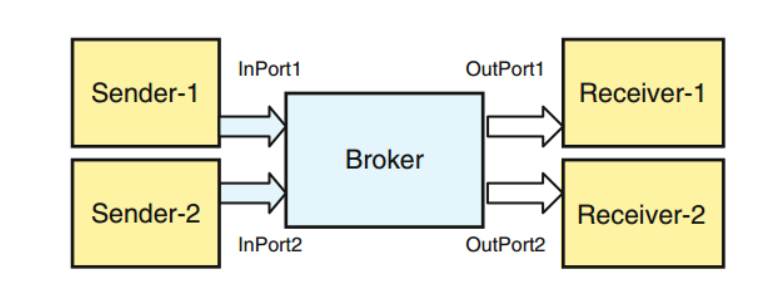


* Messaging

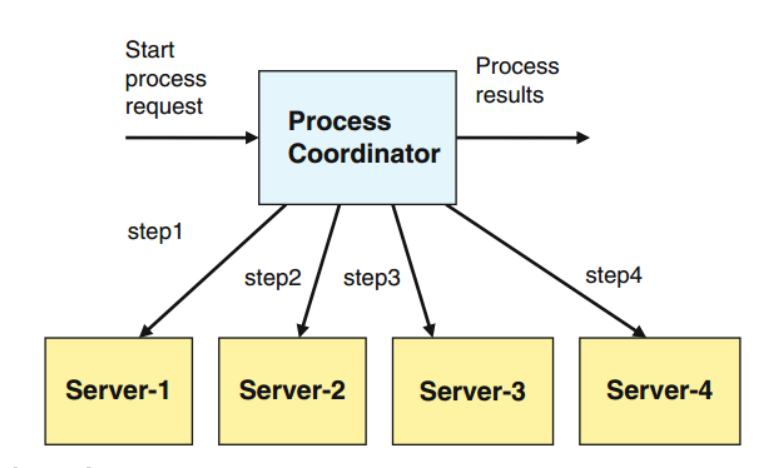




* Broker



* Process Coordinator



* Các kiểu kiến trúc khác: Model-View-Controller, Repository architecture, Client–server architecture, Pipe and filter architecture,…

### 2.2.2 Mô hình MVC

#### 2.2.2.1 Tổng quan

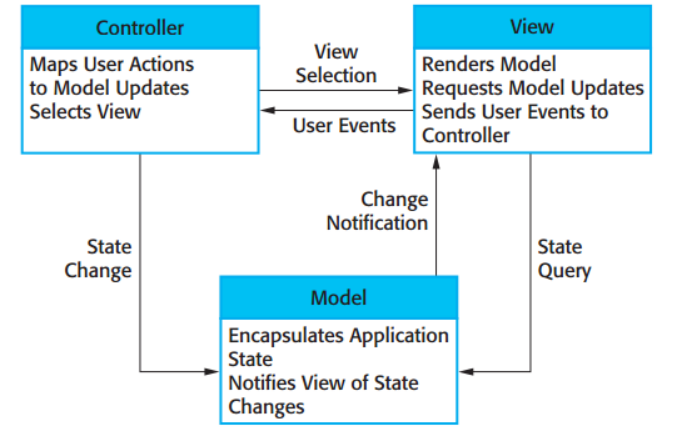
Mô hình Model-View-Controller (MVC) là một mẫu kiến ​​trúc phân tách một ứng dụng thành ba thành phần logic chính Model, View và Controller. Do đó viết tắt MVC. Mỗi thành phần kiến ​​trúc được xây dựng để xử lý khía cạnh phát triển cụ thể của một ứng dụng. MVC tách lớp logic nghiệp vụ và lớp hiển thị ra riêng biệt. Ngày nay, kiến ​​trúc MVC đã trở nên phổ biến để thiết kế các ứng dụng web cũng như ứng dụng di động.

Kiến trúc MVC được thảo luận lần đầu vào năm 1979 bởi Trygve Reenskaug.

Mô hình MVC được giới thiệu lần đầu tiên vào năm 1987 bằng ngôn ngữ lập trình Smalltalk.

MVC lần đầu tiên được chấp nhận như một khái niệm chung, trong một bài báo năm 1988.

Trong thời gian gần đây, MVC pattern được sử dụng rộng rãi trong các ứng dụng web hiện đại.



**MVC quan trọng bao gồm:**

**Model**: Nó bao gồm tất cả dữ liệu và logic liên quan của nó.

**View**: Trình bày dữ liệu cho người dùng hoặc xử lý tương tác của người dùng.

**Controller**: Là phần quan trọng nhất trong mô hình, nó liên kết phần Model và View.

**View**

View là một phần của ứng dụng đại diện cho việc trình bày dữ liệu.

View được tạo bởi các dữ liệu mà chúng ta lấy từ dữ liệu trong model. Một view yêu cầu model cung cấp đầy đủ dữ liệu để nó hiển thị đầu ra cho người dùng.

View chính là nới chứa những giao diện như một nút bấm, khung nhập, menu, hình ảnh… nó đảm nhiệm nhiệm vụ hiển thị dữ liệu và giúp người dùng tương tác với hệ thống.

**Controller**

Controller là một phần của ứng dụng xử lý tương tác của người dùng. Bộ điều khiển diễn giải đầu vào chuột và bàn phím từ người dùng, thông báo cho model và view để thay đổi khi thích hợp.

Controller là nới tiếp nhận những yêu cầu xử lý được gửi từ người dùng, nó sẽ gồm những class/ function xử lý nhiều nghiệp vụ logic giúp lấy đúng dữ liệu thông tin cần thiết nhờ các nghiệp vụ lớp Model cung cấp và hiển thị dữ liệu đó ra cho người dùng nhờ lớp View.

Controller gửi các lệnh đến model để làm thay đổi trạng thái của nó (Ví dụ: ta thêm mới 1 pin hoặc cập nhật tên 1 pin). Controller cũng gửi các lệnh đến view liên quan của nó để thay đổi cách hiển thị của view (Ví dụ: xem thông tin 1 pin).

**Model**

Thành phần model lưu trữ dữ liệu và logic liên quan của nó. Bao gồm các class function xử lý các tác vụ như truy vấn, thêm, sửa hoặc xóa dữ liệu. Ví dụ, một đối tượng Controller sẽ lấy thông tin khách hàng từ cơ sở dữ liệu. Nó thao tác dữ liệu và gửi trở lại cơ sở dữ liệu hoặc sử dụng nó để hiển thị dữ liệu.

Sự tương tác giữa các thành phần

Controller tương tác với qua lại với View.

Controller tương tác qua lại với Model.

Model và View không có sự tương tác với nhau trực tiếp mà nó tương tác với nhau thông qua Controller.

**Ví dụ cho sự tương tác**: Khi người dùng ấn đăng nhập từ view thì request sẽ được gửi từ trình duyệt đến controller, controller sẽ gọi đến model xử lý logic và trả lại kết quả đó cho user thông qua view .

#### 2.2.2.2 Hiện thực MVC theo code

**A diagram of a computer program

Description automatically generated**

Client (Browser):

* Người dùng tương tác với ứng dụng thông qua trình duyệt web. Các yêu cầu từ trình duyệt được gửi tới các servlet.

Controller (Servlet):

* Các lớp Servlet như: *( CRUDDatLichControll.java, CRUDServlet.java, LoginControll.java, LogoutControll.java, PinServlet.java, và SignUpControll.java)* nhận các yêu cầu từ người dùng, xử lý logic và tương tác với Model để lấy dữ liệu hoặc thực hiện các thao tác cần thiết.

Model (DAO, Entity):

* Các lớp trong gói entity chứa các đối tượng dữ liệu.
* Các lớp trong gói model như (DatLichModel.java, PinModel.java, và ThuongHieuModel.java )đại diện cho các thực thể dữ liệu và chứa logic nghiệp vụ.
* ConnectDB là lớp tiện ích kết nối cơ sở dữ liệu phục phụ cho việc đăng ký và đăng nhập

View (JSP, HTML, CSS):

* Các trang JSP như ( add-pin-form.jsp và list-pin.jsp) cung cấp giao diện người dùng. Dữ liệu từ các servlet được gửi tới các trang JSP này để hiển thị cho người dùng.

Database (SQL):

* Cơ sở dữ liệu lưu trữ dữ liệu của ứng dụng, được truy xuất và cập nhật bởi các lớp trong Model.

#### 2.2.2.3 Ưu và nhược điểm

**Ưu điểm của MVC:**

* Bảo trì code dễ dàng, dễ dàng mở rộng và phát triển.
* Hỗ trợ dễ dàng hơn cho khách hàng mới.
* Việc phát triển các thành phần khác nhau có thể được thực hiện song song.
* Nó giúp bạn tránh sự phức tạp bằng cách chia ứng dụng thành ba đơn vị Model, View và Controller.
* Cung cấp hỗ trợ tốt nhất cho phát triển theo hướng thử nghiệm.
* Nó hoạt động tốt cho các ứng dụng Web được hỗ trợ bởi các nhóm lớn các nhà thiết kế và phát triển web.
* Cung cấp khả năng phân tách rõ ràng các mối quan tâm.
* Thân thiện với Công cụ Tìm kiếm (SEO).
* Tất cả các đối tượng được phân loại và đối tượng độc lập với nhau để bạn có thể kiểm tra chúng một cách riêng biệt.

**Nhược điểm của việc sử dụng MVC:**

* Khó đọc, thay đổi, kiểm tra và sử dụng lại mô hình này.
* Không có hỗ trợ xác thực chính thức.
* Tăng độ phức tạp và tính kém hiệu quả của dữ liệu.
* Khó khăn khi sử dụng MVC với giao diện người dùng hiện đại.
* Cần có nhiều người lập trình để tiến hành lập trình song song.
* Cần có kiến ​​thức về nhiều công nghệ.
* Bảo trì nhiều code trong Controller.

#### 2.2.2.4 Tại sao lại sử dụng mô hình MVC cho đề tài ?

Sử dụng mô hình MVC cho thiết kế website thu mua pin để tái chế mang lại nhiều lợi ích, bao gồm:

**1. Tách biệt các mối quan tâm:**

Mô hình MVC tách biệt logic ứng dụng (Model), giao diện người dùng (View) và logic điều khiển (Controller) thành ba phần riêng biệt. Điều này giúp cho việc phát triển và bảo trì website dễ dàng hơn, vì các nhà phát triển có thể làm việc trên từng phần độc lập mà không ảnh hưởng đến các phần khác.

**2. Khả năng tái sử dụng:**

Các thành phần Model, View và Controller trong mô hình MVC có thể được tái sử dụng trong các phần khác nhau của website hoặc thậm chí trong các dự án khác. Điều này giúp tiết kiệm thời gian và công sức phát triển.

**3. Dễ dàng kiểm thử:**

Việc tách biệt các mối quan tâm trong mô hình MVC giúp cho việc viết và thực thi các bài kiểm thử đơn vị dễ dàng hơn. Điều này giúp đảm bảo chất lượng cao hơn cho website.

**4. Dễ dàng bảo trì:**

Với việc tách biệt các mối quan tâm, việc sửa lỗi và bảo trì website cũng trở nên dễ dàng hơn. Các nhà phát triển có thể dễ dàng xác định và sửa lỗi trong một phần cụ thể của website mà không cần phải ảnh hưởng đến các phần khác.

**5. Phù hợp với website thu mua Pin tái chế:**

Mô hình MVC đặc biệt phù hợp cho website thu mua pin tái chế vì nó cho phép website dễ dàng quản lý các loại pin khác nhau, hiển thị thông tin chi tiết về từng pin và xử lý các yêu cầu đặt lịch đổi pin từ khách hàng.

**Ví dụ:**

Model: Model bao gồm các lớp giúp dể dàng biểu diễn và thực hiện gọi hàm vd: (ở model pin có thể dể dàng xem các hàm liên quan đến xử lý thêm sửa xóa pin…).

View: View có thể bao gồm các trang HTML để hiển thị danh Pin có sẳn và hiện đang thu mua trao đổi để khách hàng lựa chọn, thông tin chi tiết về loại pin và biểu mẫu để khách hàng có thể đặt lịch đến để đổi pin.

Controller: Controller có thể bao gồm các lớp để xử lý các yêu cầu từ khách hàng, chẳng hạn như yêu cầu hiển thị danh sách pin, yêu cầu xem chi tiết pin và yêu cầu gửi yêu cầu đặt lich...

Sử dụng mô hình MVC có thể giúp bạn tạo ra một website thu mua pin, hiệu quả và dễ bảo trì.

**Ngoài những lợi ích trên, mô hình MVC còn có một số ưu điểm khác, chẳng hạn như:**

Cải thiện hiệu suất: Mô hình MVC có thể giúp cải thiện hiệu suất website bằng cách tách biệt logic truy cập dữ liệu khỏi logic hiển thị.

Tăng cường bảo mật: Mô hình MVC có thể giúp tăng cường bảo mật website bằng cách tách biệt logic ứng dụng khỏi giao diện người dùng.

# CHƯƠNG III: PHÂN TÍCH VÀ THIẾT KẾ

## 3.1 Thu thập làm rõ yêu cầu của hệ thống

|  |  |  |  |
| --- | --- | --- | --- |
| ***STT*** | ***Câu hỏi (Questions)*** | ***Trả lời (Answers)*** | ***Ghi chú*** |
| 1. | Ứng dụng có cần phân quyền không? | Ứng dụng cần phân quyền cho nhân viên(thu mua, quản lý): tìm kiếm thông tin các loại pin (pin AAA, pin con ó.., ),  Nhân viên quản lý phải kiểm tra được về các thông tin pin.  Nhân viên có quyền “chấp nhận” đơn đặt lịch từ khách hàng |  |
| 2. | Mật khẩu có cần điều kiện gì không? | Mật khẩu cần tối thiều 6 ký tự cả chữ lẫn số và có tối thiểu 1 ký tự đặc biệt. |  |
| 3. | Chương trình có bao nhiêu loại người sử dụng? | Có các loại người dùng như sau: nhân viên quản lý, nhân viên bán hàng, khách hàng. |  |
| 4. | Ngày thu hàng có cần ràng buộc hay không? | Ngày thu hàng được cập nhật và được hiện hữu sẵn trên hệ thống. |  |
| 5. | Người sử dụng có hay dùng phím Enter khi nhập dữ liệu không? | Có. | Vừa sử dụng Enter, vừa sử dụng chuột khi nhập dữ liệu. |
| 6. |  |  |  |
| 7. |  |  |  |
| 8. |  |  |  |
| … |  |  |  |
| … |  |  |  |

## 3.2 Phân tích yêu cầu bằng UML

### 3.2.1 Use case tổng quát



|  |  |  |
| --- | --- | --- |
| **Use case tổng quát** | | |
| Mục đích: | Quản lý toàn bộ quy trình từ khi khách hàng yêu cầu đặt lịch cho đến khi thu gom pin hoàn tất. | |
| Mô tả: | Khách hàng yêu cầu đặt lịch xử lý pin, trong khi nhân viên công ty xử lý các yêu cầu và thực hiện quy trình tái chế, thu mua. | |
| Tác nhân: | Khách hàng và nhân viên công ty tái chế. | |
| Điều kiện trước: | Khách hàng đã đăng nhập vào hệ thống và có thông tin liên lạc rõ ràng. | |
| Điều kiện sau: | Pin được thu gom hoàn tất và khách hàng được thông báo về kết quả. | |
| Luồng sự kiện chính (Basic flows) | Khách hàng | Hệ thống |
| Khách hàng đặt lịch và gửi thông tin đặt lịch đến công ty qua trang web |  |
|  | Công ty xác nhận đơn đặt lịch và gọi điện để xác nhận |
| Khách hàng đến công ty để đổi pin |  |
|  | Thanh toán cho khách hàng nếu pin và số lượng đúng với lịch đặt |
|  | Cập nhật lại số lượng pin và chi phí. |
|  | Thực hiện quy trình tái chế và tạo báo cáo định kỳ. |
|  |  |
| Luồng sự kiện phụ (Alternative Flows): |  | Các sự cố phát sinh như thông tin không đầy đủ, khách hàng đem pin không đúng với lịch đặt từ trước, khách hàng không đến điểm thu gom pin cũ. |
|  |  | |

### 3.2.2 Danh sách các tác nhân mô tả

|  |  |
| --- | --- |
| Tác nhân | Mô tả tác nhân |
| Khách hàng | Là các cá nhân , tổ chức hay một nhiều nhóm người muốn sử dụng các dịch vụ của cửa hàng. Họ là người đưa ra các quyết định có sử dụng các dịch vụ của cửa hàng hay không. |
| Nhân Viên | Là một hoặc nhiều cá nhân được thuê bởi một người hay một doanh nghiệp để sử dụng sức lao động của họ để làm một công việc cụ thể. |
| Quản lí | Là người điều hành tất cả các hoạt động của cửa hàng trên ứng dụng. |

### 3.2.3 Danh sách các tình huống hoạt động (Use cases)

|  |  |
| --- | --- |
| ID | Tên Use case |
| UC01 | Đăng Ký Thông Tin Khách Hàng |
| UC02 | Đặt lịch |
| UC03 | Nhập thông tin pin mới |
| UC04 | Thực hiện giao dịch thanh toán |
| UC05 | Gửi pin |
| UC06 | Xác nhận lại lịch đặt |
| UC07 | Kiểm tra pin với lịch đặt |
| UC08 | Báo cáo định kỳ |
| UC09 | Cập nhật pin |
| UC010 | Xóa pin |
| UC011 | Xem giỏ hàng |
| UC012 | Thêm lịch đặt |
| UC013 | Xóa lịch đặt |
| UC014 | Cập nhật lịch đặt |
|  |  |

### 3.2.4 Tình huống hoạt động

#### 3.2.4.1 Đăng ký thông tin khách hàng

**Use Case: Đăng Ký Thông Tin Khách Hàng**

• Mã Use Case: UC001

• Mục đích: Cho phép người dùng mới đăng ký thông tin cá nhân để sử dụng dịch vụ.

• Tác nhân chính: Khách hàng mới

• Luồng sự kiện chính:

1. Khách hàng truy cập vào hệ thống.

2. Chọn tùy chọn đăng ký mới.

3. Nhập thông tin cá nhân bao gồm tên, địa chỉ email, số điện thoại và địa chỉ.

4. Xác nhận thông tin và hoàn tất đăng ký.

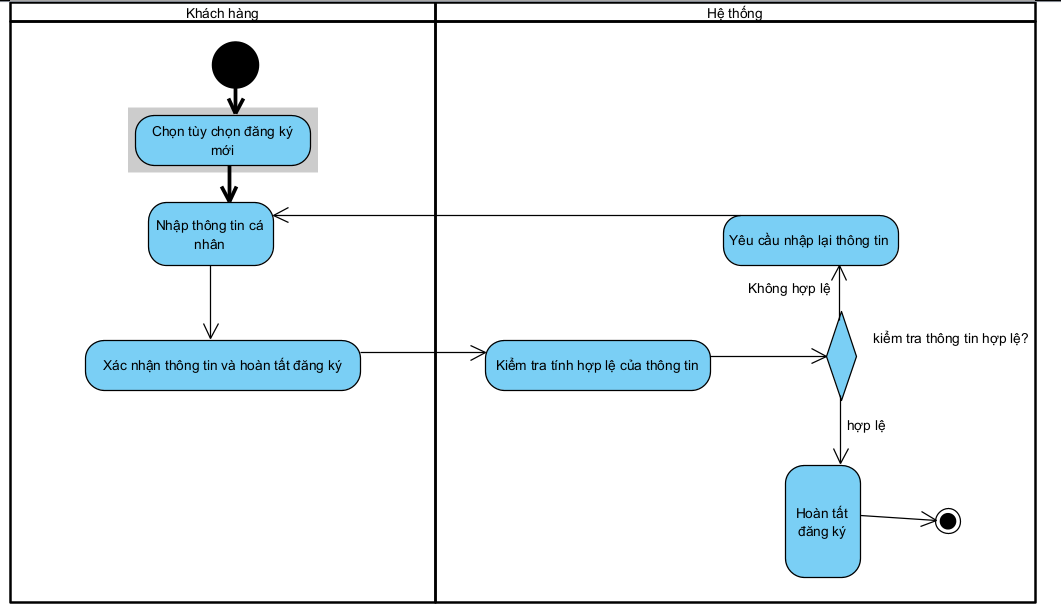
• Luồng sự kiện thay thế:

o Nếu khách hàng nhập thông tin không hợp lệ, hệ thống sẽ thông báo và yêu cầu nhập lại.

• Điều kiện trước: Khách hàng chưa có tài khoản trong hệ thống.

• Điều kiện sau: Khách hàng có tài khoản và có thể sử dụng dịch vụ.

**Sơ đồ Activity đăng ký thông tin khách hàng:**

****

**Sơ đồ sequence đăng ký thông tin khách hàng:**

**A diagram of a project

Description automatically generated**

#### 3.2.4.2 Đặt lịch

**Use Case: Đặt lịch**

• Mã Use Case: UC002

• Mục đích: Khách hàng đặt lịch nhằm lên thời gian, ngày và chi nhánh mà khách hàng sẽ đến để xử lý pin cũ.

• Tác nhân chính: Khách hàng

• Luồng sự kiện chính:

1. Khách hàng đăng nhập vào hệ thống.

2. Chọn pin mà khách hàng cần xử lý.

3. Điền các thông tin đặt lịch.

4. Gửi thông tin đặt lịch về hệ thống.

• Luồng sự kiện thay thế:

o Nếu khách hàng nhập thông tin không hợp lệ, hệ thống sẽ thông báo và yêu cầu nhập lại.

o Nếu khách hàng quên mật khẩu, hệ thống sẽ cung cấp tùy chọn khôi phục mật khẩu.

• Điều kiện trước: Khách hàng đã đăng nhập vào hệ thống.

• Điều kiện sau: Khách hàng nhận được cuộc gọi xác nhận qua số điện thoại mà khách hàng cung cấp

**Sơ đồ Activity đặt lịch**

A screenshot of a computer

Description automatically generated

**Sơ đồ Sequence đặt lịch**

A diagram of a system

Description automatically generated

#### 3.2.4.3 Nhập thông tin pin mới

**Use Case: Nhập Thông Tin Pin Mới**

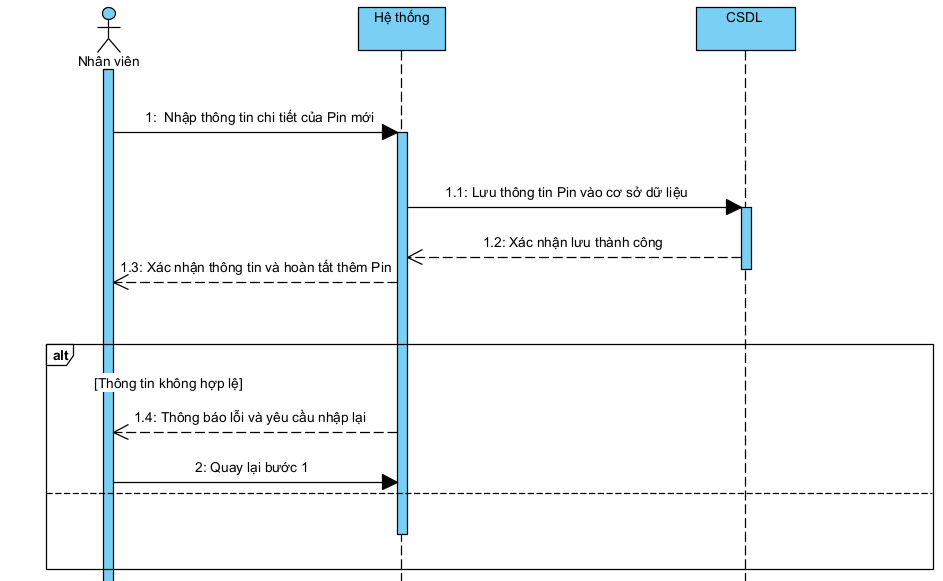
* **Mã Use Case**: UC003
* **Mục đích**: Nhập thông tin pin mới vào hệ thống để quản lý và tái chế.
* **Tác nhân chính**: Nhân viên
* **Luồng sự kiện chính**:
  1. Nhân viên đăng nhập vào hệ thống.
  2. Nhập thông tin chi tiết của pin mới bao gồm tên, số lượng, giá thu mua, mô tả …
  3. Lưu của pin mới vào hệ thống.
* **Luồng sự kiện thay thế:**
  + Nếu khách hàng nhập thông tin không hợp lệ, hệ thống sẽ thông báo và yêu cầu nhập lại.
  + Nếu khách hàng quên mật khẩu, hệ thống sẽ cung cấp tùy chọn khôi phục mật khẩu.
* **Điều kiện trước**: Nhân viên có quyền truy cập vào hệ thống.
* **Điều kiện sau**: Thông tin pin mới được cập nhật và lưu trữ trong hệ thống.

**Sơ đồ Actvity nhập thông tin pin mới**:

A diagram of a flowchart

Description automatically generated

**Sơ đồ Sequence nhập thông tin pin mới**:



#### 3.2.4.4 Thực hiện giao dịch thanh toán

**Use Case: Thực Hiện Giao Dịch Thanh Toán**

• Mã Use Case: UC004

• Mục đích: Thực hiện giao dịch thanh toán cho khách hàng khi khách hàng đến điểm thu gom pin.

• Tác nhân chính: Nhân viên

• Luồng sự kiện chính:

1. Xác nhận pin đúng với lịch xử lý mà khách hàng đã đặt.

2. kiểm tra chi phí phải trả cho khách hàng.

3. Thực hiện giao dịch thanh toán và cung cấp hóa đơn cho khách hàng.

• Luồng sự kiện thay thế:

o Nếu thông tin thanh toán không chính xác, nhân viên cần cập nhật lại thông tin.

o Nếu giao dịch không thành công, hệ thống sẽ thông báo lỗi và cung cấp hướng dẫn để thử lại.

• Điều kiện trước: pin đúng với lịch khách hàng đã đặt từ trước.

• Điều kiện sau: Khách hàng nhận được thanh toán và hóa đơn.

**Sơ đồ Activity thực hiện giao dịch thanh toán:**

A diagram with blue circles and black text

Description automatically generated

**Sơ đồ Sequence thực hiện giao dịch thanh toán:**

A diagram of a project

Description automatically generated

#### 3.2.4.5 Gửi pin

**Use Case: Gửi pin**

• Mã Use Case: UC005

• Mục đích: Cho phép khách hàng gửi pin của họ đến công ty để tái chế.

• Tác nhân chính: Khách hàng

• Luồng sự kiện chính:

1. Khách hàng đóng gói pin.

2. Gửi pin đến địa chỉ công ty tái chế qua dịch vụ bưu chính hoặc trực tiếp.

3. Nhận xác nhận từ công ty khi đơn hàng đến nơi.

• Luồng sự kiện thay thế:

* Nếu dữ liệu không đầy đủ, nhân viên sẽ tiến hành thu thập lại thông tin.
* Nếu dữ liệu nhập lại không khớp nữa thì sẽ hủy đơn và gửi lại pin.

• Điều kiện trước: Khách hàng đã đặt lịch xử lý pin và được gọi điện xác nhận thông tin.

• Điều kiện sau: Pin cũ được công ty nhận và chuẩn bị cho quá trình kiểm kê thanh toán.

**Sơ đồ Activity gửi pin:**

****

**Sequence gửi pin:**

A diagram of a diagram

Description automatically generated with medium confidence

#### 3.2.4.6 Xác nhận nhận lại lịch đặt

**Use Case: Xác Nhận Nhận Lại Lịch Đặt**

• Mã Use Case: UC006

• Mục đích: Xác nhận việc thông tin khách hàng điền trong đơn đặt lịch có chính xác chưa bằng cuộc gọi xác nhận.

• Tác nhân chính: Nhân viên

• Luồng sự kiện chính:

1. Nhận được lịch đặt xử lý pin từ khách hàng

2. Gọi điện xác nhận đơn đặt lịch

3. Xác nhận việc nhận pin (nhận trực tiếp hoặc nhận qua bưu kiện) và thanh toán (các chi phí liên quan) với khách hàng.

• Luồng sự kiện thay thế:

o Nếu dữ liệu không đầy đủ, nhân viên sẽ tiến hành thu thập lại thông tin.

• Điều kiện trước: Nhận được lịch đặt xử lý pin từ khách hàng.

• Điều kiện sau: Khách hàng và công ty đều xác nhận thông tin pin, thơi gian và chi phí thu gom.

**Sơ đồ Activity xác nhận lại lịch đặc:**

**A diagram with blue text

Description automatically generated**

**Sơ đồ Sequence xác nhận lại lịch đặc:**

A diagram of a project

Description automatically generated

#### 3.2.4.7 Kiểm tra pin và lịch đặt

**Use Case: Kiểm Tra Pin Và Lịch Đặt**

• Mã Use Case: UC007

• Mục đích: Kiểm tra loại pin, số lượng pin mà khách hàng đem lại để xác nhận nhập kho.

• Tác nhân chính: Nhân viên

• Luồng sự kiện chính:

1. Kiểm tra thông tin khách hàng cung cấp trùng với lịch khách hàng đã đặt hay chưa.

2. Chấp nhận đơn đặt lịch.

3. Thanh toán cho khách hàng.

• Luồng sự kiện thay thế:

ONếu khách hàng không đặt lịch trước tiến hành thêm khách hàng mới và nhập các thông tin của khách hàng này.

o Nếu dữ liệu không đầy đủ, nhân viên sẽ tiến hành thu thập lại thông tin.

o Nếu hệ thống báo cáo gặp sự cố, nhân viên sẽ tạo báo cáo thủ công.

• Điều kiện trước: khách hàng đã đem hoặc gửi pin đến địa điểm thu gom.

• Điều kiện sau: xác nhận và thanh toán.

**Sơ đồ activity kiểm tra pin và lịch đặt:**

A screenshot of a diagram

Description automatically generated

**Sơ đồ sequence kiểm tra pin và đặt lịch:**

A diagram of a diagram

Description automatically generated with medium confidence

#### 3.2.4.8 Báo cáo định kỳ

**Use Case: Báo Cáo Định Kỳ**

• Mã Use Case: UC008

• Mục đích: Thực hiện báo cáo định kỳ về hoạt động tái chế của công ty.

• Tác nhân chính: Nhân viên

• Luồng sự kiện chính:

1. Thu thập dữ liệu hoạt động thu gom pin cũ.

2. Phân tích và tổng hợp thông tin.

3. Tạo báo cáo và gửi cho ban quản lý.

• Luồng sự kiện thay thế:

o Nếu pin không nằm trong chương trình thu gom tái chế, đề xuất phương án tiêu hủy.

• Điều kiện trước: Có đủ dữ liệu về hoạc động thu gom pin tái chế.

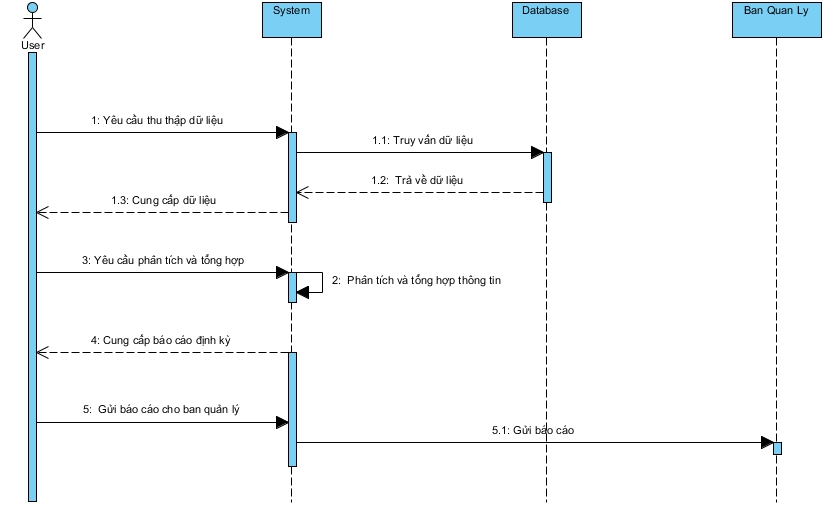
• Điều kiện sau: Báo cáo được ban quản lý xem xét và phê duyệt.

**Sơ đồ activity báo cáo định kỳ:**

**A screenshot of a computer

Description automatically generated**

**Sơ đồ Sequence báo cáo định kỳ:**



#### 3.2.4.9 Cập nhật pin

**Use Case: Cập nhật pin**

• Mã Use Case: UC013

• Mục đích: Giúp khách hàng có thể cập nhật lại số lượng pin cần đưa đi tái chế.

• Tác nhân chính: khách hàng

• Luồng sự kiện chính:

1. Khách hàng nhập lượng pin theo loại.

2. Hệ thống ghi nhận pin đã cập nhật của khách hàng.

3. Hệ thống hiển thị thông tin về số lượng pin đã thu gom của khách hàng

4. Khách hàng có thể chỉnh sửa số lượng pin đã cập nhật nếu cần thiết

5. Khách hàng xác nhận thông tin và hoàn tất việc cập nhật.

• Luồng sự kiện thay thế:

o Hệ thống hiển thị thông báo lỗi cho khách hàng, yêu cầu họ nhập số lượng pin hợp lệ.

o Khách hàng sửa lại số lượng pin và tiếp tục quy trình

• Điều kiện trước: Khách hàng đã cập nhật thành công số lượng pin cần tái chế

• Điều kiện sau: Hệ thống đã cập nhật số lượng pin cần tái chế của khách hàng.

**Sơ đồ Activity cập nhật pin**

**A screenshot of a computer screen

Description automatically generated**

**Sơ đồ sequence cập nhật pin**

**A diagram of a system

Description automatically generated**

#### 3.2.4.10 Xóa pin

**Use Case: Xóa pin**

• Mã Use Case: UC014

• Mục đích: Xóa bỏ thông tin pin đã được tái chế khỏi hệ thống.

• Tác nhân chính: Nhân viên thu gom

• Luồng sự kiện chính:

1. Nhân viên đăng nhập tài khoản admin thành công vào hệ thống.

2. Nhân viên chọn pin cần xóa từ danh sách

3. Hệ thống hiển thị chi tiết thông tin pin và yêu cầu xác nhận việc xóa.

4. Nhân viên xác nhận yêu cầu xóa pin.

5. Hệ thống thực hiện xóa pin khỏi cơ sở dữ liệu.

6. Hệ thống cập nhật danh sách pin tái chế và hiển thị thông báo thành công cho nhân viên.

• Luồng sự kiện thay thế:

o Nhân viên không có quyền xóa pin tái chế Hệ thống hiển thị thông báo lỗi và yêu cầu kiểm tra quyền hạn

• Điều kiện trước: Nhân viên có quyền xóa pin tái chế

• Điều kiện sau: Hệ thống cập nhật danh sách pin hiện có sau khi xóa.

**Sơ đồ Activity xóa pin**

A screenshot of a diagram

Description automatically generated

**Sơ đồ sequence xóa pin**

**A diagram of a system

Description automatically generated**

#### 3.2.4.11 Xem giỏ hàng

**Use Case: Xem Giỏ Hàng**

• Mã Use Case: UC011

• Mục đích: Giúp khách hàng có thể xem lại lịch đã đặt ở giỏ hàng , kiểm tra được thời gian địa điểm và ngày đặt lịch….

• Tác nhân chính: Khách hàng

• Luồng sự kiện chính:

1: khách hàng đặt lịch thành công

2: hệ thống thông báo đã thêm lịch xử lý pin vào giỏ hàng

3: khách hàng nhấp vào giỏ hàng để xem lịch

4: hệ thống hiển thị lịch của khách hàng

• Luồng sự kiện thay thế:

o Đặt lịch thất bại do chưa đăng nhập

• Điều kiện trước: Khách hàng đặt lịch thành công

• Điều kiện sau: Hệ thống lưu trữ được thông tin đặt lịch xử lý pin của khách hàng.

**Sơ đồ activity xem giỏ hàng:**

**A screenshot of a diagram

Description automatically generated**

#### 3.2.4.12 Thêm lịch đặt

**Use Case: Thêm Lịch Đặt**

• Mã Use Case: UC012

• Mục đích: Giúp cho nhân viên thêm lịch đặt cho khách hàng chưa đặt lịch trên web một cách nhanh chống

• Tác nhân chính: Nhân viên

• Luồng sự kiện chính:

1: Nhân viên mở thêm lịch đặt cho khách hàng vãn lai

2: Hệ thống hiển thi giao diện điền thông tin

3: Nhân viên nhập thông tin

4: Hệ thống thêm khách hàng vào danh sách đặt lịch

• Luồng sự kiện thay thế:

o Cập nhật thất bại do nhập sai yêu cầu nhập lại

• Điều kiện trước: Nhân viên đăng nhập vào hệ thống

• Điều kiện sau: Hệ thống lưu trữ được thông tin đặt lịch xử lý pin của khách hàng.

**Sơ đồ activity thêm lịch đặt:**

**A diagram of a diagram

Description automatically generated with medium confidence**

#### 3.2.4.13 Xóa lịch đặt

**Use Case: Xóa Lịch Đặt**

• Mã Use Case: UC013

• Mục đích: Giúp cho nhân viên xóa lịch đặt cho khách hàng

• Tác nhân chính: Nhân viên

• Luồng sự kiện chính:

1: Nhân viên tìm lịch đặt

2: Nhan viên nhấn nút xóa lịch

2: Hệ thống thông báo chắc chắn xóa

3: Nhân viên chọn chắc chắn xóa

4: Hệ thống xóa khách hàng ra khỏi danh sách đặt lịch và thông báo xóa thành công

• Luồng sự kiện thay thế:

o xóa thất bại do lỗi hệ thống

• Điều kiện trước: Nhân viên đăng nhập vào hệ thống

• Điều kiện sau: Hệ thống cập nhật lại dánh sách đặt lịch sau khi xóa

**Sơ đồ activity xóa lịch đặt:**

A screenshot of a diagram

Description automatically generated

#### 3.2.4.14 Cập nhật lịch đặt

**Use Case: Cập Nhật Lịch Đặt**

• Mã Use Case: UC014

• Mục đích: Giúp cho nhân viên cập nhật lại lịch đặt cho khách hàng nếu khách hàng đem pin đến không đúng vs lịch đặt

• Tác nhân chính: Nhân viên

• Luồng sự kiện chính:

1: Nhân viên tìm lịch đặt

2: Nhan viên nhấn nút cập nhật

2: Hệ thống hiển thị giao diện cập nhật

3: Nhân viên điền các thông tin cập nhật mới

4: Hệ thống cập nhật lại cho lịch đặt của khách hàng

• Luồng sự kiện thay thế:

o Cập nhật thất bại do sai các trường thông tin

• Điều kiện trước: Nhân viên đăng nhập vào hệ thống

• Điều kiện sau: Hệ thống cập nhật lại dánh sách đặt lịch sau khi cập nhật

**Sơ đồ cập nhật thêm lịch đặt:**

A screenshot of a computer

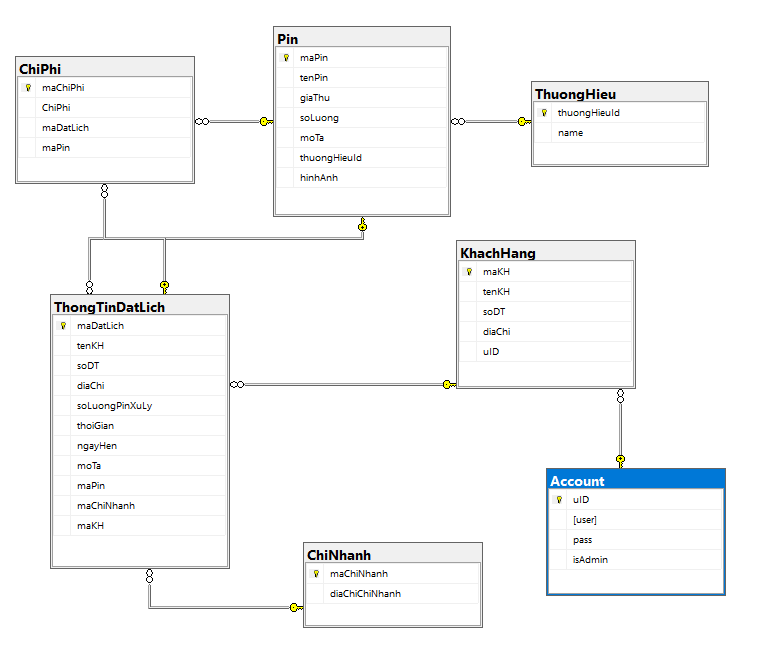
Description automatically generated

## 3.3 Class diagram

A computer screen shot of a computer

Description automatically generated

## 3.4 Database diagram



# CHƯƠNG IV: HIỆN THỰC

## 4.1 Cấu hình phần cứng, phần mềm

**Phần cứng**

CPU: Intel Core i5, 2.3 GHz

RAM: 8 GB

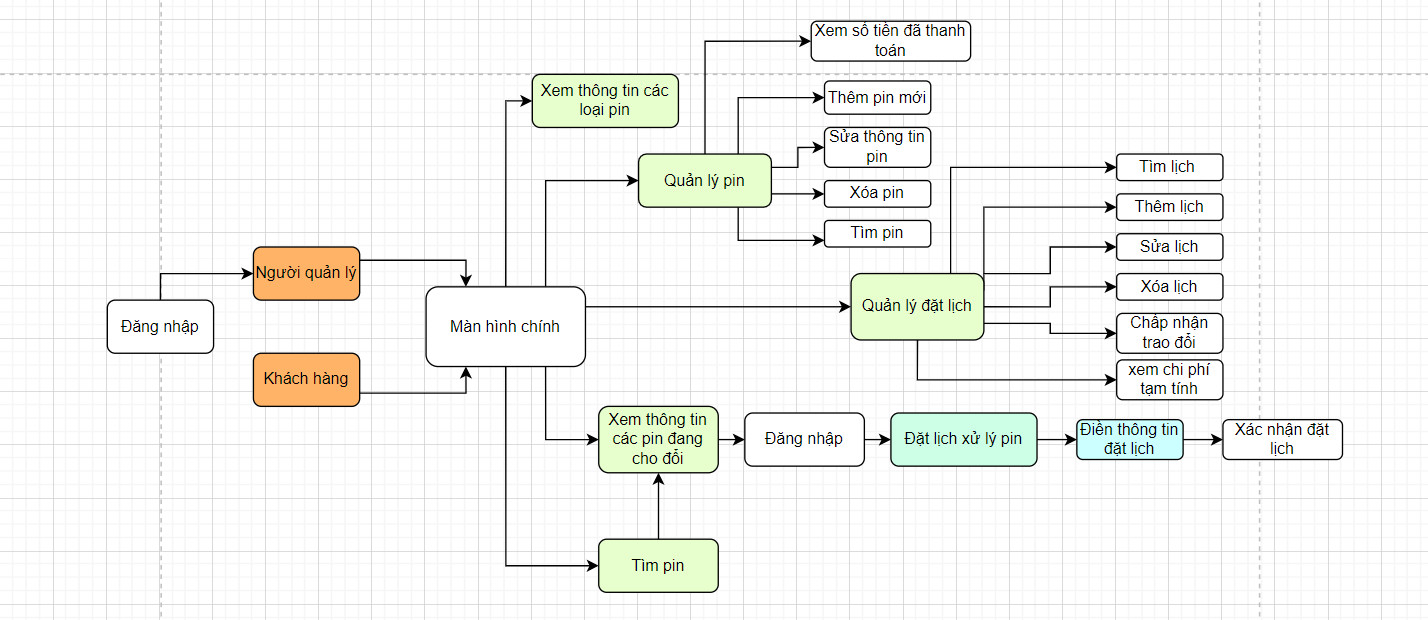
HDD: 360 GB

Architecture: 64 bit

**Phần mềm**

|  |  |  |
| --- | --- | --- |
| **Tên phần mềm** | **Phiên bản** | **Loại** |
| Eclipse IDE for Java EE Developers | 12.0 | IDE cho Javax |
| Microsoft SQL Server 2014 | 19.0.1084.56 | Hệ quản trị cơ sở dữ liệu |
| Microsoft Windows 10 | 10 | Hệ điều hành |

## 4.2 Screen Flow: Phân luồng màn hình



## 4.3 Chức năng chính của các actor

|  |  |  |
| --- | --- | --- |
|  | Actor | Chức năng |
| 1 | Nhân viên | Cập nhật trạng thái đơn hàng. |
|  |  | Xem và chỉnh sửa thông tin đặt lịch của khách hàng. |
|  |  | Chỉnh sửa thông tin pin. |
|  |  | Chấp nhận đặt lịch |
|  |  | Xuất hóa đơn |
|  |  | Xuất thống kê thu chi |
|  |  |  |
| 2 | Khách hàng | Xem sản phẩm. |
|  |  | Đặt lịch hẹn thu gom pin. |
|  |  | Xem lịch hẹn |
|  |  | Đổi lịch hẹn |
|  |  | Đăng ký tài khoản |
|  |  |  |
| 3 | Admin | Quản lý nhân viên. |
|  |  | Quản lý lịch hẹn. |
|  |  | Quản lý khách hàng. |
|  |  | Quản lý tài khoản. |
|  |  | Quản lý thu chi. |

## 4.4 Giao diện của hệ thống

**1: Trang chủ**

A screenshot of a computer

Description automatically generated

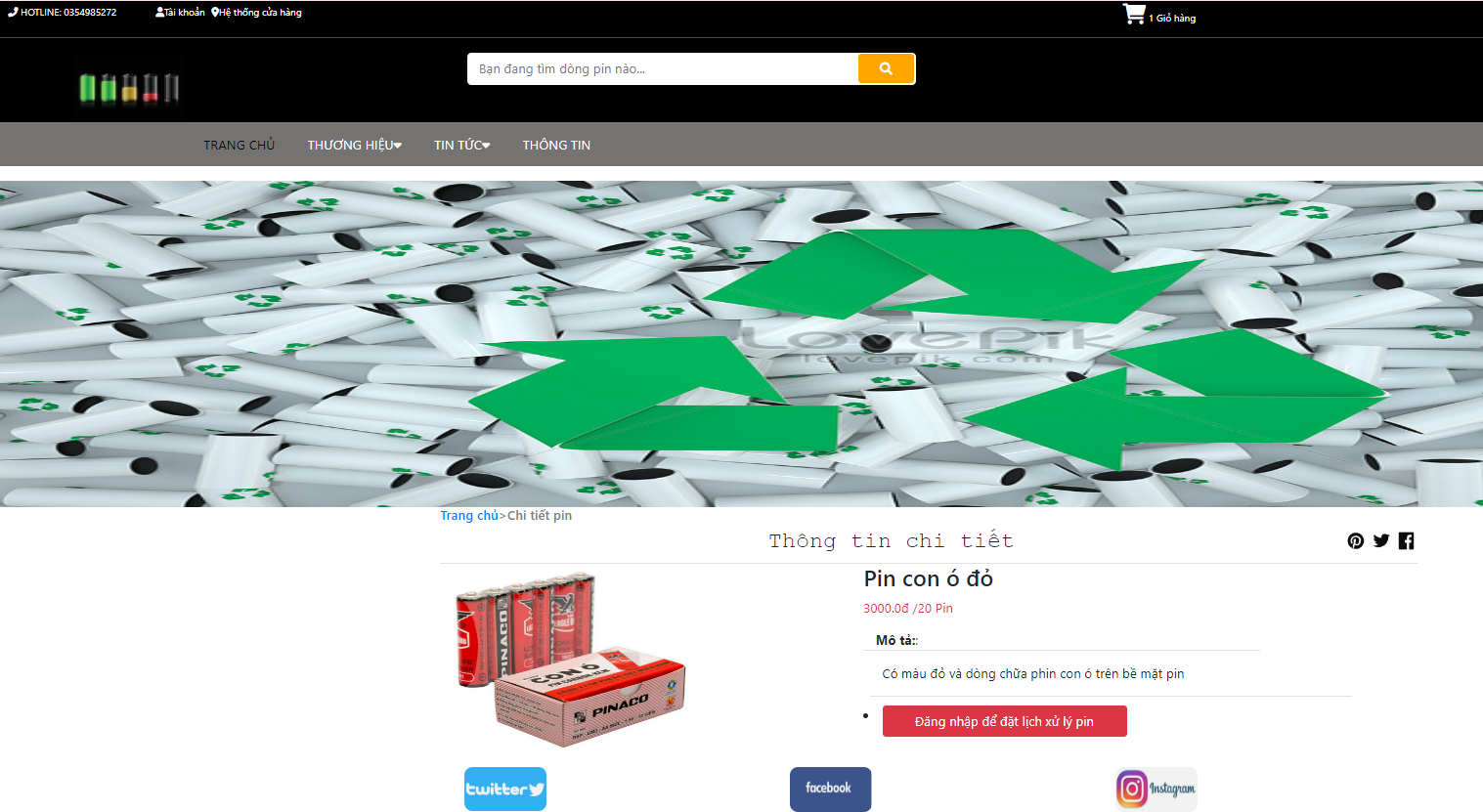
A screenshot of a computer

Description automatically generated

A container full of objects

Description automatically generated

**2: Trang thông tin chi tiết**

****

**3: Trang đăng nhập đăng ký tài khoản**

**Đăng ký:**

**A screenshot of a computer

Description automatically generated  
Đăng nhập:**

**A screenshot of a login box

Description automatically generated**

**4: Trang đặt lịch xử lý pin**

**A screenshot of a computer

Description automatically generated**

**5: Trang quản lý pin**

**A screenshot of a computer

Description automatically generated**

**6: Giao diện thêm pin mới**

A screenshot of a computer

Description automatically generated

**7: Giao diện cập nhật pin**

A screenshot of a computer

Description automatically generated

8: Giao diện quản lý đặt lịch

A screenshot of a computer

Description automatically generated

**9: Giao diện thêm lịch:**

A screenshot of a computer

Description automatically generated

**10: Giao diện cập nhật lại thông tin đặt lịch**

**A screenshot of a computer

Description automatically generated**

## 4.5 Kế hoạch kiểm thử của hệ thống

### 4.5.1 Những yêu cầu về tài nguyên cho kiểm thử

#### 4.5.1.1.1 Phần cứng

Máy tính cá nhân có kết nối mạng LAN.

|  |  |  |  |
| --- | --- | --- | --- |
| **CPU** | **RAM** | **HDD** | **Architecture** |
| Intel Core i5, 2.3 GHz | 8 GB | 360 GB | 64 bit |

#### 4.5.1.1.2 Phần mềm

|  |  |  |
| --- | --- | --- |
| **Tên phần mềm** | **Phiên bản** | **Loại** |
| Eclipse IDE for Java EE Developers | 12.0 | IDE cho Javax |
| Microsoft SQL Server 2014 | 19.0.1084.56 | Hệ quản trị cơ sở dữ liệu |
| Microsoft Windows 10 | 10 | Hệ điều hành |

### 4.5.2 Danh sách các tình huống kiểm tra

# CHƯƠNG V: KẾT LUẬN

## 5.1 Kết quả đạt được

**Về phần trang web**

Giao diện website:

Giao diện website cần đơn giản, dễ sử dụng và thân thiện với người dùng.

Website cần có bố cục rõ ràng, khoa học và dễ điều hướng.

Website cần sử dụng màu sắc hài hòa, phù hợp với thương hiệu của doanh nghiệp.

Website cần có hình ảnh và video chất lượng cao để thu hút khách hàng.

Chức năng website:

Website có chức năng cho phép khách hàng tìm kiếm thông tin về pin mà họ muốn bán.

Website có chức năng cho phép khách hàng đặt lịch thu gom pin.

## 5.2 Hạn chế của đồ án

Website cần có chức năng cho phép khách hàng theo dõi trạng thái đơn hàng.

Website cần có chức năng cho phép khách hàng thanh toán cho các dịch vụ thu mua.

Website cần có chức năng cho phép khách hàng đánh giá và xếp hạng dịch vụ.

Website cần có chức năng cho phép khách hàng liên hệ với bộ phận hỗ trợ khách hàng.

Chưa thực hiện được việc hiển thị lịch đặt trong giỏ hàng của khách hàng

## 5.3 Hướng phát triển

Kết hợp thêm các chức năng quản lý để tối ưu hóa quy trình của chức năng.

Phát triển nâng cao trải nghiệm người dùng.