**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**



**ReCYCLEPRO: ỨNG DỤNG Tái chế thiết bị di động cũ**

***Nhóm 07 - Sinh viên thực hiện***

1. Võ Hoàng Lộc – 20052391 (Nhóm trưởng)
2. Huỳnh Văn Sang – 20060121
3. Nguyễn Văn Giàu – 20053331

# MỤC LỤC

[MỤC LỤC 1](#_Toc167044157)

[DANH MỤC CÁC HÌNH VẼ 3](#_Toc167044158)

[DANH MỤC CÁC BẢNG BIỂU 4](#_Toc167044159)

[CHƯƠNG 1 : GIỚI THIỆU 5](#_Toc167044160)

[1.1 Tổng quan 5](#_Toc167044161)

[1.2 Mục tiêu đề tài 5](#_Toc167044162)

[1.3 Phạm vi đề tài 5](#_Toc167044163)

[1.4 Mô tả yêu cầu chức năng 6](#_Toc167044164)

[CHƯƠNG 2 : CƠ SỞ LÝ THUYẾT 8](#_Toc167044165)

[2.1 Java JDK 17 8](#_Toc167044166)

[2.2 DynamoDB17 9](#_Toc167044167)

[2.3 Simple Storage Service 12](#_Toc167044168)

[2.4 Android SDK 34 13](#_Toc167044169)

[2.5 Kiến trúc monolithic 15](#_Toc167044170)

[CHƯƠNG 3 : PHÂN TÍCH VÀ THIẾT KẾ 19](#_Toc167044171)

[3.1 Kiến trúc 19](#_Toc167044172)

[3.2 Mã giả thuật toán 27](#_Toc167044173)

[3.2.1 Mã Giả cho convertRatingToPercentage 27](#_Toc167044174)

[3.2.2 Mã Giả cho costingPrice 28](#_Toc167044175)

[3.3 Phân tích yêu cầu bằng UML 30](#_Toc167044176)

[3.3.1 Usecase tổng quát 30](#_Toc167044177)

[3.3.2 Danh sách tác nhân và mô tả 30](#_Toc167044178)

[3.3.3 Danh sách các tình huống hoạt động (Use cases) 30](#_Toc167044179)

[3.3.4 Tình huống hoạt động 32](#_Toc167044180)

[3.3.4.1 UC01\_Đăng ký 32](#_Toc167044181)

[3.3.4.2 UC02\_Đăng nhập 33](#_Toc167044182)

[3.3.4.3 UC03\_Quên mật khẩu 35](#_Toc167044183)

[3.3.4.4 UC04\_ Gửi yêu cầu thông tin sản phẩm tái chế 37](#_Toc167044184)

[3.3.4.5 UC05\_Xem lịch sử yêu cầu thông tin sản phẩm tái chế 40](#_Toc167044185)

[3.3.4.6 UC06\_Xem lịch sử đã đánh giá 44](#_Toc167044186)

[3.3.4.7 UC07\_Kiểm tra yêu cầu tái chế 47](#_Toc167044187)

[3.3.4.8 UC08\_Xem danh sách xác nhận tái chế 52](#_Toc167044188)

[3.3.4.9 UC09\_Thống kê của khách hàng 55](#_Toc167044189)

[3.3.4.10 UC10\_Thống kê của đánh giá viên 57](#_Toc167044190)

[3.4 Class diagram 59](#_Toc167044191)

[3.5 Deployment diagram 61](#_Toc167044192)

[CHƯƠNG 4 : HIỆN THỰC 62](#_Toc167044193)

[4.1 Cấu hình phần cứng, phần mềm 62](#_Toc167044194)

[4.2 Giao diện của hệ thống 62](#_Toc167044195)

[4.3 Kế hoạch và hiện thực kiểm thử hệ thống 73](#_Toc167044196)

[4.3.1 Kế hoạch kiểm thử 73](#_Toc167044197)

[4.3.2 Kiểm thử hệ thống 74](#_Toc167044198)

[CHƯƠNG 5 : KẾT LUẬN 81](#_Toc167044199)

[5.1 Kết quả đạt được 81](#_Toc167044200)

[5.2 Hạn chế của đồ án 82](#_Toc167044201)

[5.3 Hướng phát triển 83](#_Toc167044202)

[TÀI LIỆU THAM KHẢO 86](#_Toc167044203)

# DANH MỤC CÁC HÌNH VẼ

[*1. AWS DynamoDB* 10](#_Toc167044204)

[*2. AWS Simple Storage Service* 12](#_Toc167044205)

[*3. Android SDK* 14](#_Toc167044206)

[*4. Lớp DynamoDBManager* 20](#_Toc167044207)

[*5.Layout activity\_submit\_product\_side.xml* 22](#_Toc167044208)

[*6. Cấu trúc view* 23](#_Toc167044209)

[*7. Lớp SubmitProductSide nằm trong activities* 24](#_Toc167044210)

[*8. Lớp SubmitProductSide* 26](#_Toc167044211)

[*9. Mã giả cho hàm convertRatingToPercentage* 27](#_Toc167044212)

[*10. Mã giả cho hàm costingPrice* 28](#_Toc167044213)

[*11. Usecase tổng quát* 30](#_Toc167044214)

[*12. Giao diện đăng nhập, đăng ký (từ trái sang)* 62](#_Toc167044215)

[*13. Giao diện khách hàng, đánh giá viên (từ trái sang)* 63](#_Toc167044216)

[*14. Giao diện gửi sản phẩm của khách hàng.* 64](#_Toc167044217)

[*15. Giao diện xem lại lịch sử gửi và xác nhận tái chế của khách hàng* 66](#_Toc167044218)

[*16. Giao diện thống kê của khách hàng* 67](#_Toc167044219)

[*17. Giao diện kiểm tra đánh giá sản phẩm của đánh giá viên.* 69](#_Toc167044220)

[*18. Giao diện lịch sử đánh giá của đánh giá viên.* 71](#_Toc167044221)

[*19. Giao diện thống kê của đánh giá viên.* 72](#_Toc167044222)

[*20. Giao diện tình trạng tái chế của đánh giá viên.* 73](#_Toc167044223)

# DANH MỤC CÁC BẢNG BIỂU

[*Bảng 3.1 Danh sách tác nhân và mô tả* 30](#_Toc167044224)

[*Bảng 3.2 Danh sách các tình huống trong hệ thống* 31](#_Toc167044225)

[*4.1 Phần cứng thực hiện kiểm thử* 73](#_Toc167044226)

[*4.2 Phần mềm thực hiện kiểm thử* 74](#_Toc167044227)

[*5.1 Kịch bản kiểm thử.* 74](#_Toc167044228)

[*6. Bảng kết quả kiểm thử.* 80](#_Toc167044229)

# : GIỚI THIỆU

## Tổng quan

Nhằm đơn giản hóa quá trình tái chế sản phẩm điện thoại cũ.

Khách hàng gửi yêu cầu đánh giá thiết bị cần tái chế thông qua thông qua ứng dụng RecyclePro.

Đánh giá viên sẽ thực hiện: Đánh giá tổng quan dựa trên mô tả, hình ảnh từ người gửi sau đó đó báo giá cho khách hàng.

Nếu hài lòng, khách hàng sẽ xác nhận thông tin qua ứng dụng kèm theo địa chỉ chi nhánh gửi kèm theo thiết bị của mình qua đường vận chuyển.

Sau khi nhận được thiết bị, đánh giá viên sau đó sẽ đánh giá thiết bị để xác định xem thiết bị có hoạt động tốt không có điều kiện hay không nếu thiết bị hoạt động giống như mô tả tình trạng trước đó sẽ xác nhận thanh toán.

Dựa trên đánh giá, thiết bị được sau đó được tái chế hoặc bán lại.

Cuối cùng, công ty định kỳ chạy thống kê và báo cáo hoạt động dựa trên việc tái chế hoạt động.

## Mục tiêu đề tài

Nhằm giảm thiểu tình trạng thiết bị điện thoại sau khi sử dụng được tái chế đúng cách thay vì bị vứt bỏ ra môi trường:

Đối với khách hàng xác định được chi phí được trả cho mỗi sản phẩm cần gửi tái chế.

Đối với đánh giá viên giảm bớt quy trình phức tạp trong quá trình giao tiếp với khách hàng.

## Phạm vi đề tài

Ứng dụng được sử dụng bới đánh giá viên và khách hàng.

Ứng dụng hỗ trợ cho nền tảng android 7.0 hoặc cao hơn.

Mục dích ứng dụng:

+ Khách hàng: Có thể biết được giá sản phẩm được định giá bao nhiêu trước khi gửi sản phẩm.

+ Đánh giá viên: Tiện lợi trong công việc giải quyết các vấn đề giao tiếp, kiểm tra, xác nhận các yêu cầu của khách hàng minh bạch hơn trong công việc.

Đối tượng sử dụng tài liệu:

+ Người phát triển phần mềm.

+ Người bảo trì hệ thống phần mềm.

+ Người kiểm thử phần mềm.

## Mô tả yêu cầu chức năng

**Yêu cầu chức năng:**

Quản lý thông tin được tất cả thông tin của khách hàng và đánh giá viên

+ Thông tin được hiển thị theo vai trò.

+ Kiểm soát các vai trò như khách hàng và người đánh giá.

+ Lọc các danh sách như sản phẩm cần đánh giá của khách, lịch sử đánh giá, lịch sử sản phẩm của khách hang đã cung cấp.

Khách hàng quản lý được các thông tin của mình đã cung cấp

+ Các thông tin mà khách hàng cung cấp sẽ hiển thị đầy đủ cho đánh giá viên xem xét.

+ Các sản phẩm đã được đánh giá khách hàng có quyền xem lại các đánh giá chi tiết và rõ ràng và có thể quyết định việc đưa thiết bị tới chi nhánh gần nhất để tái chế

+ Các thông tin khách hàng xác nhận gửi đi tái chế phải đầy đủ rõ ràng để bên tái chế có thể dễ dàng liên lạc và hỗ trợ.

Đánh giá viên có thể quản lý được các đánh giá mà đánh giá viên cung cấp và có thể xem được rõ ràng các yêu cầu xác nhận tái chế của khách hàng sau khi người đánh giá đánh giá sản phẩm.

+ Các thông tin mà đánh giá viên cung cấp phải đầy đủ, chi tiết về tình trạng sản phẩm và đưa ra được mức giá hợp lý cho khách hàng

+ Các thông tin xác nhận tái chế của khách hàng phải đầy đủ.

Thống kê:

+ Khách hàng có thể xem được chi tiết số % qia biểu đồ pie để biết có bao nhiêu % sản phẩm tái chế của khách hàng đã được đánh giá, phân tích số lượng mà khách hàng tái chế qua từng tháng.

+ Đánh giá viên có thể xem được chi tiết số % các sản phẩm mà họ đã đánh giá thuộc loại tái chế hay bán lại bao nhiêu. Xem được sự phân bổ của giá chót mà họ đưa ra thông qua biểu đồ botplox và các biểu đồ khác…

**Yêu cầu phi chức năng:**

Đảm bảo hoạt động được trên mọi thiết bị android.

Thời gian cập nhật dữ liệu dưới 2 giây.

Thời gian hiển thị các danh sách phải dưới 5 giây

Có thể chuyển đổi giao diện tối và sáng

Khả năng quản lý tối thiểu 500 đối tượng

Các thông tin quan trọng phải được hiển thị to và rõ để nhận biết

# : CƠ SỞ LÝ THUYẾT

## Java JDK 17

Java SDK 17, còn được gọi là JDK 17, là phiên bản mới nhất của bộ công cụ phát triển phần mềm Java (SDK) được phát hành vào tháng 9 năm 2021. Nó đi kèm với nhiều tính năng mới và cải tiến, bao gồm:

Records: Records là một tính năng mới giúp đơn giản hóa việc tạo các lớp dữ liệu đơn giản.

Switched Expressions: Switched Expressions là một tính năng mới cung cấp cách viết các biểu thức switch dễ đọc và mạnh mẽ hơn.

Sealed Classes: Sealed Classes là một tính năng mới giúp cải thiện tính an toàn loại và hỗ trợ các lớp không thể mở rộng.

Pattern Matching for instanceof: Pattern Matching for instanceof là một tính năng mới giúp đơn giản hóa việc kiểm tra loại đối tượng.

TLS 1.3: TLS 1.3 là giao thức bảo mật mới được hỗ trợ trong JDK 17.

ZGC (Z Garbage Collector): ZGC là bộ thu gom rác mới được giới thiệu trong JDK 17, hứa hẹn cải thiện hiệu suất và khả năng mở rộng.



*Hình 1 JDK 17*

**Ưu điểm:**

Cải thiện hiệu suất: JDK 17 bao gồm nhiều cải tiến giúp cải thiện hiệu suất của các ứng dụng Java.

Cải thiện bảo mật: JDK 17 bao gồm nhiều cải tiến bảo mật giúp bảo vệ các ứng dụng Java khỏi các mối đe dọa bảo mật.

Cải thiện khả năng sử dụng: JDK 17 bao gồm nhiều tính năng mới giúp cải thiện khả năng sử dụng của ngôn ngữ Java.

Hỗ trợ phiên bản Java mới nhất: JDK 17 cho phép bạn phát triển các ứng dụng hỗ trợ Java 17, phiên bản mới nhất của ngôn ngữ Java.

**Nhược điểm:**

Yêu cầu Java 17: JDK 17 yêu cầu Java 17 để chạy, vì vậy bạn sẽ cần phải cập nhật môi trường phát triển của mình lên Java 17 để sử dụng nó.

Khả năng tương thích: Một số tính năng mới trong JDK 17 có thể không tương thích với các phiên bản Java cũ hơn.

Độ phức tạp: Một số tính năng mới trong JDK 17 có thể phức tạp hơn để sử dụng so với các tính năng cũ.

Java SDK 17 là một phiên bản mạnh mẽ và linh hoạt của SDK Java cung cấp nhiều tính năng mới và cải tiến. Tuy nhiên, nó cũng có một số nhược điểm cần xem xét trước khi sử dụng nó.

## DynamoDB17

Amazon DynamoDB là một cơ sở dữ liệu NoSQL cung cấp khả năng lưu trữ dữ liệu không có cấu trúc, có thể mở rộng và độ bền cao. Nó được sử dụng rộng rãi bởi các doanh nghiệp và cá nhân để lưu trữ các loại dữ liệu khác nhau, bao gồm thông tin người dùng, dữ liệu ứng dụng và nhiều hơn nữa.



*1. AWS DynamoDB*

DynamoDB hoạt động bằng cách chia dữ liệu thành các bảng và các bảng này được chia nhỏ thành các phân vùng. Mỗi phân vùng được lưu trữ trên một máy chủ riêng biệt, điều này giúp đảm bảo rằng dữ liệu được bảo vệ khỏi mất mát dữ liệu và có thể truy cập được từ bất cứ đâu trên thế giới.

DynamoDB cung cấp nhiều tính năng, bao gồm:

Khả năng mở rộng: DynamoDB có thể mở rộng để lưu trữ bất kỳ lượng dữ liệu nào.

Độ bền cao: DynamoDB được thiết kế để bảo vệ dữ liệu khỏi mất mát dữ liệu. Dữ liệu của bạn được sao chép nhiều lần trên nhiều máy chủ trong các trung tâm dữ liệu của Amazon.

Độ trễ thấp: DynamoDB cung cấp độ trễ thấp để truy cập dữ liệu, điều này khiến nó trở nên lý tưởng cho các ứng dụng web và di động.

Khả năng thích ứng: DynamoDB có thể tự động điều chỉnh tải trọng, vì vậy bạn không cần phải lo lắng về việc cung cấp tài nguyên.

Giá cả phải chăng: DynamoDB có giá cả phải chăng và có nhiều tùy chọn giá cả để phù hợp với nhu cầu của bạn. Bạn chỉ trả tiền cho dung lượng lưu trữ và truy cập dữ liệu mà bạn sử dụng.

**Ưu điểm:**

Khả năng mở rộng: DynamoDB là một cơ sở dữ liệu có thể mở rộng cao, có nghĩa là bạn có thể lưu trữ bất kỳ lượng dữ liệu nào bạn cần.

Độ bền cao: DynamoDB được thiết kế để bảo vệ dữ liệu khỏi mất mát dữ liệu. Dữ liệu của bạn được sao chép nhiều lần trên nhiều máy chủ trong các trung tâm dữ liệu của Amazon.

Độ trễ thấp: DynamoDB cung cấp độ trễ thấp để truy cập dữ liệu, điều này khiến nó trở nên lý tưởng cho các ứng dụng web và di động.

Khả năng thích ứng: DynamoDB có thể tự động điều chỉnh tải trọng, vì vậy bạn không cần phải lo lắng về việc cung cấp tài nguyên.

Giá cả phải chăng: DynamoDB có giá cả phải chăng và có nhiều tùy chọn giá cả để phù hợp với nhu cầu của bạn. Bạn chỉ trả tiền cho dung lượng lưu trữ và truy cập dữ liệu mà bạn sử dụng.

**Nhược điểm:**

Độ phức tạp: DynamoDB có thể phức tạp để thiết lập và sử dụng. Có nhiều tùy chọn cấu hình có sẵn và bạn có thể cần trợ giúp từ chuyên gia để thiết lập DynamoDB cho nhu cầu của mình.

Không phù hợp cho tất cả các trường hợp sử dụng: DynamoDB không phù hợp cho tất cả các trường hợp sử dụng. Ví dụ, nó không phù hợp cho các ứng dụng yêu cầu truy vấn dữ liệu phức tạp.

Giá cả: Mặc dù DynamoDB có giá cả phải chăng, nhưng nó có thể đắt đỏ nếu bạn cần truy cập dữ liệu thường xuyên.

Kết luận

DynamoDB là một cơ sở dữ liệu NoSQL mạnh mẽ và linh hoạt có thể được sử dụng bởi các doanh nghiệp và cá nhân để lưu trữ nhiều loại dữ liệu. Tuy nhiên, nó có thể phức tạp để thiết lập và sử dụng và có thể đắt đỏ nếu bạn cần truy cập dữ liệu thường xuyên.

## Simple Storage Service

Amazon S3 (Simple Storage Service) là một dịch vụ lưu trữ đám mây cung cấp khả năng lưu trữ dữ liệu không có cấu trúc, có thể mở rộng và có độ bền cao. Nó được sử dụng rộng rãi bởi các doanh nghiệp và cá nhân để lưu trữ các loại dữ liệu khác nhau, bao gồm ảnh, video, tài liệu, mã nguồn và nhiều hơn nữa.



*2. AWS Simple Storage Service*

S3 hoạt động bằng cách chia dữ liệu thành các đối tượng nhỏ và lưu trữ chúng trên nhiều máy chủ trong các trung tâm dữ liệu của Amazon. Điều này giúp đảm bảo rằng dữ liệu được bảo vệ khỏi mất mát dữ liệu và có thể truy cập được từ bất cứ đâu trên thế giới.

S3 cung cấp nhiều tính năng, bao gồm:

Khả năng mở rộng: S3 có thể mở rộng để lưu trữ bất kỳ lượng dữ liệu nào.

Độ bền cao: S3 được thiết kế để bảo vệ dữ liệu khỏi mất mát dữ liệu.

Bảo mật: S3 cung cấp nhiều tính năng bảo mật để giúp bảo vệ dữ liệu của bạn.

Hiệu suất: S3 cung cấp hiệu suất cao để truy cập dữ liệu.

Giá cả phải chăng: S3 có giá cả phải chăng và có nhiều tùy chọn giá cả để phù hợp với nhu cầu của bạn.

**Ưu điểm:**

Khả năng mở rộng: S3 là một dịch vụ lưu trữ có thể mở rộng cao, có nghĩa là bạn có thể lưu trữ bất kỳ lượng dữ liệu nào bạn cần.

Độ bền cao: S3 được thiết kế để bảo vệ dữ liệu khỏi mất mát dữ liệu. Dữ liệu của bạn được sao chép nhiều lần trên nhiều máy chủ trong các trung tâm dữ liệu của Amazon.

Bảo mật: S3 cung cấp nhiều tính năng bảo mật để giúp bảo vệ dữ liệu của bạn. Bạn có thể kiểm soát quyền truy cập vào dữ liệu của mình và S3 sử dụng mã hóa để bảo vệ dữ liệu của bạn khi nó được lưu trữ và truyền tải.

Hiệu suất: S3 cung cấp hiệu suất cao để truy cập dữ liệu. Bạn có thể nhanh chóng truy cập dữ liệu của mình từ bất cứ đâu trên thế giới.

Giá cả phải chăng: S3 có giá cả phải chăng và có nhiều tùy chọn giá cả để phù hợp với nhu cầu của bạn. Bạn chỉ trả tiền cho dung lượng lưu trữ mà bạn sử dụng.

**Nhược điểm:**

Độ phức tạp: S3 có thể phức tạp để thiết lập và sử dụng. Có nhiều tùy chọn cấu hình có sẵn và bạn có thể cần trợ giúp từ chuyên gia để thiết lập S3 cho nhu cầu của mình.

Chi phí: Mặc dù S3 có giá cả phải chăng, nhưng nó có thể đắt đỏ nếu bạn cần lưu trữ một lượng lớn dữ liệu.

Khả năng truy cập: S3 là một dịch vụ lưu trữ dựa trên đám mây, vì vậy bạn cần có kết nối internet để truy cập dữ liệu của mình.

## Android SDK 34

Android SDK 34 là phiên bản mới nhất của bộ công cụ phát triển phần mềm Android, được phát hành vào tháng 11 năm 2023. Nó đi kèm với nhiều tính năng mới và cải tiến, bao gồm:

Hỗ trợ Android 13: SDK 34 bao gồm tất cả các API mới và cập nhật cho Android 13, phiên bản mới nhất của hệ điều hành Android.

Cải tiến về hiệu suất: SDK 34 bao gồm nhiều cải tiến về hiệu suất để giúp các ứng dụng Android chạy nhanh hơn và mượt mà hơn.

Cải tiến về bảo mật: SDK 34 bao gồm nhiều cải tiến về bảo mật để giúp bảo vệ các ứng dụng Android khỏi các mối đe dọa bảo mật.

Các tính năng mới: SDK 34 bao gồm nhiều tính năng mới, chẳng hạn như hỗ trợ cho các thiết bị có thể gập lại, camera tốt hơn và nhiều hơn nữa.



*3. Android SDK*

**Ưu điểm:**

Hỗ trợ phiên bản Android mới nhất: SDK34 cho phép bạn phát triển các ứng dụng hỗ trợ Android 13, phiên bản mới nhất của hệ điều hành Android. Điều này giúp đảm bảo rằng các ứng dụng của bạn tương thích với hầu hết các thiết bị Android.

Cải thiện hiệu suất: Các cải tiến về hiệu suất trong SDK 34 có thể giúp các ứng dụng của bạn chạy nhanh hơn và mượt mà hơn, điều này có thể cải thiện trải nghiệm người dùng.

Cải thiện bảo mật: Các cải tiến về bảo mật trong SDK 34 có thể giúp bảo vệ các ứng dụng của bạn khỏi các mối đe dọa bảo mật, điều này có thể giúp bảo vệ dữ liệu người dùng.

Các tính năng mới: Các tính năng mới trong SDK 34 có thể cho phép bạn tạo các ứng dụng sáng tạo và hấp dẫn hơn.

**Nhược điểm:**

Yêu cầu Android 13: SDK 34 yêu cầu Android 13 để chạy, vì vậy bạn sẽ cần phải cập nhật thiết bị phát triển của mình lên Android 13 để sử dụng nó.

Khả năng tương thích: Một số tính năng mới trong SDK 34 có thể không tương thích với các thiết bị Android cũ hơn.

Độ phức tạp: SDK 34 có thể phức tạp hơn các phiên bản trước, điều này có thể khiến việc phát triển các ứng dụng Android trở nên khó khăn hơn.

## Kiến trúc monolithic

**Kiến trúc đang được sử dụng :**

Loại kiến trúc : monolithic

Trong kiến trúc monolithic, tất cả các thành phần của ứng dụng được đóng gói thành một đơn vị duy nhất, được triển khai và quản lý như một thể thống nhất. Kiến trúc này có một số ưu điểm, bao gồm:

**Loại mô hình kiến trúc: MVC**

Phân tích kiến trúc MVC và áp dụng kiến trúc với ứng dụng

Kiến trúc MVC (Model-View-Controller) là một mẫu thiết kế phần mềm phổ biến để tổ chức mã nguồn của ứng dụng một cách rõ ràng và dễ bảo trì

**Ưu điểm của kiến trúc Monolithic:**

Dễ dàng phát triển và bảo trì: Với tất cả các thành phần được đặt tại một nơi, việc phát triển và bảo trì ứng dụng trở nên dễ dàng hơn. Các nhà phát triển có thể dễ dàng truy cập và sửa đổi bất kỳ phần nào của mã, và việc theo dõi các thay đổi được thực hiện cũng đơn giản hơn.

Hiệu suất tốt: Kiến trúc monolithic thường có hiệu suất tốt hơn các kiến trúc phân tán, do sự gần gũi giữa các thành phần của ứng dụng. Điều này có thể giúp giảm thiểu độ trễ và cải thiện thời gian phản hồi.

Dễ dàng triển khai: Việc triển khai một ứng dụng monolithic thường đơn giản hơn việc triển khai một ứng dụng phân tán. Điều này là do chỉ có một đơn vị cần được triển khai và quản lý.

**Nhược điểm:**

Khả năng mở rộng hạn chế: Khó có thể mở rộng một ứng dụng monolithic để đáp ứng nhu cầu lưu lượng truy cập tăng cao. Điều này là do tất cả các thành phần của ứng dụng được đặt tại một nơi, và việc thêm tài nguyên có thể dẫn đến tắc nghẽn.

Độ tin cậy thấp: Nếu một phần nào đó của ứng dụng monolithic gặp sự cố, toàn bộ ứng dụng có thể bị sập. Điều này có thể dẫn đến thời gian ngừng hoạt động đáng kể và mất mát dữ liệu.

Phức tạp hơn khi quản lý: Khi một ứng dụng monolithic trở nên lớn hơn và phức tạp hơn, việc quản lý nó có thể trở nên khó khăn hơn. Điều này là do có nhiều thành phần cần được theo dõi và quản lý.

Kiến trúc monolithic thường được sử dụng cho các ứng dụng nhỏ và đơn giản, nơi hiệu suất và khả năng bảo trì là những ưu tiên hàng đầu. Tuy nhiên, đối với các ứng dụng lớn và phức tạp hơn, các kiến trúc phân tán, chẳng hạn như kiến trúc microservices, có thể là lựa chọn tốt hơn.

**Mô hình kiến trúc MVC**

Trong trường hợp của ứng dụng RecyclePro, kiến trúc monolithic có thể phù hợp vì ứng dụng tương đối đơn giản và không có yêu cầu mở rộng cao. Tuy nhiên, nếu ứng dụng tiếp tục phát triển và nhu cầu của người dùng tăng cao, có thể cần phải xem xét chuyển sang một kiến trúc phân tán.

**Ưu điểm:**

Phân tách logic: MVC phân tách ứng dụng thành ba thành phần chính: Model (dữ liệu và logic), View (giao diện người dùng) và Controller (điều khiển hành vi của ứng dụng). Điều này giúp dễ dàng quản lý, duy trì và mở rộng ứng dụng.

Tính tái sử dụng cao: Với MVC, bạn có thể tái sử dụng các thành phần như model hoặc controller trong nhiều phần của ứng dụng, giúp giảm thiểu việc lặp lại mã.

Phát triển đồng thời: Do mỗi thành phần trong MVC có trách nhiệm riêng biệt, nên các nhóm phát triển có thể làm việc đồng thời trên các phần khác nhau của ứng dụng mà không gây xung đột.

Dễ kiểm thử:Đối với các thành phần của MVC, việc kiểm thử có thể được thực hiện một cách độc lập, dễ dàng hơn.

**Nhược điểm:**

Phức tạp khi ứng dụng lớn:

Trong các ứng dụng lớn, MVC có thể dẫn đến việc phức tạp hóa quá mức do sự phụ thuộc mạnh mẽ giữa các thành phần.

Khó duy trì: Đôi khi việc theo dõi luồng điều khiển trong MVC có thể trở nên phức tạp, đặc biệt là khi ứng dụng phát triển lớn.

Khó kiểm thử toàn diện: Trong MVC, kiểm thử có thể trở nên khó khăn khi cần phải kiểm tra tương tác giữa các thành phần, đặc biệt là khi có sự phụ thuộc mạnh mẽ giữa chúng.

Rủi ro về logic ứng dụng phía Client: Trong trường hợp các logic ứng dụng phía client không được quản lý cẩn thận, có thể dẫn đến việc loằng ngoằng logic và khó bảo trì.

**So sánh giữa các mô hình MVC, MVP và MVVM có thể giúp bạn hiểu rõ hơn về điểm mạnh và yếu của từng mô hình:**

**MVC (Model-View-Controller):**

Model: Chứa dữ liệu và logic xử lý.

View: Đại diện cho giao diện người dùng.

Controller: Xử lý tương tác của người dùng và cập nhật model và view tương ứng.

MVP (Model-View-Presenter):

Model: Tương tự như trong MVC, chứa dữ liệu và logic xử lý.

View: Đại diện cho giao diện người dùng, nhưng có trách nhiệm tương tác với người dùng.

Presenter: Lớp trung gian giữa Model và View, xử lý logic kinh doanh và tương tác với model, sau đó cập nhật view.

**MVVM (Model-View-ViewModel):**

Model: Tương tự như trong MVC và MVP, chứa dữ liệu và logic xử lý.

View: Đại diện cho giao diện người dùng, nhưng không chứa logic ứng dụng.

ViewModel: Chứa trạng thái của view và logic hiển thị, tương tác với model và cung cấp dữ liệu cho view thông qua data binding.

**So sánh:**

Tính phân tách logic:

MVC: Logic phân tách không tốt, controller thường chịu trách nhiệm quá nhiều công việc.

MVP: Phân tách logic tốt hơn với presenter đảm nhận nhiều công việc.

MVVM: Phân tách logic tốt nhất với ViewModel, giúp view trở nên đơn giản và dễ kiểm thử.

Dễ kiểm thử:

MVP và MVVM thường dễ kiểm thử hơn so với MVC do presenter và view model có thể được kiểm thử một cách độc lập.

Độ phức tạp:

MVC có thể trở nên phức tạp khi ứng dụng lớn.

MVP và MVVM thường giảm được sự phức tạp thông qua việc phân chia trách nhiệm rõ ràng.

Tính tương tác giữa các thành phần:

Trong MVP, view không biết đến model, chỉ nắm giữ presenter, điều này làm cho việc tái sử dụng và kiểm thử dễ dàng hơn.

Trong MVVM, ViewModel cung cấp dữ liệu trực tiếp cho view thông qua data binding, giảm thiểu sự phụ thuộc và tăng tính linh hoạt.

# : PHÂN TÍCH VÀ THIẾT KẾ

## Kiến trúc

Model: Chứa dữ liệu thao tác dữ liệu.

DynamoDBManager: Lớp này chứa các phương thức truy vấn và thao tác với DynamoDB. Nó đại diện cho phần Model, nơi xử lý dữ liệu. DynamoDBManager thực hiện các thao tác CRUD (Create, Read, Update, Delete) với cơ sở dữ liệu, chẳng hạn như tạo tài khoản mới, kiểm tra tài khoản tồn tại, khách hàng cung cấp thông tin thiết bị để tái chế, đánh giá viên xem thông tin của khách hang cung cấp và đánh giá thiết bị đưa ra giá cuối cùng ,...

Ví dụ về cấu trúc bên trong class dynamoDBManager:

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

*4. Lớp DynamoDBManager*

Lớp DynamoDBManager là một thư viện các lớp được sử dụng để tạo và quản lý cơ sở dữ liệu DynamoDB. Lớp công khai này chịu trách nhiệm tạo và duy trì các bảng, truy vấn dữ liệu và thực hiện các hoạt động khác nhau trên cơ sở dữ liệu.

Thành phần của DynamoDBManager:

Context: Biến này lưu trữ bối cảnh của ứng dụng. Bối cảnh này được sử dụng để truy cập các dịch vụ khác của AWS, chẳng hạn như dịch vụ IAM và S3.

ddbClient: Biến này lưu trữ một đối tượng Amazon DynamoDBClient. Đối tượng này được sử dụng để tương tác với cơ sở dữ liệu DynamoDB.

Các phương thức: Lớp DynamoDBManager cung cấp một số phương thức để thực hiện các hoạt động khác nhau trên cơ sở dữ liệu DynamoDB. Các phương thức này bao gồm:

* checkDynamoDBConnection(): Kiểm tra xem có kết nối thành công với cơ sở dữ liệu DynamoDB hay không.
* initializeDynamoDB(): Khởi tạo đối tượng Amazon DynamoDBClient.
* submitProductInformationForRecycling(String productID, String customerName, String phoneNumber): Gửi thông tin sản phẩm để tái chế vào cơ sở dữ liệu DynamoDB.
* loadProduct(String state, LoadRecyclingProductListListener listener): Tải danh sách các sản phẩm cần tái chế từ cơ sở dữ liệu DynamoDB.
* loadImagesOfProduct(String id, LoadImagesOfProductListener listener): Tải danh sách các hình ảnh của một sản phẩm từ cơ sở dữ liệu DynamoDB.
* createAccount(String email, String password, String userName, String role): Tạo tài khoản mới trong cơ sở dữ liệu DynamoDB.

Mục đích của DynamoDBManager:

Cung cấp một lớp trừu tượng đơn giản để tạo và quản lý cơ sở dữ liệu DynamoDB.

Ẩn đi các chi tiết phức tạp của API DynamoDB.

Cung cấp một giao diện dễ sử dụng để thực hiện các hoạt động khác nhau trên cơ sở dữ liệu DynamoDB.

1. View: Chứa giao diện người dùng (UI)

Ví dụ activity\_submit\_product\_side.xml: Đây là tệp giao diện XML định nghĩa UI của màn hình đăng ký. Tệp này chứa các thành phần UI như EditText, Button, TextView để người dùng nhập thông tin sản phẩm tái chế và tương tác với ứng dụng.

Ví dụ về cấu trúc bên trong class của file XML:

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

*5.Layout activity\_submit\_product\_side.xml*

Ví dụ cấu trúc thư mục view:

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, thiết kế

Mô tả được tạo tự động

*6. Cấu trúc view*

Ngoài ra:

SubmitProductSide (Activity): Activity này có trách nhiệm hiển thị UI cho người dùng và xử lý các sự kiện giao diện người dùng (như nhấn nút). Nó tương tác với các thành phần trong activity\_submit\_product\_side.xml để nhận và hiển thị dữ liệu từ người dùng.

Trong android studio thực hiện bằng cách khai báo các thành phần như Edittext, Button, ImageButton

Ảnh có chứa văn bản, ảnh chụp màn hình

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

*7. Lớp SubmitProductSide nằm trong activities*

Trong Android Studio, findViewById(R.id.etProductName) là một phương thức được sử dụng để tìm và tạo một tham chiếu đến một phần tử giao diện người dùng (UI element) trong layout XML. Trong trường hợp này, etProductName có thể là một EditText (hộp văn bản cho phép người dùng nhập văn bản).

Việc này là cần thiết vì khi bạn định nghĩa các phần tử trong layout XML của mình, chúng không tự động trở thành các đối tượng trong mã Java của bạn. Do đó, bạn cần phải sử dụng findViewById để tìm và liên kết chúng với các biến trong mã của bạn, để bạn có thể thao tác với chúng trong mã Java của mình.

Phân tích chi tiết:

* etProductName: Đây là một biến được sử dụng để lưu trữ đối tượng View đại diện cho trường nhập liệu tên sản phẩm trên màn hình.
* findViewById(): Đây là phương thức được sử dụng để tìm kiếm View trên màn hình dựa vào ID của View đó. ID của View được định nghĩa trong file bố cục (layout file) của màn hình.
* R.id.etProductName: Đây là ID của View được sử dụng để tìm kiếm trường nhập liệu tên sản phẩm. ID này được định nghĩa trong file bố cục của màn hình tìm kiếm ngày mua sản phẩm.
* Controller: Xử lý logic giữa Model và View

1. SubmitProductSide (Activity): Lớp này cũng đóng vai trò như một Controller. Nó nhận sự kiện từ người dùng (như nhấn nút gửi thông tin sản phẩm tái chế) và sử dụng DynamoDBManager để thực hiện các thao tác nghiệp vụ (như them thông tin sản phẩm tái chế khách hang cung cấp, xem thông tin sản phẩm tái chế để thực hiện quy trình đánh giá đối với đánh giá viên). Nó tương tác với cả Model và View để cập nhật dữ liệu và hiển thị phản hồi cho người dùng.

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phần mềm đa phương tiện

Mô tả được tạo tự động

*8. Lớp SubmitProductSide*

-Luồng xử lý

Luồng xử lý của ứng dụng được mô tả như sau:

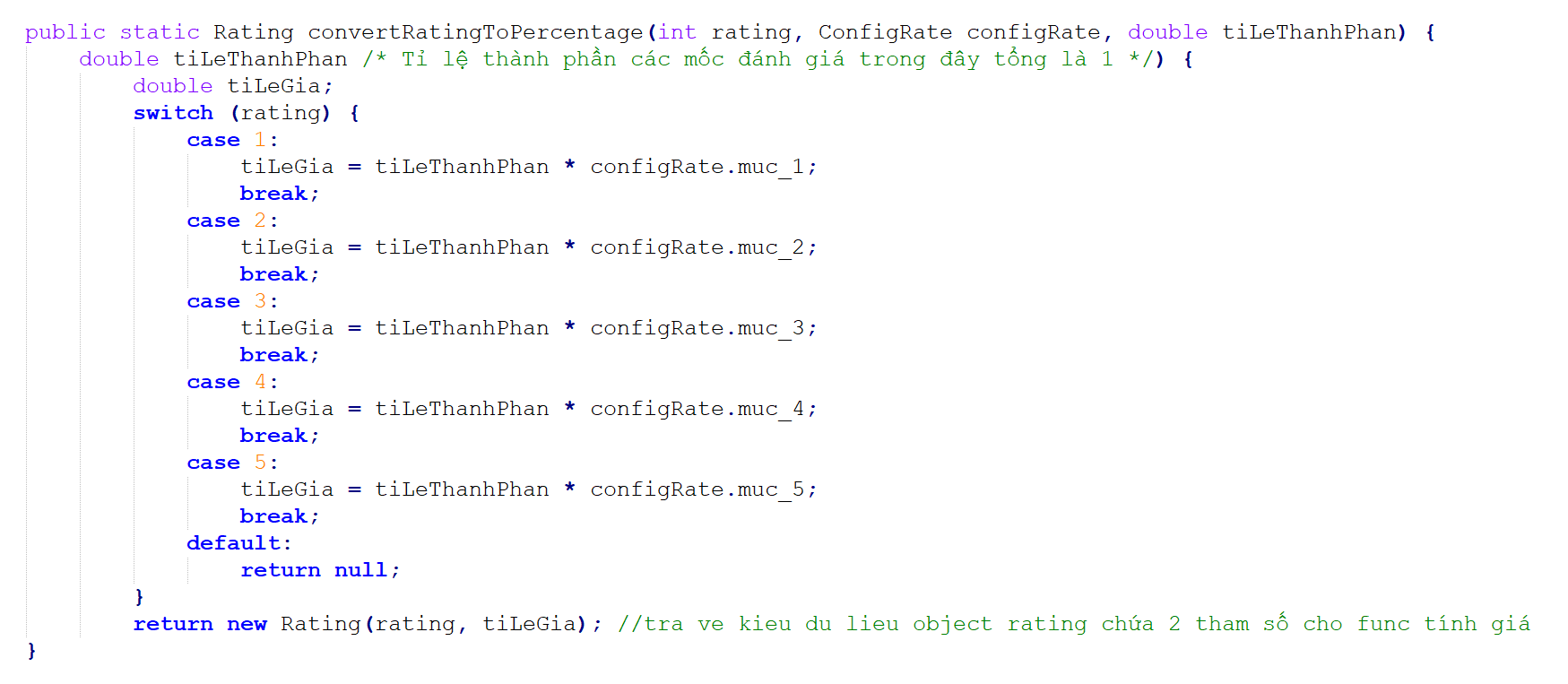
1. Người dùng nhập dữ liệu vào các trường nhập liệu.
2. Khi người dùng nhấp vào nút "Submit", ứng dụng sẽ:
   * Kiểm tra xem người dùng đã nhập đầy đủ thông tin hay chưa.
   * Nếu người dùng chưa nhập đầy đủ thông tin, ứng dụng sẽ hiển thị thông báo lỗi.
   * Nếu người dùng đã nhập đầy đủ thông tin, ứng dụng sẽ gửi dữ liệu cho cơ sở dữ liệu DynamoDB.
   * Hiển thị thông báo thành công.
   * Mở màn hình menu khách hàng.
3. Khi người dùng nhấp vào nút "Back", ứng dụng sẽ quay lại màn hình trước.

-Mục đích

Mục đích của ứng dụng này là thu thập thông tin từ người dùng về các sản phẩm cần được tái chế và gửi thông tin đó cho cơ sở dữ liệu DynamoDB. Dữ liệu này có thể được sử dụng để theo dõi và quản lý quá trình tái chế sản phẩm.

## Mã giả thuật toán

### Mã Giả cho convertRatingToPercentage



*9. Mã giả cho hàm convertRatingToPercentage*

**Input:**

“rating’: Giá trị đánh giá (1 đến 5).

“configRate”: Cấu hình các giá trị tỷ lệ cho mỗi mức đánh giá.

“tiLeThanhPhan”: Tỷ lệ thành phần, tổng của các tỷ lệ thành phần là 1.

**Output:**

Đối tượng Rating chứa rating (chỉ số) và ratePrice (tỷ lệ giá).

**Algorithm:**

Bắt đầu hàm “convertRatingToPercentage”.

Khởi tạo “tiLeGia” là biến double.

Sử dụng câu lệnh switch trên rating:

Nếu “rating” là 1: Gán “tiLeGia” bằng “tiLeThanhPhan” nhân với đối tượng “configRate.muc\_1”.

Nếu “rating” là 2: Gán “tiLeGia” bằng “tiLeThanhPhan” nhân với đối tượng “configRate.muc\_2”.

Nếu “rating” là 3: Gán “tiLeGia” bằng “tiLeThanhPhan” nhân với đối tượng “configRate.muc\_3”.

Nếu “rating” là 4: Gán “tiLeGia” bằng “tiLeThanhPhan” nhân với đối tượng “configRate.muc\_4”.

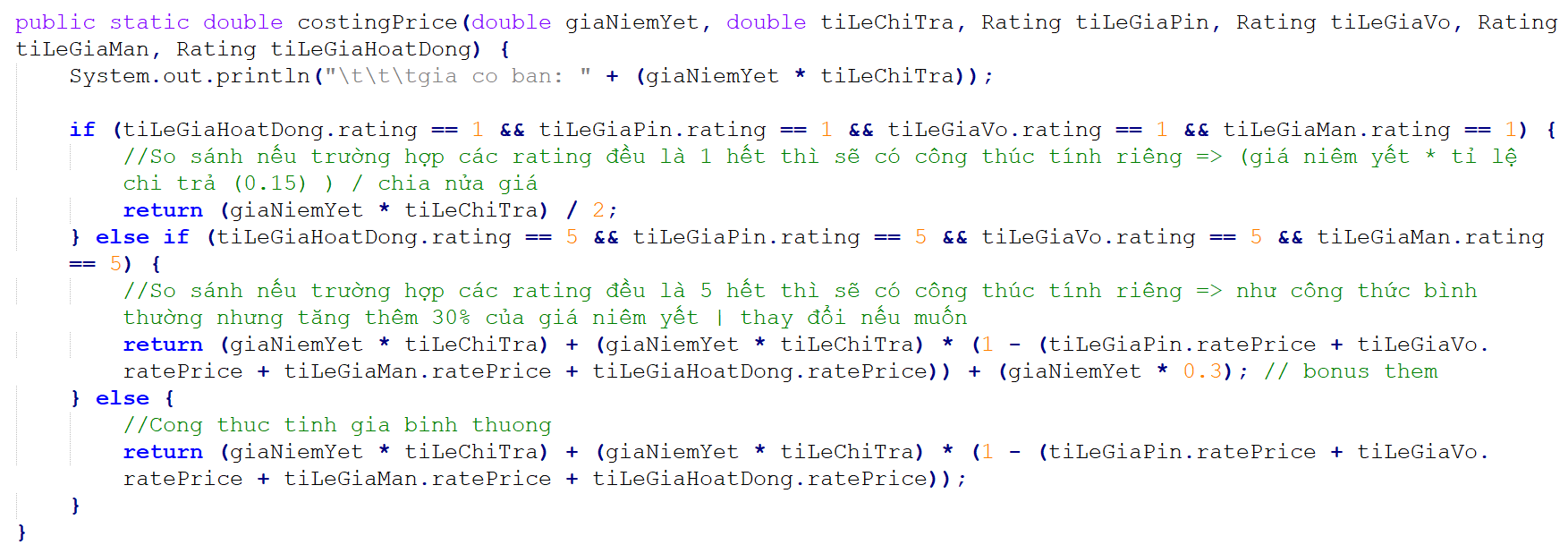
Nếu “rating” là 5: Gán “tiLeGia” bằng “tiLeThanhPhan” nhân với đối tượng “configRate.muc\_5”.

default: Trả về null.

Trả về đối tượng “Rating” gồm “rating” và “tiLeGia”.

Kết thúc hàm “convertRatingToPercentage”.

### Mã Giả cho costingPrice



*10. Mã giả cho hàm costingPrice*

**Input:**

“giaNiemYet”: Giá niêm yết của sản phẩm điện thoại lấy từ nhập hoặc danh sách sản phẩm từ bảng “Products”.

“tiLeChiTra”: Tỉ lệ phần trăm chi trả cơ bản cho một sản phẩm.

“tiLeGiaPin”, “tiLeGiaVo”, “tiLeGiaMan”, “tiLeGiaHoatDong”: Các đối tượng thuộc kiểu “Rating” chứa thông tin đánh giá và tỷ lệ giá của các thành phần.

**Output:**

Trả về finalPrice: Giá chỉ trả cho sản phẩm điện thoại.

**Algorithm:**

Bắt đầu hàm “costingPrice”.

Kiểm tra điều kiện đặc biệt:

Nếu tất cả các đánh giá (“tiLeGiaHoatDong”, “tiLeGiaPin”, “tiLeGiaVo”, “tiLeGiaMan”) đều là 1:

Trả về “(giaNiemYet \* tiLeChiTra) / 2”.

Nếu tất cả các đánh giá đều là 5:

Tính giá bằng công thức: “(giaNiemYet \* tiLeChiTra) + (giaNiemYet \* tiLeChiTra) \* (1 - (tiLeGiaPin.ratePrice + tiLeGiaVo.ratePrice + tiLeGiaMan.ratePrice + tiLeGiaHoatDong.ratePrice)) + (giaNiemYet \* 0.3)”.

Nếu không thuộc trường hợp đặc biệt:

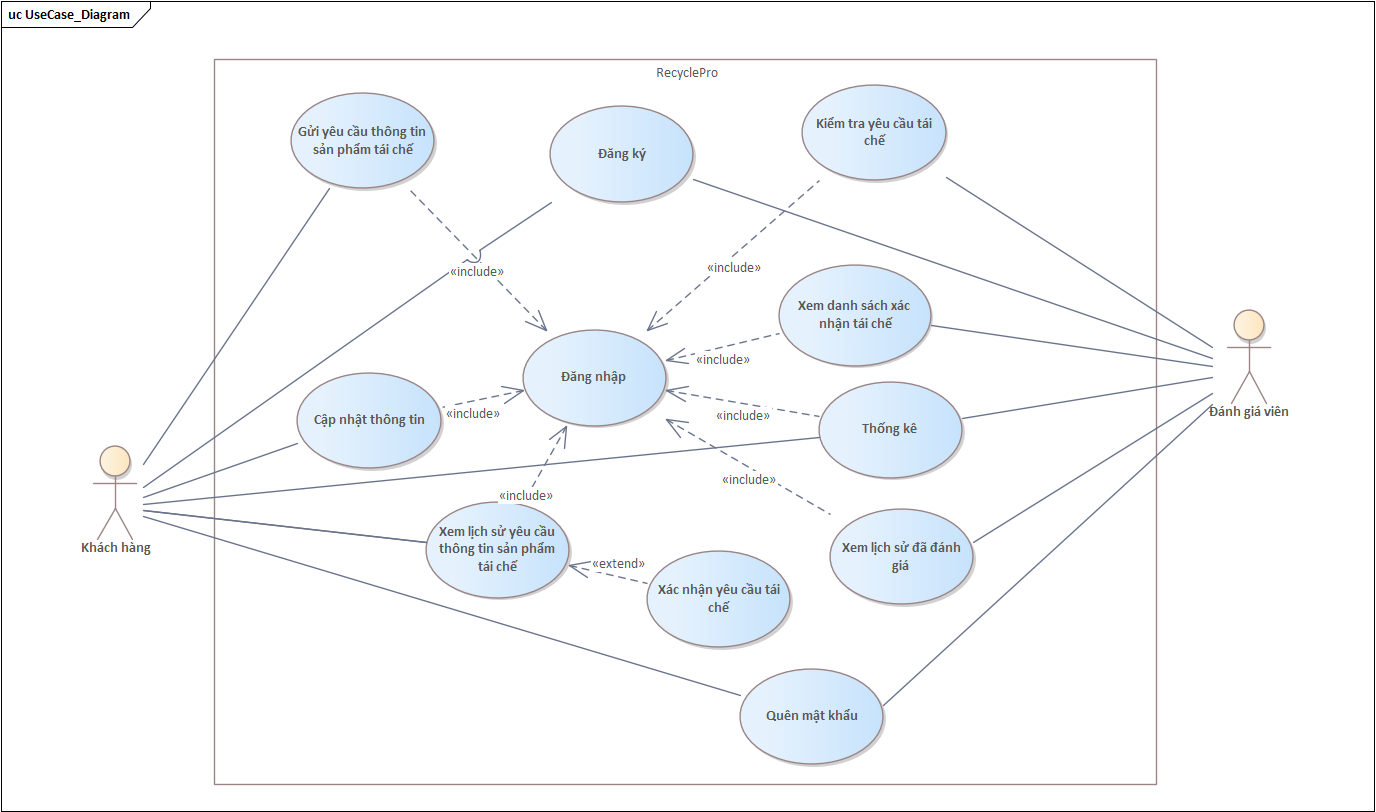
Tính giá bằng công thức: “(giaNiemYet \* tiLeChiTra) + (giaNiemYet \* tiLeChiTra) \* (1 - (tiLeGiaPin.ratePrice + tiLeGiaVo.ratePrice + tiLeGiaMan.ratePrice + tiLeGiaHoatDong.ratePrice))”.

Trả về giá cuối cùng.

Kết thúc hàm “costingPrice”.

## Phân tích yêu cầu bằng UML

### Usecase tổng quát



*11. Usecase tổng quát*

### Danh sách tác nhân và mô tả

Danh sách tác nhân và mô tả xem ở Bảng 3-1.

|  |  |
| --- | --- |
| Tác nhân | Mô tả tác nhân |
| Khách hàng | Thực hiện gửi các sản phẩm, xem lịch sử, thống kê các sản phẩm đã gửi, xác nhận tái chế. |
| Đánh giá viên | Thực hiện các đánh giá được gửi lên từ khách hàng để báo giá, xem sản phẩm tái chế, thống kê. |

*Bảng 3.1 Danh sách tác nhân và mô tả*

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

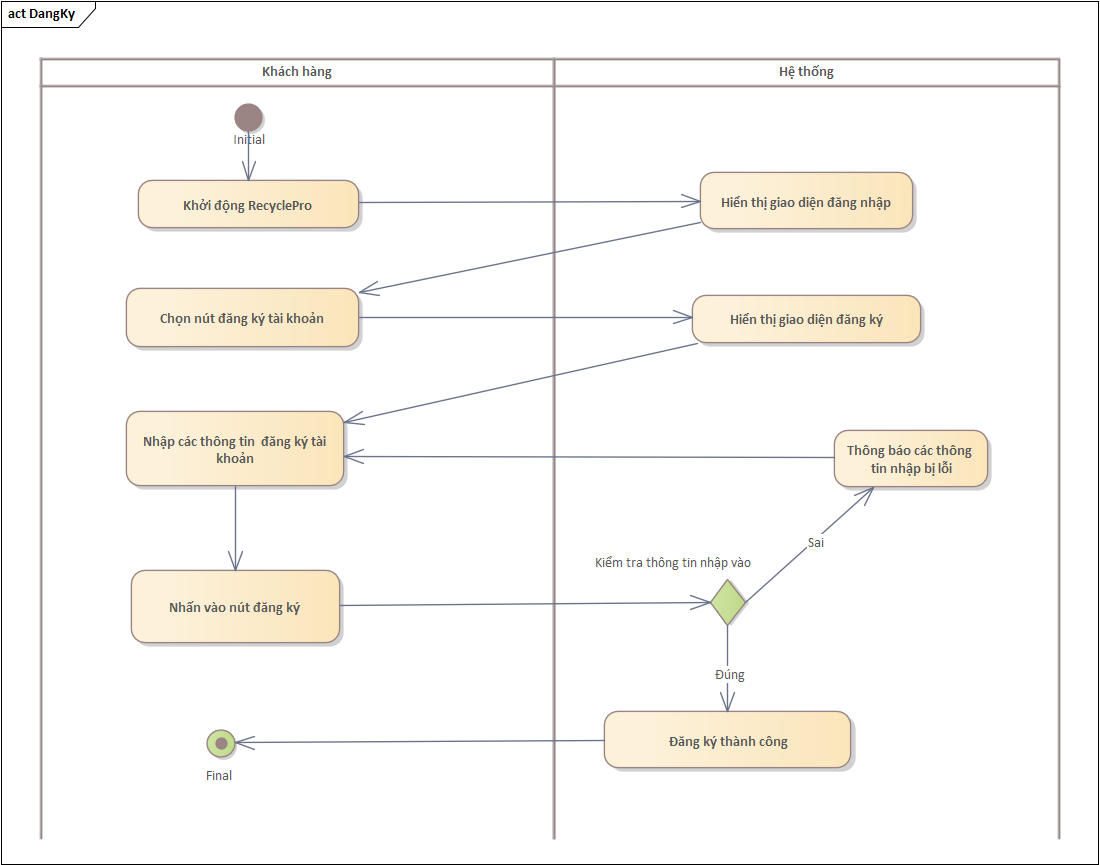
|  |  |
| --- | --- |
| ID | Tên Use case |
| UC01 | Đăng ký |
| UC02 | Đăng nhập |
| UC03 | Quên mật khẩu |
| UC04 | Gửi yêu cầu thông tin sản phẩm tái chế |
| UC05 | Xem lịch sử yêu cầu thông tin sản phẩm tái chế |
| UC06 | Xem lịch sử đã đánh giá |
| UC07 | Kiểm tra yêu cầu tái chế |
| UC08 | Xem danh sách xác nhận tái chế |
| UC09 | Thống kê của khách hàng |
| UC010 | Thống kê của đánh giá viên |
| UC011 | Cập nhật thông tin tài khoản |

*Bảng 3.2 Danh sách các tình huống trong hệ thống*

### Tình huống hoạt động

#### UC01\_Đăng ký

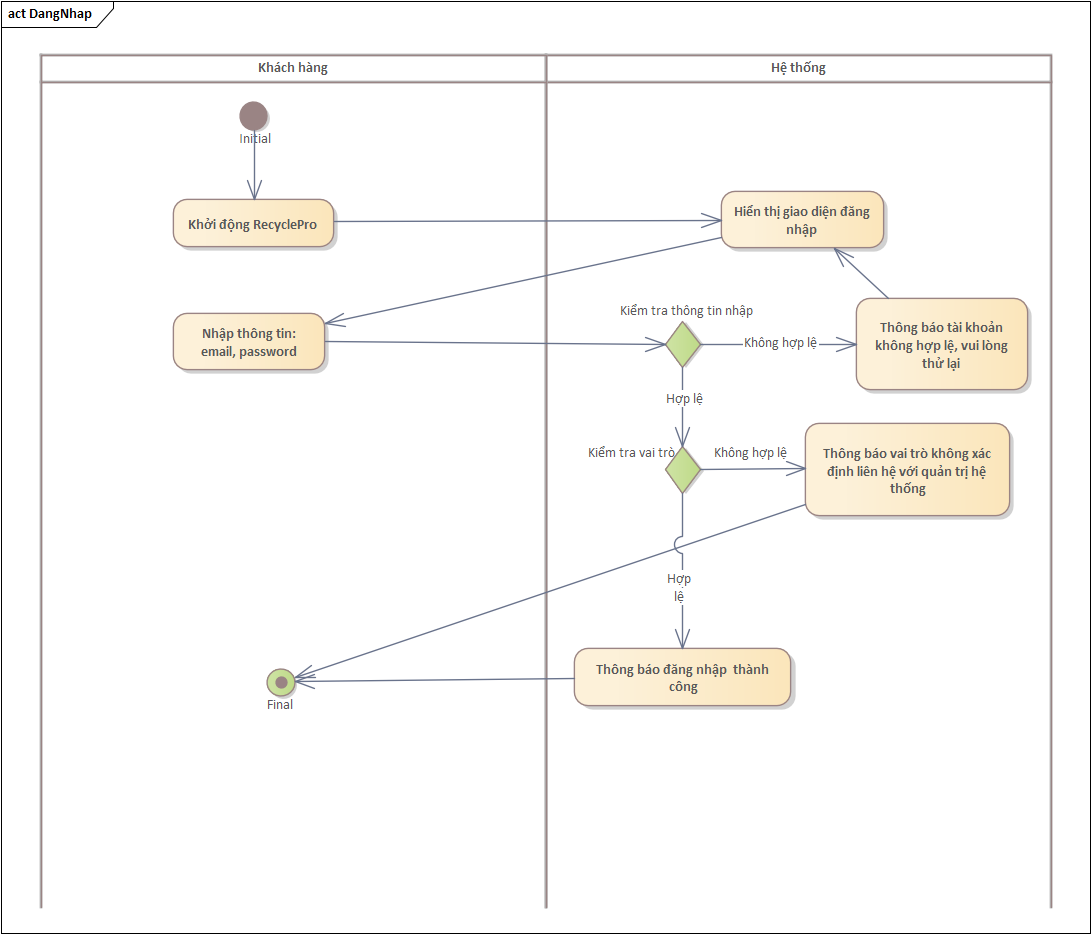
|  |  |
| --- | --- |
| * **Tên use case**: UC01\_Đăng ký | |
| * **Mô tả sơ lược**: Đăng ký tài khoản của khách hàng trước khi sử dụng các dịch vụ của RecyclePro | |
| * **Actor chính**: Khách hàng | |
| * **Actor phụ**: Không có | |
| * **Tiền điều kiện (Pre-condition):** Chưa có tài khoản sử dụng dịch vụ RecyclePro, có kết nối mạng. | |
| * **Hậu điều kiện (Post-condition):** Tài khoản đăng ký thành công sẽ được lưu vào cơ sở dữ liệu. | |
| * **Luồng sự kiện chính (main flow):** | |
| **Khách hàng** | **Hệ thống** |
| 1. Khởi động ứng dụng RecyclePro. |  |
|  | 1. Hiển thị giao diện đăng nhập. |
| 1. Chọn nút đăng ký tài khoản. |  |
|  | 1. Hiển thị giao diện đăng ký tài khoản. |
| 1. Nhập các thông tin: tên người dùng(username), địa chỉ mail (email), mật khẩu(password) và nhập lại mật khẩu (confirm password). |  |
| 1. Nhấn vào nút đăng ký (signup) |  |
|  | 1. Kiểm tra các thông tin nhập vào. Nếu hợp lệ thông báo thành công. |
| 1. Xác nhận |  |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | 7.1. Nếu không hợp lệ thông báo trường nhập vào nào bị lỗi. |
| 7.2. Người dùng nhập lại thông tin. |  |
| 7.3. Quay lại bước 7. |  |



*1. Sơ đồ hoạt động đăng ký*

#### UC02\_Đăng nhập

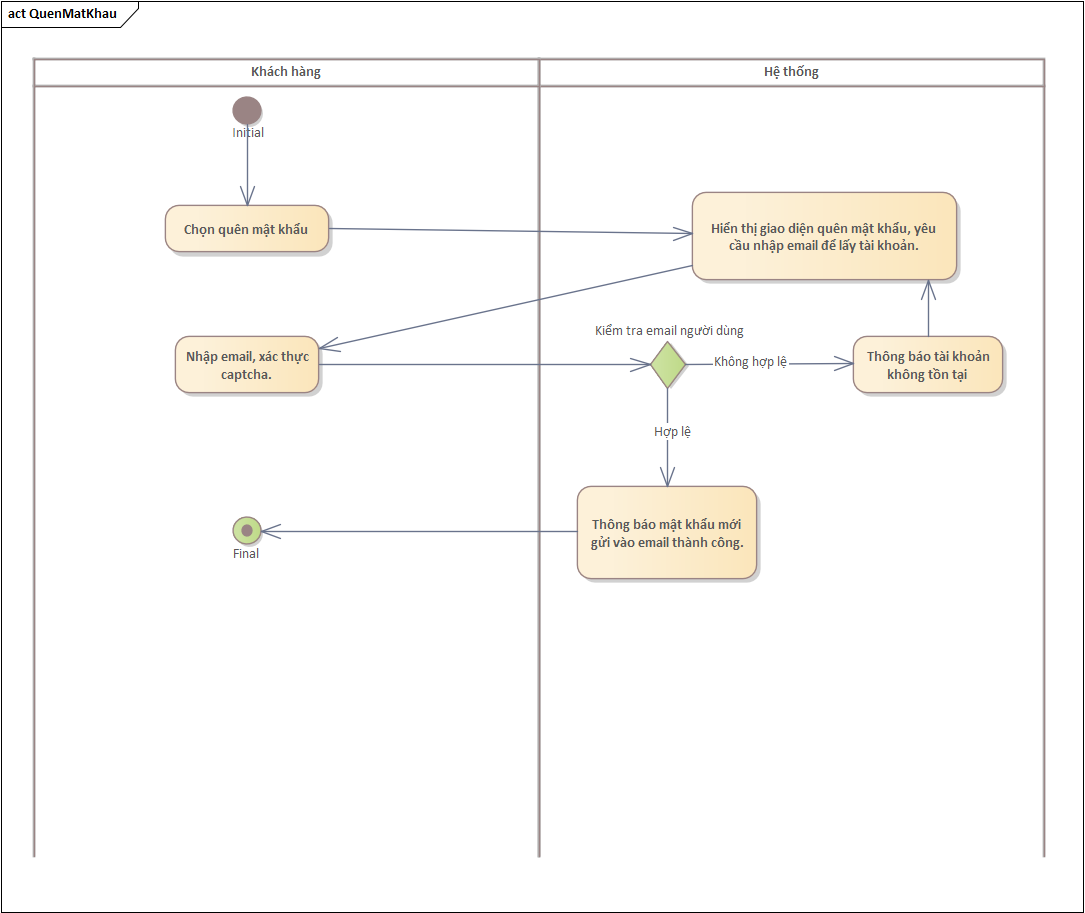
|  |  |
| --- | --- |
| * **Tên use case**: UC02\_Đăng nhập | |
| * **Mô tả sơ lược**: Người dùng muốn đăng nhập vào hệ thống để sử dụng dịch vụ của RecyclePro | |
| * **Actor chính**: Người dùng | |
| * **Actor phụ**: Không có | |
| * **Tiền điều kiện (Pre-condition):** Kết nối internet, người dùng đã có tài khoản | |
| * **Hậu điều kiện (Post-condition):** | |
| * **Luồng sự kiện chính (main flow):** | |
| **Khách hàng** | **Hệ thống** |
| 1. Khởi động RecyclePro |  |
|  | 1. Hiển thị giao diện đăng nhập |
| 1. Nhập vào các thông tin: Địa chỉ mail (email), mật khẩu (password) |  |
|  | 1. Kiểm tra thông tin nhập vào. |
|  | 1. Kiểm tra vai trò |
|  | 1. Thông báo đăng nhập thành công |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | 4.1. Thông báo thông tin đăng nhập không đúng. |
|  | 4.2. Quay lại bước 2 |
|  | 5.1. Thông báo vai trò không hợp lệ |
|  | 5.2. Quay lại bước 2 |



*2. Sơ đồ hoạt động đăng nhập.*

#### UC03\_Quên mật khẩu

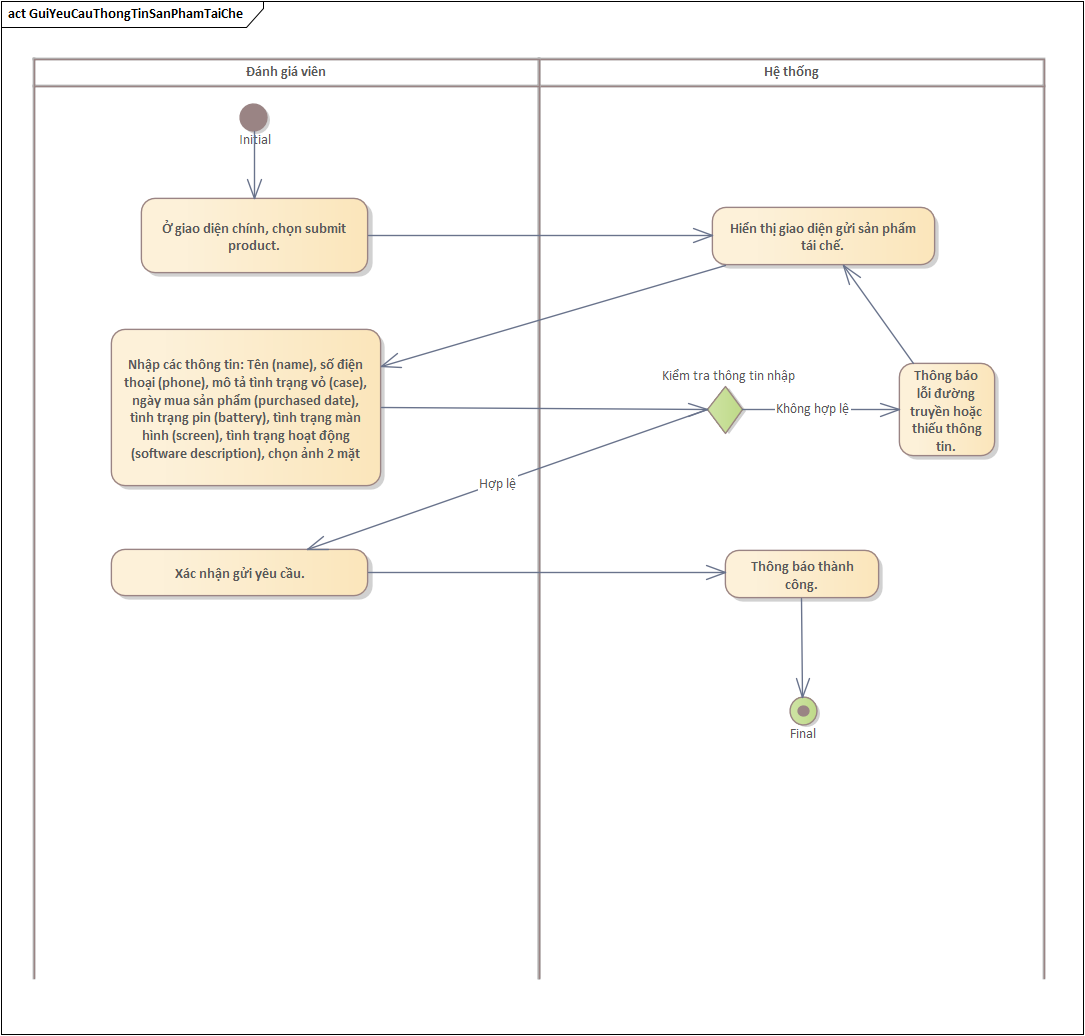
|  |  |
| --- | --- |
| * **Tên use case**: UC03\_Quên mật khẩu | |
| * **Mô tả sơ lược**: Người dùng nhập email sử dụng đăng ký để lấy lại mật khẩu thông qua email. | |
| * **Actor chính**: Khách hàng, Đánh giá viên | |
| * **Actor phụ**: Không có | |
| * **Tiền điều kiện (Pre-condition):** Đã có tài khoản trong hệ thống | |
| * **Hậu điều kiện (Post-condition):** Thông tin mật khẩu đã được gửi tới email người yêu cầu. | |
| * **Luồng sự kiện chính (main flow):** | |
| **Người dùng** | **Hệ thống** |
| 1. Chọn quên mật khẩu tại giao diện đăng nhập. |  |
|  | 1. Hiển thị giao diện quên mật khẩu, yêu cầu nhập email để lấy tài khoản. |
| 1. Nhập email, xác thực captcha. |  |
|  | 1. Kiểm tra email người dùng. |
|  | 1. Thông báo mật khẩu mới gửi vào email thành công. |
|  | 1. Sau 10s chuyển sang giao diện đăng nhập. |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | 4.1. Thông báo tài khoản không tồn tại. |
|  | 4.2. Quay lại bước 2. |
|  |  |



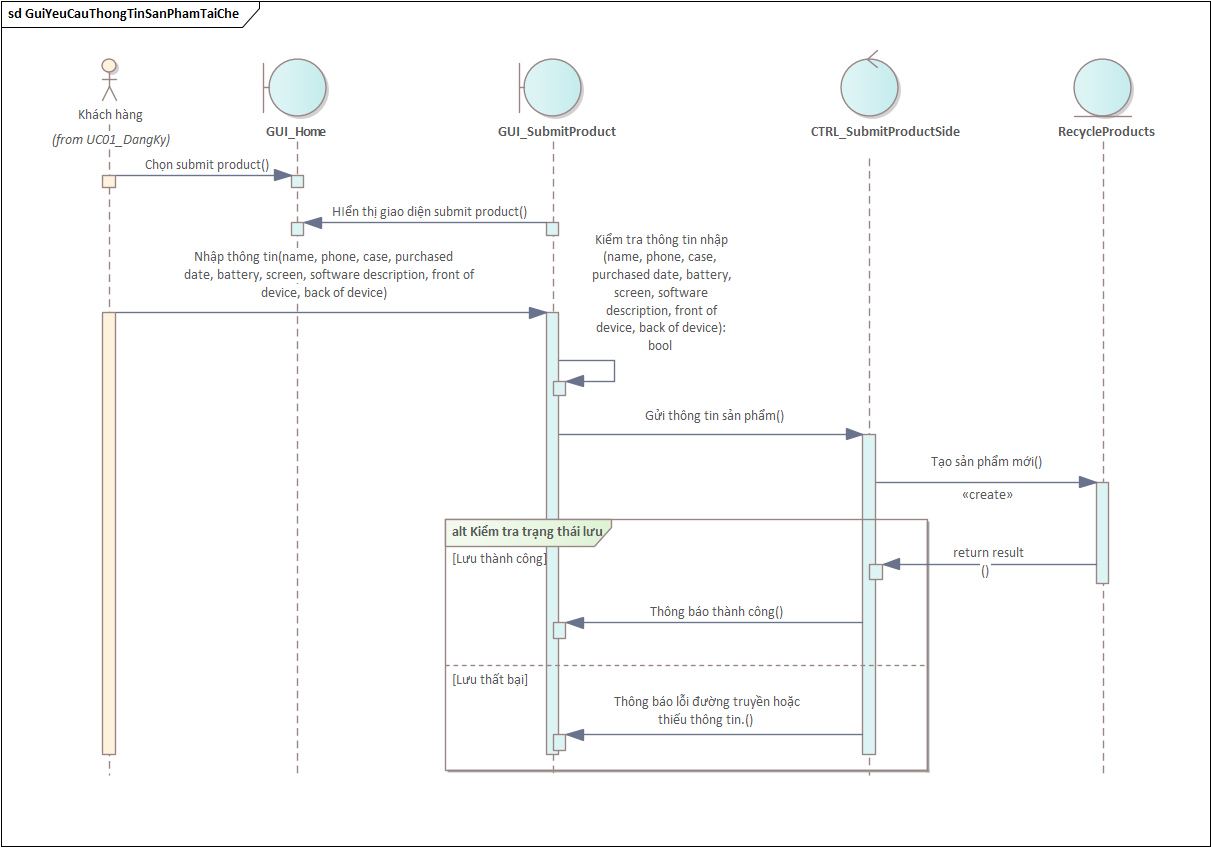
*4. Sơ đồ hoạt động quên mật khẩu.*

#### UC04\_ Gửi yêu cầu thông tin sản phẩm tái chế

|  |  |
| --- | --- |
| * **Tên use case**: UC04\_ Gửi yêu cầu thông tin sản phẩm tái chế | |
| * **Mô tả sơ lược**: Khách hàng có nhu cầu muốn xem xét giá tái chế sản phẩm. | |
| * **Actor chính**: Khách hàng | |
| * **Actor phụ**: Không có | |
| * **Tiền điều kiện (Pre-condition):** Có tài khoản khách hàng của hệ thống và có kết nối mạng. | |
| * **Hậu điều kiện (Post-condition):** Thông tin sản phẩm yêu cầu được lưu vào cơ sở dữ liệu. | |
| * **Luồng sự kiện chính (main flow):** | |
| **Khách hàng** | **Hệ thống** |
| 1. Ở giao diện chính, chọn submit product. |  |
|  | 1. Hiển thị giao diện gửi sản phẩm tái chế. |
| 1. Nhập các thông tin: Tên người gửi (name), số điện thoại (phone), mô tả tình trạng vỏ (case), ngày mua sản phẩm (purchased date), tình trạng pin (battery), tình trạng màn hình (screen), tình trạng hoạt động (software description), chọn ảnh 2 mặt trước và sau của thiết bị (front of device, back of device). |  |
|  | 1. Kiểm tra thông tin nhập. |
| 1. Xác nhận gửi yêu cầu. |  |
|  | 1. Thông báo thành công. |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | * 1. Thông báo lỗi đường truyền hoặc thiếu thông tin. |
|  | * 1. Quay lại bước 2. |



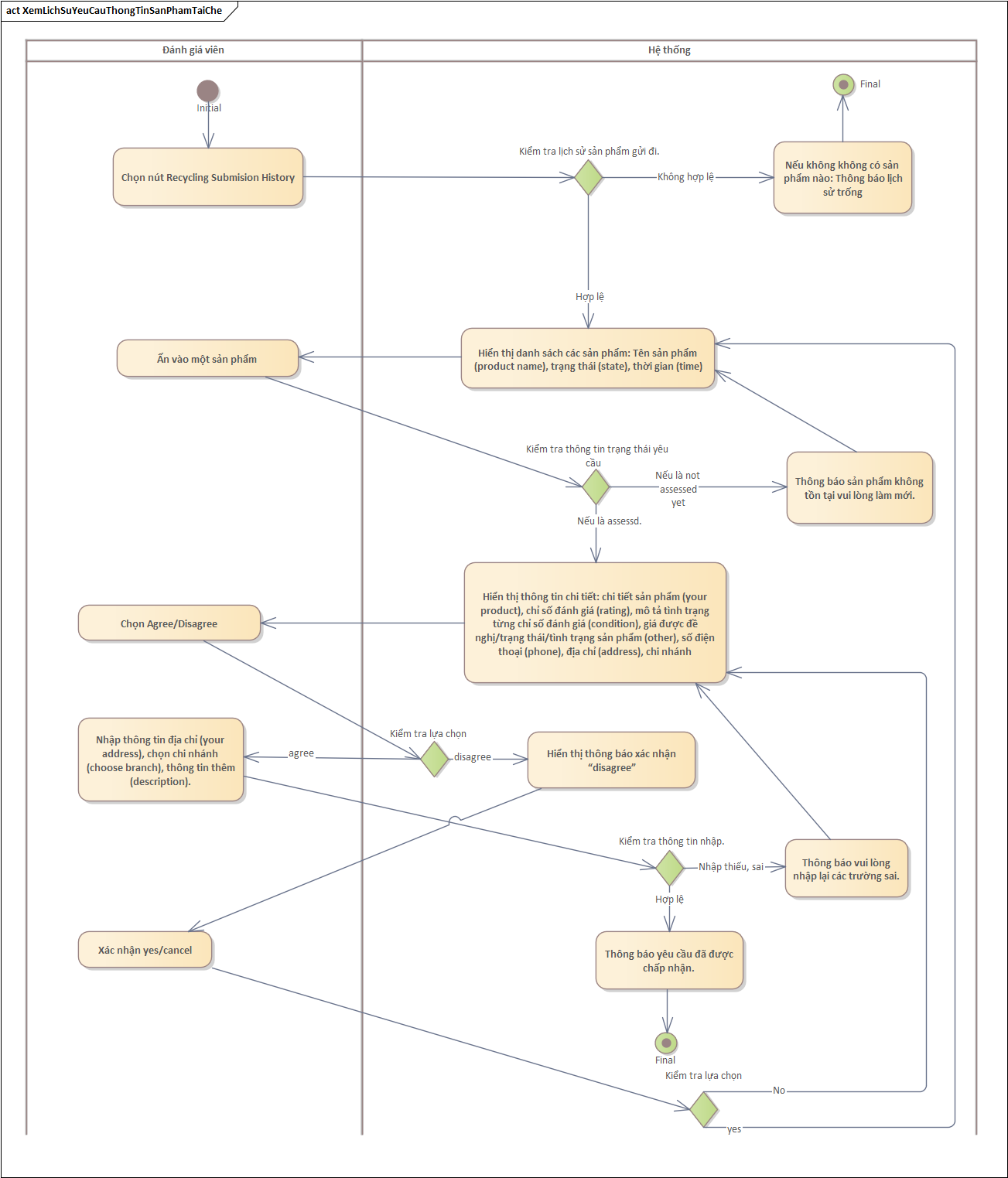
*5. Sơ đồ hoạt động gửi yêu cầu thông tin sản phẩm tái chế.*



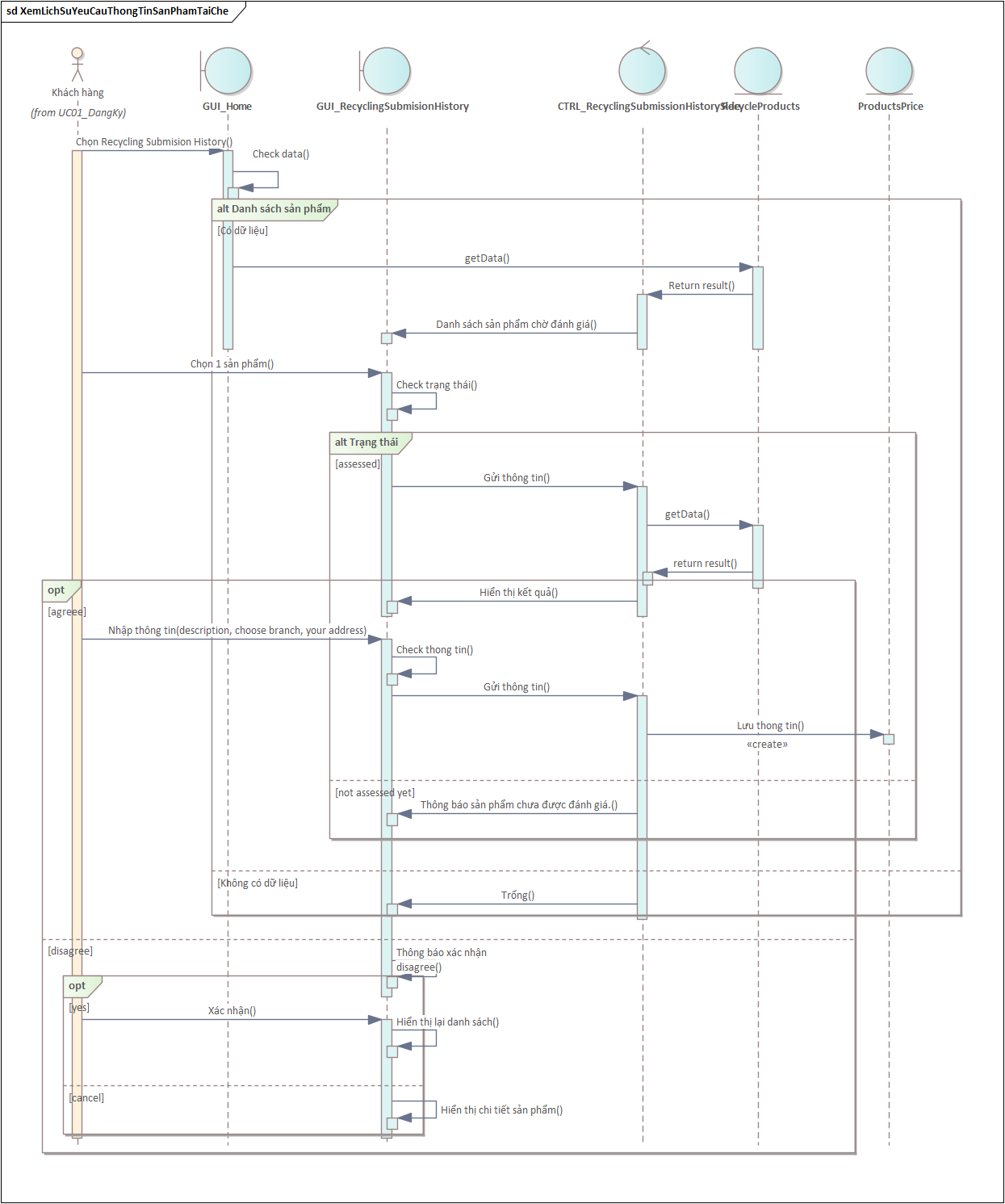
*1. Sơ đồ tuần tự gửi yêu cầu thông tin sản phẩm tái chế*

#### UC05\_Xem lịch sử yêu cầu thông tin sản phẩm tái chế

|  |  |
| --- | --- |
| * **Tên use case**: UC05\_Xem lịch sử yêu cầu thông tin sản phẩm tái chế | |
| * **Mô tả sơ lược**: Cho phép người dùng xem lịch các sản phẩm đã gửi | |
| * **Actor chính**: Khách hàng | |
| * **Actor phụ**: Không | |
| * **Tiền điều kiện (Pre-condition):** Đăng nhập vào hệ thống với vai trò khách hàng có kết nối mạng. | |
| * **Hậu điều kiện (Post-condition):** Thông tin chi tiết sản phẩm được hiển thị ra màn hình | |
| * **Luồng sự kiện chính (main flow):** | |
| **Khách hàng** | **Hệ thống** |
| 1. Chọn nút Recycling Submision History |  |
|  | 1. Kiểm tra lịch sử sản phẩm gửi đi từ cơ sở dữ liệu. |
|  | 1. Hiển thị danh sách các sản phẩm: Tên sản phẩm (product name), trạng thái (state), thời gian (time) |
| 1. Ấn vào một sản phẩm |  |
|  | 1. Kiểm tra thông tin trạng thái yêu cầu: Nếu là assessd yet. |
|  | 1. Hiển thị thông tin chi tiết sản phẩm: chi tiết sản phẩm (your product), chỉ số đánh giá (rating), mô tả tình trạng từng chỉ số đánh giá (condition), giá được đề nghị/trạng thái/tình trạng sản phẩm (other), số điện thoại (phone), địa chỉ (address), chi nhánh (branchs) |
| 1. Chọn Agree/Disagree |  |
|  | 1. Kiểm tra lựa chọn: nếu là “Agree” |
| 1. Nhập thông tin địa chỉ (your address), chọn chi nhánh (choose branch), thông tin thêm (description). |  |
|  | 1. Kiểm tra thông tin nhập. |
|  | 1. Nếu đúng: Thông báo yêu cầu đã được chấp nhận. |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | * 1. Nếu không không có sản phẩm nào: Thông báo lịch sử trống. |
|  | * 1. Nếu trạng thái not assessed yet: Thông báo sản phẩm chưa được đánh giá. |
|  | * 1. Quay lại bước 3 |
|  | * 1. Nếu là “Disagree”: Hiển thị thông báo xác nhận “disagree”. |
| * 1. Xác nhận yes/cancel |  |
|  | * 1. Nếu yes: Quay lại bước 2. |
|  | * 1. Nếu cancel: Quay lại bước 5. |
|  | * 1. Nếu thông tin nhập sai: Thông báo vui lòng nhập lại các trường sai. |
|  | * 1. Quay lại bước 6. |



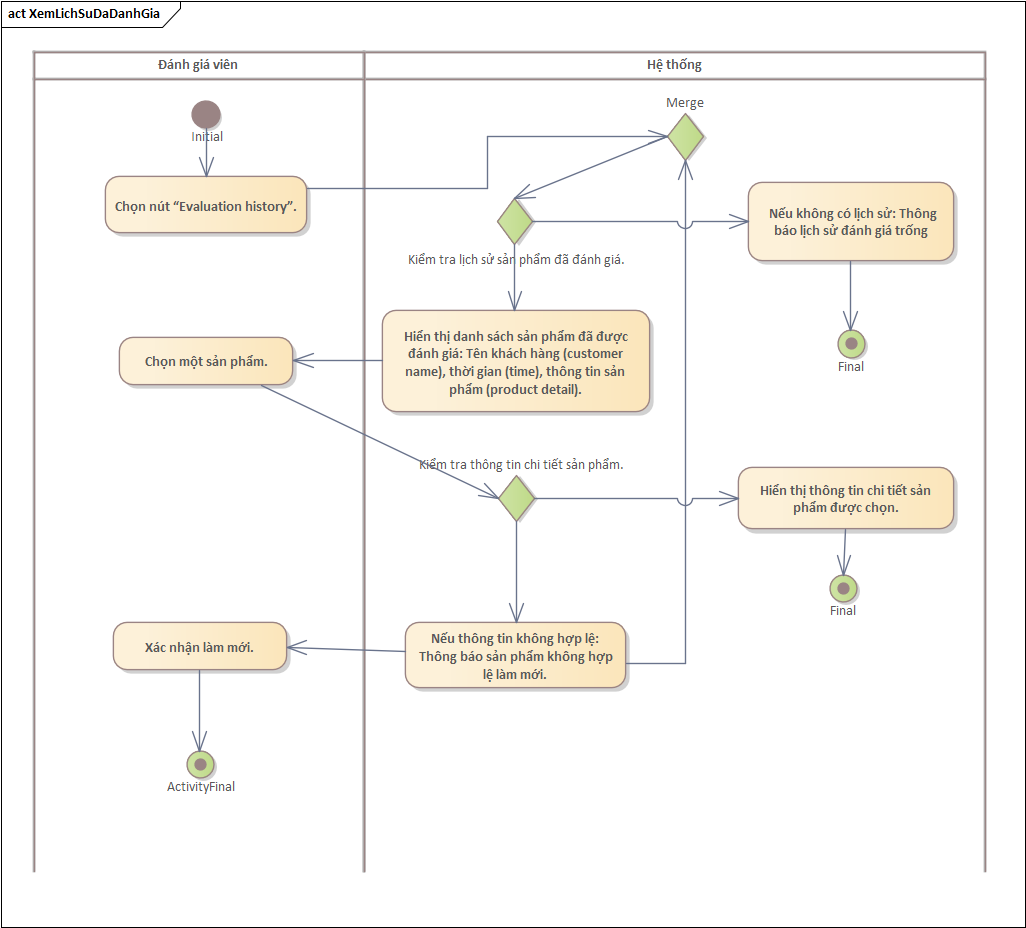
*6. Sơ đồ hoạt động xem lịch sử yêu cầu thông tin sản phẩm tái chế.*



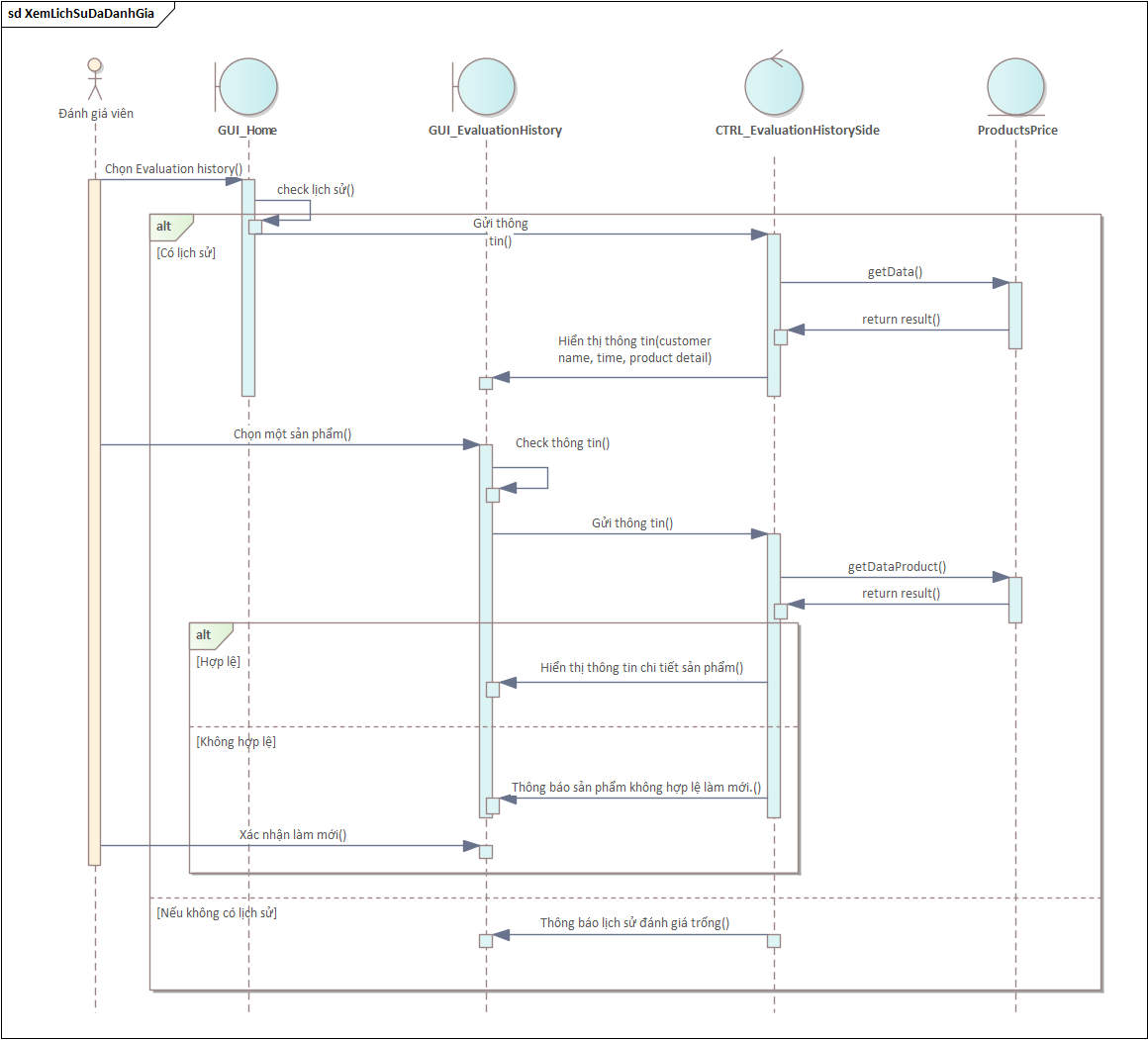
*2. Sơ đồ tuần tự xem lịch sử yêu cầu thông tin sản phẩm*

#### UC06\_Xem lịch sử đã đánh giá

|  |  |
| --- | --- |
| * **Tên use case**: UC06\_Xem lịch sử đã đánh giá | |
| * **Mô tả sơ lược**: Đánh giá viên có thể xem lịch sử các đánh giá, có thể phân loại theo thời gian. | |
| * **Actor chính**: Đánh giá viên | |
| * **Actor phụ**: Không | |
| * **Tiền điều kiện (Pre-condition):** Đăng nhập tài khoản người dùng với vai trò là đánh giá viên | |
| * **Hậu điều kiện (Post-condition):** Lấy được các thông tin các sản phẩm đã đánh giá | |
| * **Luồng sự kiện chính (main flow):** | |
| **Đánh giá viên** | **Hệ thống** |
| 1. Chọn nút “Evaluation history”. |  |
|  | 1. Kiểm tra lịch sử sản phẩm đã đánh giá. |
|  | 1. Hiển thị danh sách sản phẩm đã được đánh giá: Tên khách hàng (customer name), thời gian (time), thông tin sản phẩm (product detail). |
| 1. Chọn một sản phẩm. |  |
|  | 1. Kiểm tra thông tin chi tiết sản phẩm từ cơ sở dữ liệu. |
|  | 1. Hiển thị thông tin chi tiết sản phẩm được chọn. |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | 2.1. Nếu không có lịch sử: Thông báo lịch sử đánh giá trống |
|  | 5.1. Nếu thông tin không hợp lệ: Thông báo sản phẩm không hợp lệ làm mới. |
| 5.2. Xác nhận làm mới. |  |
|  | 5.3. Quay lại bước 2. |



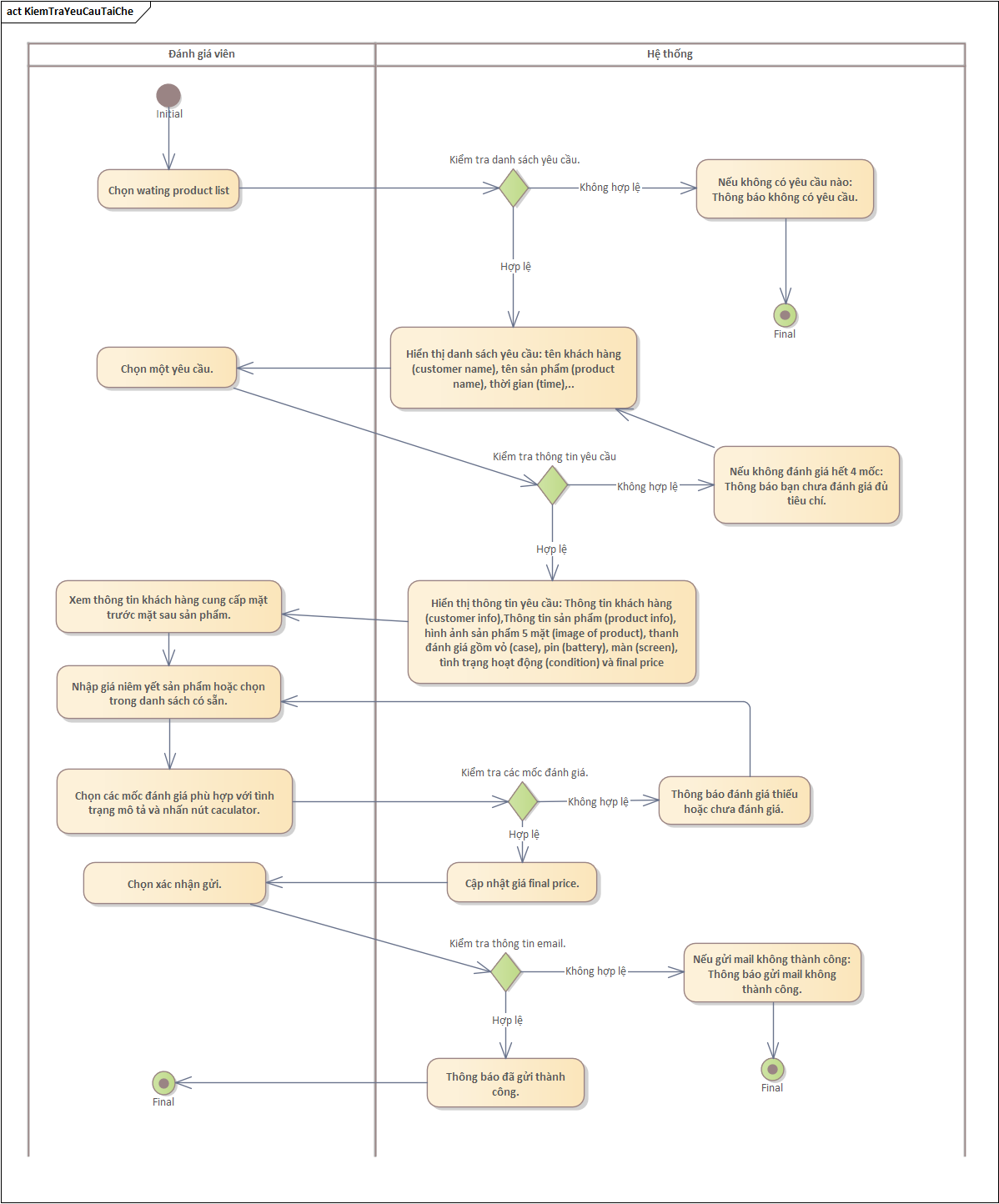
*7. Sơ đồ hoạt động xem lịch sử đánh giá.*



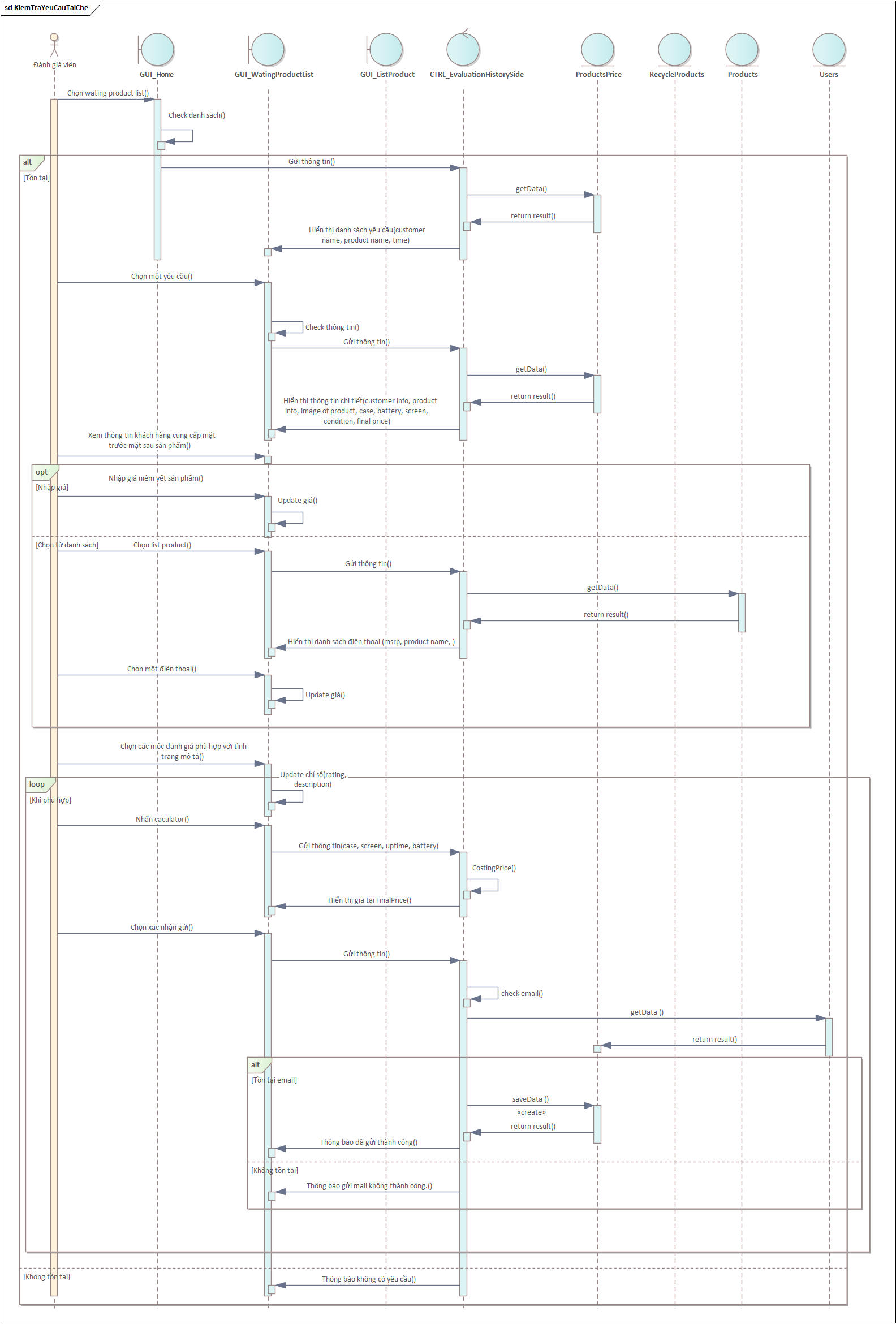
*3. Sơ đồ tuần tự xem lịch sử đã đánh giá.*

#### UC07\_Kiểm tra yêu cầu tái chế

|  |  |
| --- | --- |
| * **Tên use case**: UC07\_Kiểm tra yêu cầu tái chế | |
| * **Mô tả sơ lược**: Sau khi khách hàng gửi yêu cầu lên hệ thống thì đánh giá viên có nhiệm vụ vào xem thông tin sản phẩm so với hình ảnh và xác định giá sản phẩm. | |
| * **Actor chính**: Đánh giá viên | |
| * **Actor phụ**: Không | |
| * **Tiền điều kiện (Pre-condition):** Có kết nối mạng. | |
| * **Hậu điều kiện (Post-condition):** Lưu giá sản phẩm vào cơ sở dữ liệu và gửi mail thông báo cho khách hàng | |
| * **Luồng sự kiện chính (main flow):** | |
| **Đánh giá viên** | **Hệ thống** |
| 1. Chọn “wating product list”. |  |
|  | 1. Kiểm tra danh sách từ cơ sở dữ liệu. |
|  | 1. Hiển thị danh sách yêu cầu: tên khách hàng (customer name), tên sản phẩm (product name), thời gian (time),.. |
| 1. Chọn một yêu cầu. |  |
|  | 1. Kiểm tra thông tin yêu cầu |
|  | 1. Hiển thị chi tiết thông tin yêu cầu: Thông tin khách hàng (customer info),Thông tin sản phẩm (product info), hình ảnh sản phẩm 2 mặt (image of product) mặt trước và sau, thanh đánh giá gồm vỏ (case), pin (battery), màn (screen), tình trạng hoạt động (condition) và cuối cùng giá sản phẩm (final price). |
| 1. Xem thông tin khách hàng cung cấp mặt trước mặt sau sản phẩm. |  |
| 1. Nhập giá niêm yết sản phẩm hoặc chọn trong danh sách có sẵn. |  |
| 1. Chọn các mốc đánh giá phù hợp với tình trạng mô tả. |  |
| 1. Nhấn nút “caculator”. |  |
|  |  |
|  | 1. Kiểm tra các mốc đánh giá. |
|  | 1. Cập nhật giá final price. |
| 1. Chọn xác nhận gửi. |  |
|  | 1. Kiểm tra thông tin email. |
|  | 1. Thông báo đã gửi thành công. |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | * 1. Nếu không có yêu cầu nào: Thông báo không có yêu cầu. |
|  | * 1. Nếu không đánh giá hết 4 mốc: Thông báo bạn chưa đánh giá đủ tiêu chí. |
|  | * 1. Nếu gửi mail không thành công: Thông báo gửi mail không thành công. |
|  |  |



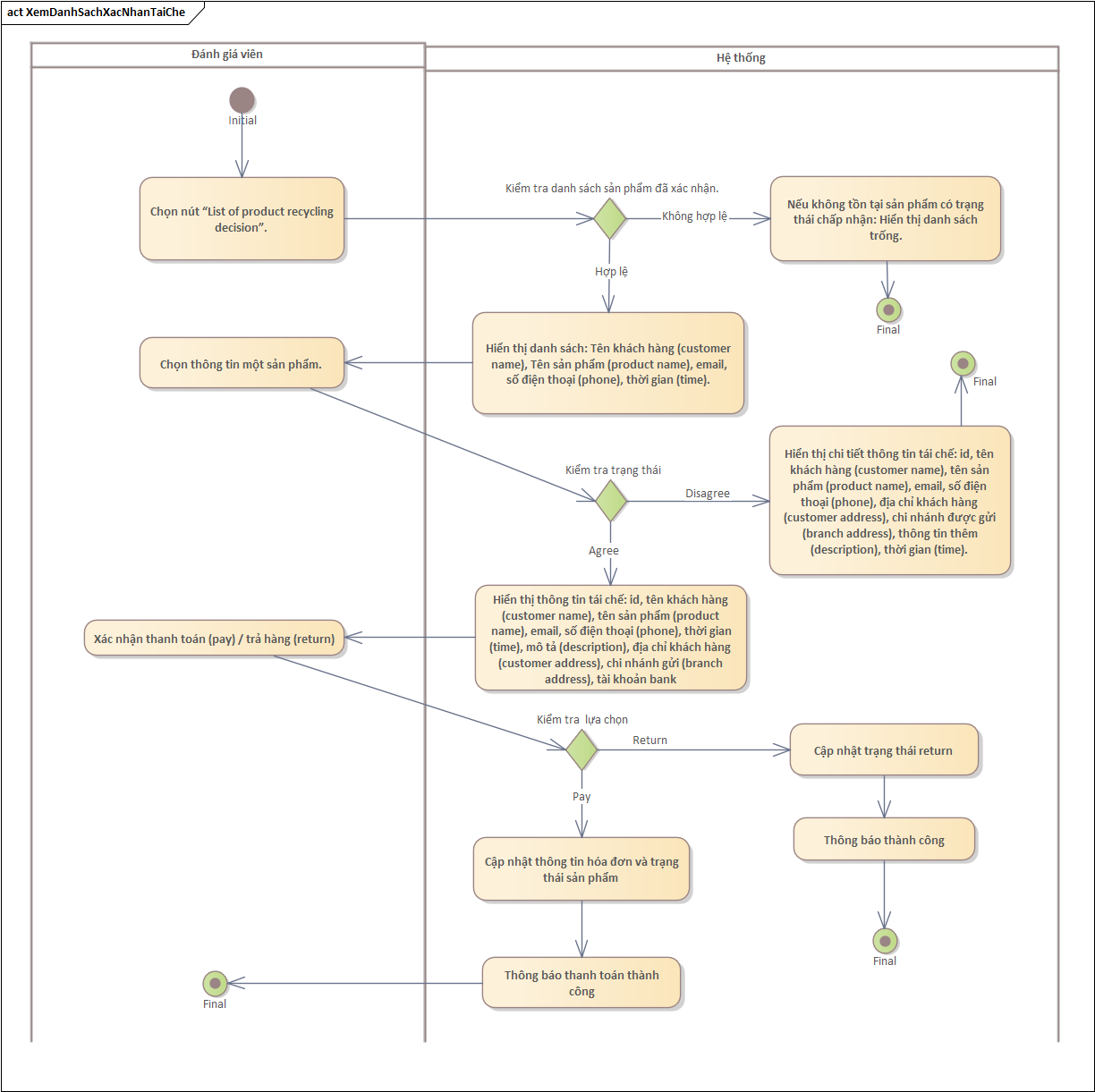
*8. Sơ đồ hoạt động kiểm tra yêu cầu tái chế.*



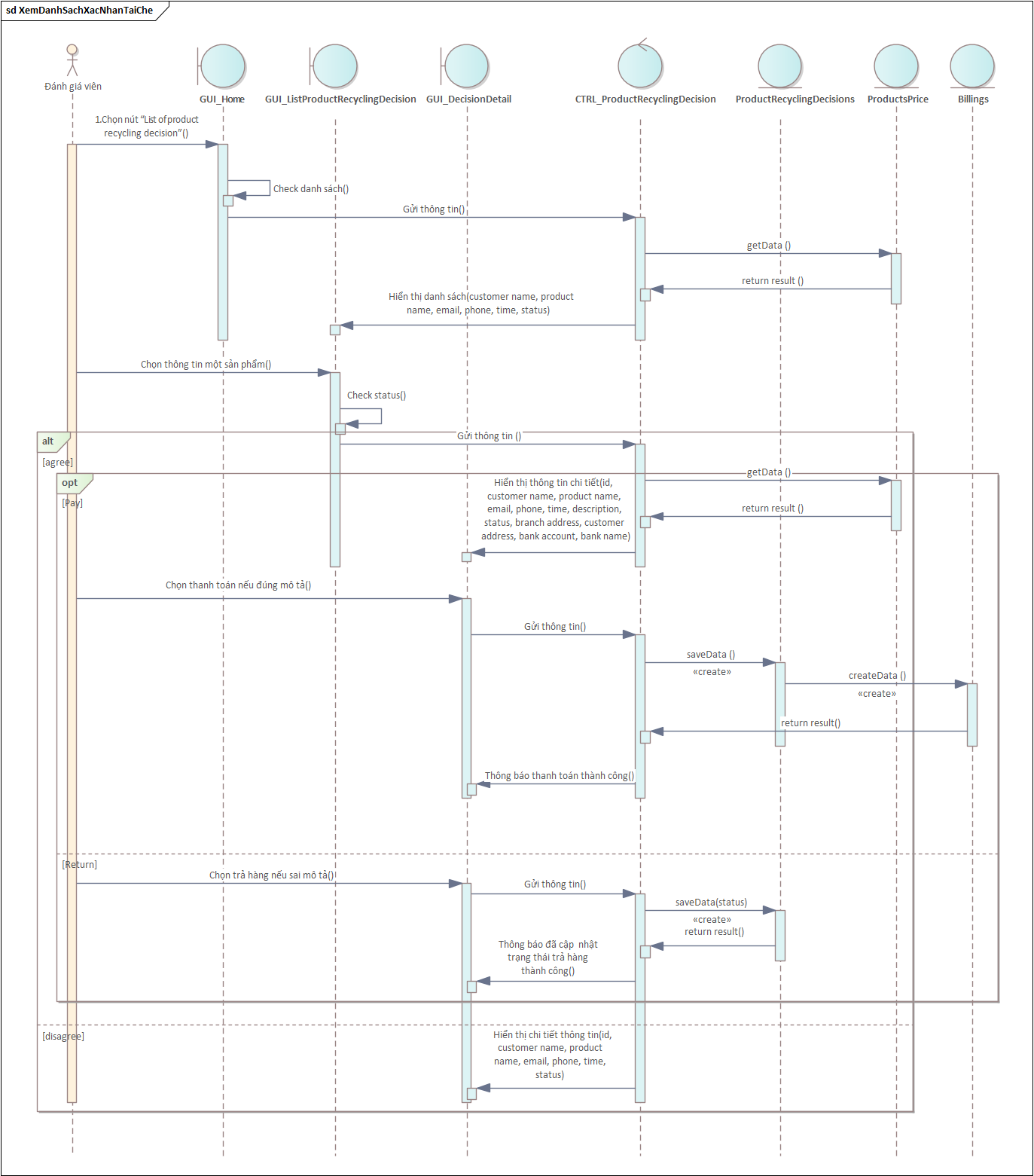
*4. Sơ đồ tuần tự kiểm tra yêu cầu tái chế.*

#### UC08\_Xem danh sách xác nhận tái chế

|  |  |
| --- | --- |
| * **Tên use case**: UC08\_Xem danh sách xác nhận tái chế | |
| * **Mô tả sơ lược**: Sau khi khách hàng chấp nhận mức giá và đánh giá viên nhận được hàng họ so so sánh với mô tả ban đầu. | |
| * **Actor chính**: Đánh giá viên | |
| * **Actor phụ**: Không | |
| * **Tiền điều kiện (Pre-condition):** Có kết nối mạng. | |
| * **Hậu điều kiện (Post-condition):** Thanh toán chi trả được ghi nhận hoặc trả sản phẩm được cập nhật vào cơ sở dữ liệu. | |
| * **Luồng sự kiện chính (main flow):** | |
| **Actor** | **System** |
| 1. Chọn nút “List of product recycling decision”. |  |
|  | 1. Kiểm tra danh sách yêu cầu từ cơ sở dữ liệu. |
|  | 1. Hiển thị danh sách: Tên khách hàng (customer name), Tên sản phẩm (product name), email, số điện thoại (phone), thời gian (time), trạngt thái (status). |
| 1. Chọn thông tin một sản phẩm. |  |
|  | 1. Kiểm tra trạng thái (status) yêu cầu. |
|  | 1. Nếu status là “agree”: Hiển thị chi tiết thông tin tái chế: id, tên khách hàng (customer name), tên sản phẩm (product name), email, số điện thoại (phone), thời gian (time), mô tả (description), địa chỉ khách hàng (customer address), chi nhánh gửi (branch address), tài khoản ngân hàng (bank account), tên ngân hàng (bank name). |
| 1. Xác nhận thanh toán (pay) / trả hàng (return) |  |
|  | 1. Kiểm tra lựa chọn. |
|  | 1. Nếu chọn xác nhận thanh toán (pay): Cập nhật thông tin hóa đơn và trạng thái sản phẩm. |
|  | 1. Thông báo thanh toán thành công. |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | * 1. Nếu không tồn tại sản phẩm có trạng thái chấp nhận: Hiển thị danh sách trống. |
|  | * 1. Nếu status là “disagree”: Hiển thị chi tiết thông tin tái chế: id, tên khách hàng (customer name), tên sản phẩm (product name), email, số điện thoại (phone), địa chỉ khách hàng (customer address), chi nhánh được gửi (branch address), thông tin thêm (description), thời gian (time). |
|  | * 1. Quay lại bước 2. |
|  | 8.1. Nếu chọn trả hàng (return): Cập nhật trạng thái return và thông báo thành công. |
|  | 8.2. Quay lại bước 3. |



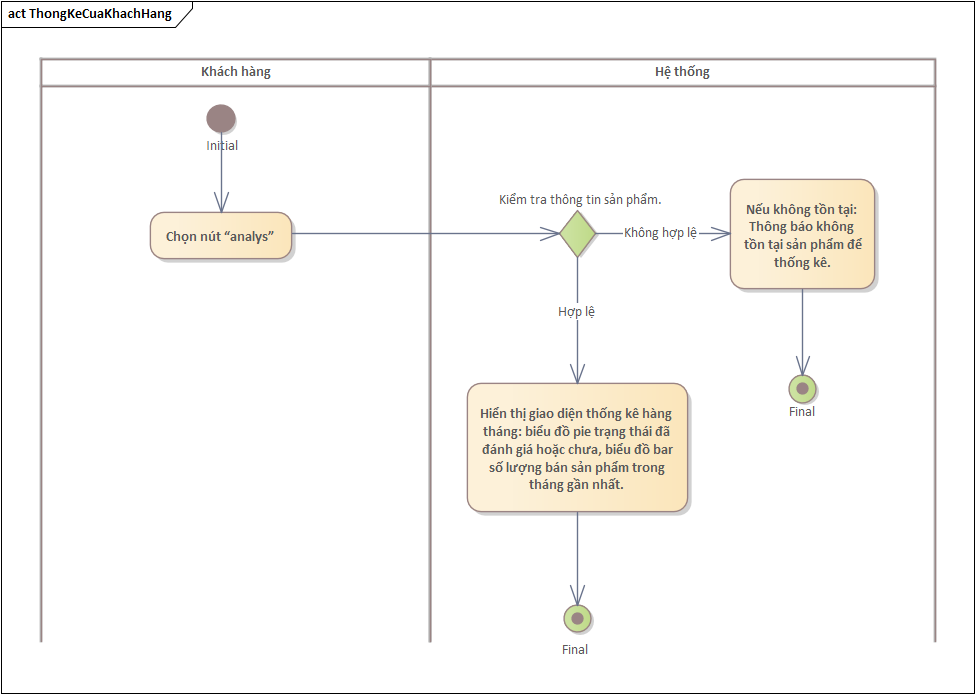
*9. Sơ đồ hoạt động xem danh sách xác nhận tái chế.*



*5. Sơ đồ tuần tự xem danh sách xác nhận tái chế.*

#### UC09\_Thống kê của khách hàng

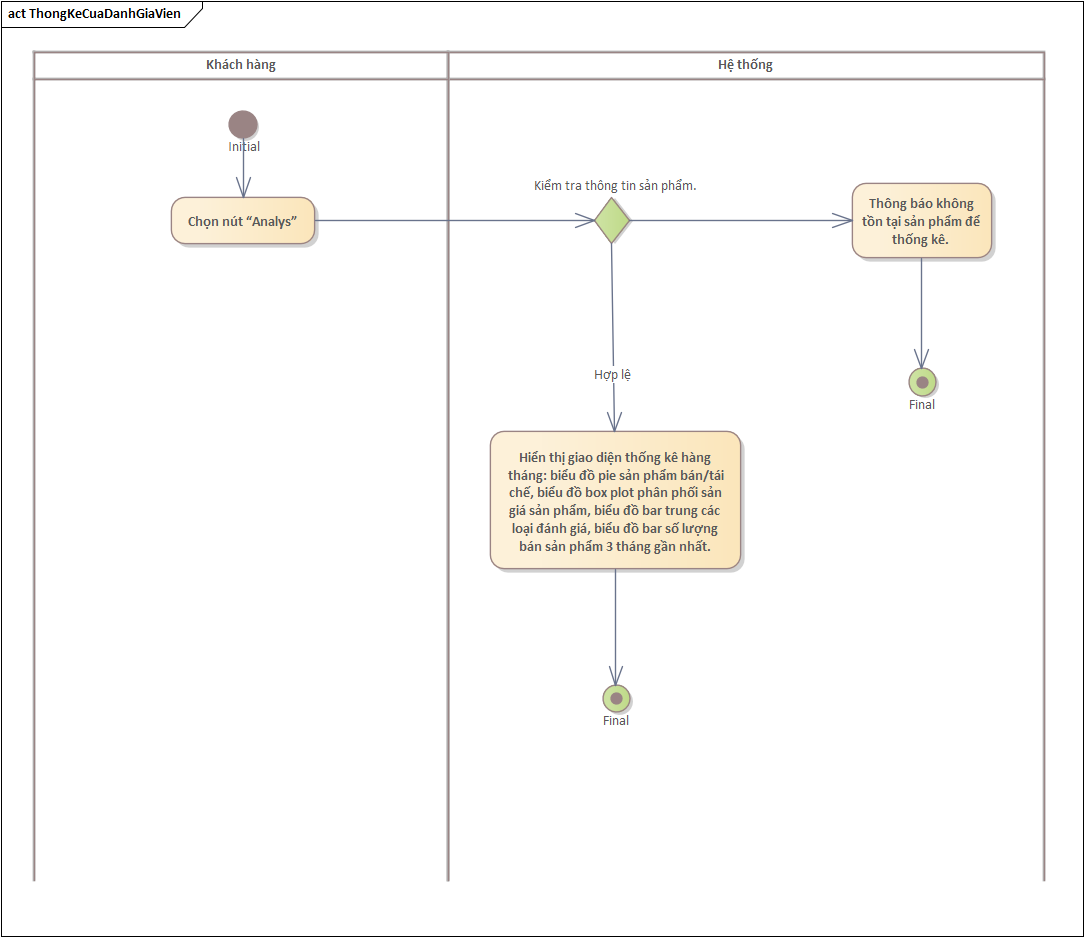
|  |  |
| --- | --- |
| * **Tên use case**: UC09\_Thống kê của khách hàng | |
| * **Mô tả sơ lược**: Lập biểu đồ để theo dõi. | |
| * **Actor chính**: Khách hàng | |
| * **Actor phụ**: Không | |
| * **Tiền điều kiện (Pre-condition):** Có kết nối mạng | |
| * **Hậu điều kiện (Post-condition):** | |
| * **Luồng sự kiện chính (main flow):** | |
| **Khách hàng** | **Hệ thống** |
| 1. Chọn nút “Analysis” |  |
|  | 1. Kiểm tra thông tin sản phẩm. |
|  | 1. Hiển thị giao diện thống kê hàng tháng: biểu đồ pie trạng thái đã đánh giá hoặc chưa, biểu đồ bar số lượng bán sản phẩm trong tháng gần nhất. |
|  |  |
|  |  |
|  |  |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | * 1. Nếu không tồn tại: Thông báo không tồn tại sản phẩm để thống kê. |
|  |  |



*10. Sơ đồ hoạt động thông kê khách hàng.*

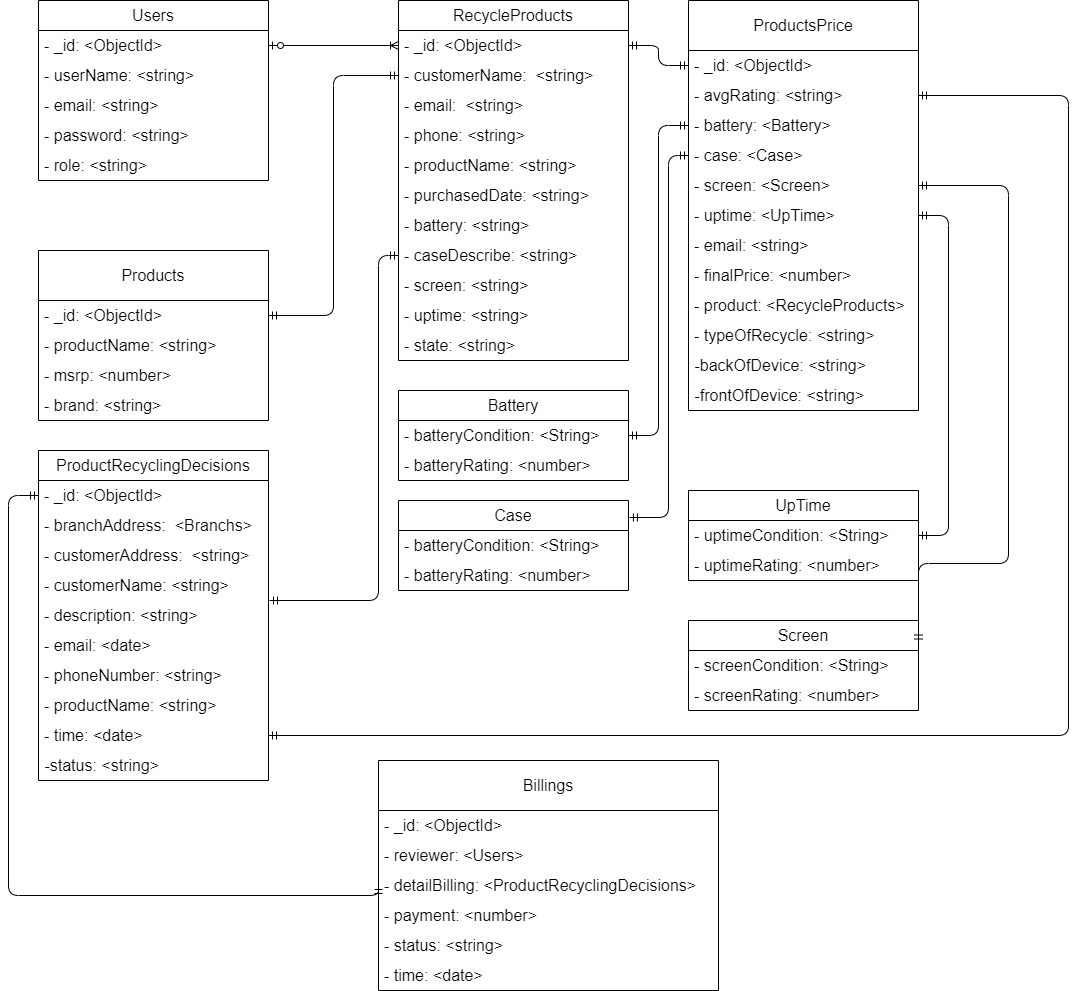
#### UC10\_Thống kê của đánh giá viên

|  |  |
| --- | --- |
| * **Tên use case**: UC10\_Thống kê của đánh giá viên | |
| * **Mô tả sơ lược**: Xem biểu đồ thống kê các sản phẩm được bán lại hoặc thống kê | |
| * **Actor chính**: Đánh giá viên | |
| * **Actor phụ**: Không | |
| * **Tiền điều kiện (Pre-condition):** Có kết nối mạng | |
| * **Hậu điều kiện (Post-condition):** Không | |
| * **Luồng sự kiện chính (main flow):** | |
| **Đánh giá viên** | **Hệ thống** |
| 1. Chọn nút “Analysis” |  |
|  | 1. Kiểm tra thông tin sản phẩm. |
|  | 1. Hiển thị giao diện thống kê hàng tháng: biểu đồ pie sản phẩm bán/tái chế, biểu đồ box plot phân phối sản giá sản phẩm, biểu đồ bar trung các loại đánh giá, biểu đồ bar số lượng bán sản phẩm 3 tháng gần nhất. |
|  |  |
|  |  |
|  |  |
|  |  |
| * **Luồng sự kiện thay thế (alternate flow):** | |
|  | * 1. Nếu không tồn tại: Thông báo không tồn tại sản phẩm để thống kê. |
|  |  |



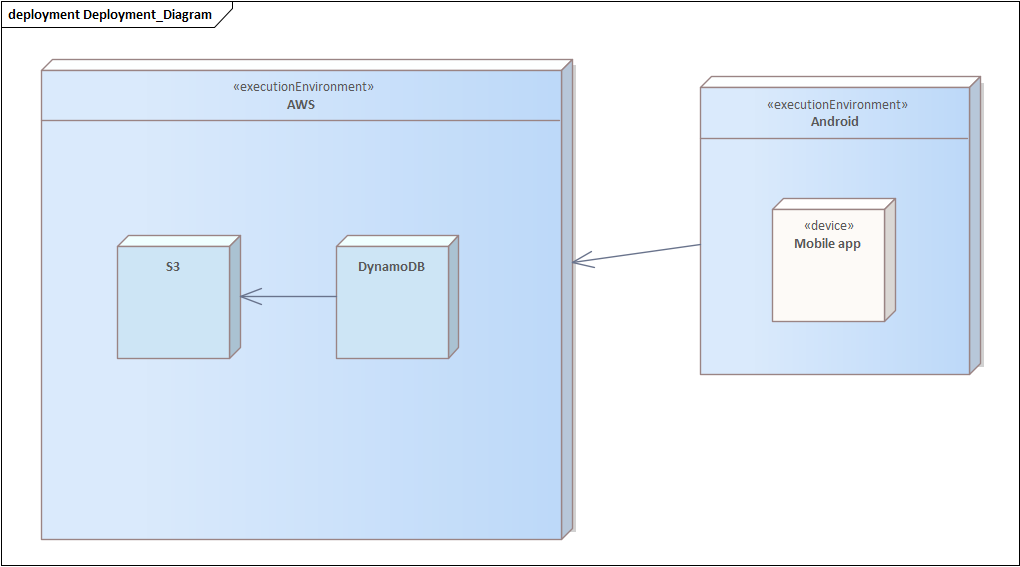
*11. Sơ đồ hoạt động thống kê đánh giá viên.*

## Class diagram



*1. Sơ đồ lớp entity*

## Deployment diagram



*1. Sơ đồ triển khai RecyclePro.*

# : HIỆN THỰC

## Cấu hình phần cứng, phần mềm

Thiết bị sử dụng android 7.0 hoặc cao hơn.

## Giao diện của hệ thống

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, thiết kế

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, Phông chữ

Mô tả được tạo tự động

*12. Giao diện đăng nhập, đăng ký (từ trái sang)*

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, thiết kế

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, thiết kế

Mô tả được tạo tự động

*13. Giao diện khách hàng, đánh giá viên (từ trái sang)*

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Hệ điều hành

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, Trang web, Website

Mô tả được tạo tự động

*14. Giao diện gửi sản phẩm của khách hàng.*

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Trang web

Mô tả được tạo tự động

*15. Giao diện xem lại lịch sử gửi và xác nhận tái chế của khách hàng*

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Hệ điều hành

Mô tả được tạo tự động

*16. Giao diện thống kê của khách hàng*

Ảnh có chứa văn bản, ảnh chụp màn hình, Phông chữ, số

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Phông chữ

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, số, Phông chữ

Mô tả được tạo tự động

Ảnh có chứa văn bản, Điện thoại di động, đồ phụ tùng, ảnh chụp màn hình

Mô tả được tạo tự độngẢnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

*17. Giao diện kiểm tra đánh giá sản phẩm của đánh giá viên.*

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, đa phương tiện, Điện thoại di động

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, đa phương tiện, đồ phụ tùng

Mô tả được tạo tự động

*18. Giao diện lịch sử đánh giá của đánh giá viên.*

Ảnh có chứa văn bản, ảnh chụp màn hình, phần mềm, Hệ điều hành

Mô tả được tạo tự độngẢnh có chứa văn bản, ảnh chụp màn hình, phần mềm, biểu đồ

Mô tả được tạo tự động

*19. Giao diện thống kê của đánh giá viên.*

Ảnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự độngẢnh có chứa văn bản, đồ điện tử, ảnh chụp màn hình, phần mềm

Mô tả được tạo tự động

*20. Giao diện tình trạng tái chế của đánh giá viên.*

## Kế hoạch và hiện thực kiểm thử hệ thống

### Kế hoạch kiểm thử

*4.1 Phần cứng thực hiện kiểm thử*

|  |  |  |  |
| --- | --- | --- | --- |
| **CPU** | **RAM** | **SSD** | **Architecture** |
| Intel Core i5 1135G7, 2.40 GHz - 4.20 GHz | 8 GB | 512GB | Windows 11 23H2 64 bit |

*4.2 Phần mềm thực hiện kiểm thử*

|  |  |  |
| --- | --- | --- |
| **Tên phần mềm** | **Phiên bản** | **Loại** |
| Android Studio | 2023.2.1 | IDE cho Mobile. |
| AWS DynamoDB |  | Cơ sở dữ liệu NoSQL dịch vụ cloud |
| AWS S3 |  | Kho lưu trữ trên nền tảng cloud |
| Microsoft Windows 11 | 11 23H2 | Hệ điều hành |

### Kiểm thử hệ thống

*5.1 Kịch bản kiểm thử.*

| **Test ID** | **Chức năng** | **Mô tả** | **Điều kiện trước** | **Dữ liệu Test** | **Kết quả mong muốn** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | Đăng nhập | Kiểm tra chức năng dăng nhập. | Khởi động ứng dụng, kết nối internet. | Nhập đúng tài khoản và mật khẩu. | Màn hình chính hiện lên. |  |
| 2 | Đăng nhập | Kiểm tra chức năng đăng nhập. | Khởi động ứng dụng, kết nối internet. | Nhập sai tài khoản. | Hệ thống thông báo “Đăng nhập thất bại”. |  |
| 3 | Đăng nhập | Kiểm tra chức năng đăng nhập. | Khởi động ứng dụng, kết nối internet. | Nhập sai mật khẩu. | Hệ thống thông báo “Đăng nhập thất bại”. |  |
| 4 | Đăng nhập | Kiểm tra chức năng đăng nhập. | Khởi động ứng dụng, kết nối internet. | Không nhập tài khoản | Hệ thống thông báo “Đăng nhập thất bại”. |  |
| 5 | Đăng nhập | Kiểm tra chức năng đăng nhập. | Khởi động ứng dụng, kết nối internet. | Mật khẩu <6 kí tự. | Hệ thống thông báo “Đăng nhập thất bại”. | … |
| 6 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Nhập username, email, mật khẩu(6 kí tự), xác nhận mật khẩu đúng định dạng. | Hệ thống thông báo “Đăng ký thành công”. |  |
| 7 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Nhập tài khoản email tồn tại. | Thông báo email đã tồn tại. |  |
| 8 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Không nhập username | Thông báo vui lòng nhập đầy đủ thông tin. |  |
| 9 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Nhập sai định dạng username | Thông báo sai định dạng tên. |  |
| 10 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Không nhập email | Thông báo vui lòng nhập đầy đủ thông tin. |  |
| 11 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Nhập sai định dạng email | Thông báo nhập sai định dạng email. |  |
| 12 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Không nhập password. | Thông báo vui lòng nhập đầy đủ thông tin. |  |
| 13 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Nhập password < 6 kí tự. | Thông báo mật khẩu phải có ít nhất 6 kí tự. |  |
| 14 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Không nhập xác nhận password. | Thông báo vui lòng nhập đầy đủ thông tin. |  |
| 15 | Đăng ký | Kiểm tra chức năng đăng ký. | Khởi động ứng dụng, kết nối internet, chưa có tài khoản. | Nhập xác nhận passord không trùn g password. | Hệ thống thông báo: ”Nhập sai mật khẩu”. |  |
| 16 | Gửi thông tin sản phẩm. | Kiểm tra thông tin nhập sau khi gửi thông tin sản phẩm. | Khởi động ứng dụng, kết nối internet. | Điền đầy đủ thông tin và nhập đúng định dạng số điện thoại | Thông báo “Submit successful” |  |
| 17 | Gửi thông tin sản phẩm. | Kiểm tra thông tin nhập sau khi gửi thông tin sản phẩm. | Khởi động ứng dụng, kết nối internet. | Điền thiếu thông tin | Thông báo “You have not entered enough data” |  |
| 18 | Gửi thông tin sản phẩm. | Kiểm tra thông tin nhập sau khi gửi thông tin sản phẩm. | Khởi động ứng dụng, kết nối internet. | Nhập sai định dạng số điện thoại. | Thông báo "Phone number is wrong format" |  |
| 19 | Chi tiết thông tin tái chế. | Kiểm tra thông tin nhập trước khi gửi thông tin xác nhận tái chế sản phẩm. | Khởi động ứng dụng, kết nối internet, sản phẩm đã được đánh giá. | Điền đầy đủ thông tin và nhập đúng định dạng số điện thoại | Thông báo Your information has been submited |  |
| 20 | Chi tiết thông tin tái chế. | Kiểm tra thông tin nhập trước khi gửi thông tin xác nhận tái chế sản phẩm. | Khởi động ứng dụng, kết nối internet, sản phẩm đã được đánh giá. | Điền thiếu thông tin | Thông báo You have not entered enough information |  |
| 21 | Chi tiết thông tin tái chế. | Kiểm tra thông tin nhập trước khi gửi thông tin xác nhận tái chế sản phẩm. | Khởi động ứng dụng, kết nối internet, sản phẩm đã được đánh giá. | Nhập sai định dạng số điện thoại. | Thông báo "Phone number is wrong format" |  |
| 22 | Đánh giá sản phẩm tái chế | Kiểm tra thông tin đã hợp lệ sau khi đánh giá sản phẩm của khách hàng, sau đó nhấn nút gửi mail về cho khách hàng. | Khởi động ứng dụng, kết nối internet, đăng nhập với vai trò đánh giá viên. | Nhập đầy đủ thông tin về giá, đánh giá các chỉ tiêu, tính toán giá cuối cùng. | Hệ thống thông báo đã gửi thông tin cho khách hàng. |  |
| 24 | Đánh giá sản phẩm tái chế | Kiểm tra thông tin đã hợp lệ sau khi đánh giá sản phẩm của khách hàng, sau đó nhấn nút gửi mail về cho khách hàng. | Khởi động ứng dụng, kết nối internet, đăng nhập với vai trò đánh giá viên. | Nhập thiếu dữ liệu price. | Thông báo bạn chưa nhập giá |  |
| 25 | Đánh giá sản phẩm tái chế | Kiểm tra thông tin đã hợp lệ sau khi đánh giá sản phẩm của khách hàng, sau đó nhấn nút gửi mail về cho khách hàng. | Khởi động ứng dụng, kết nối internet, đăng nhập với vai trò đánh giá viên. | Nhập sai định dạng price( không phải là số) | Thông báo giá không đúng định dạng |  |
| 26 | Đánh giá sản phẩm tái chế | Kiểm tra thông tin đã hợp lệ sau khi đánh giá sản phẩm của khách hàng, sau đó nhấn nút gửi mail về cho khách hàng. | Khởi động ứng dụng, kết nối internet, đăng nhập với vai trò đánh giá viên. | Chưa đánh giá các chỉ tiêu | Thông báo bạn chưa đánh giá đầy đủ. |  |
| 27 | Đánh giá sản phẩm tái chế | Kiểm tra thông tin đã hợp lệ sau khi đánh giá sản phẩm của khách hàng, sau đó nhấn nút gửi mail về cho khách hàng. | Khởi động ứng dụng, kết nối internet, đăng nhập với vai trò đánh giá viên. | Chưa tính giá cuối cùng | Thông báo bạn chưa tính giá cuối cùng |  |

*6. Bảng kết quả kiểm thử.*

| **Test ID** | **Ngày testing** | **Người tham gia Test** | **Pass/Fail** | **Độ nghiêm trọng** | **Tóm tắt lỗi** | **Ghi chú** |
| --- | --- | --- | --- | --- | --- | --- |
| 1 | 19/05/2024 | Huỳnh Văn Sang | 44/1 | Thấp | Lỗi sai định dạng giá |  |
|  |  |  |  |  |  |  |

# : KẾT LUẬN

## Kết quả đạt được

**Chức năng cho Khách hàng**

**Đăng ký và Đăng nhập:**

Người dùng cần có khả năng đăng ký tài khoản mới và đăng nhập vào hệ thống bằng thông tin xác thực của mình.

Gửi thông tin sản phẩm cho đánh giá viên:

Giao diện để khách hàng nhập thông tin chi tiết về sản phẩm mà họ muốn đánh giá cho mục đích tái chế.

Khả năng tải lên hình ảnh hoặc tài liệu liên quan đến sản phẩm.

**Xem lại đánh giá của đánh giá viên:**

Khách hàng có thể xem các đánh giá chi tiết từ đánh giá viên về sản phẩm đã gửi.

Giao diện hiển thị thông tin đánh giá, nhận xét và kết quả đánh giá (chấp nhận hoặc từ chối tái chế).

**Quyết định chấp nhận hoặc từ chối tái chế:**

Khách hàng có thể chấp nhận hoặc từ chối kết quả đánh giá từ đánh giá viên.

Hệ thống lưu trữ quyết định của khách hàng và thông báo cho đánh giá viên.

**Xem thống kê:**

Khách hàng có thể xem thống kê về số lượng sản phẩm đã gửi, số sản phẩm được đánh giá hoặc chưa, và các dữ liệu liên quan khác.

**Chức năng cho Đánh giá viên**

**Đăng nhập:**

Đánh giá viên có khả năng đăng nhập vào hệ thống bằng thông tin xác thực của mình.

Xem thông tin sản phẩm và tiến hành đánh giá:

Giao diện để đánh giá viên xem thông tin chi tiết về sản phẩm do khách hàng gửi.

Khả năng nhập các đánh giá chi tiết và đưa ra kết quả (chấp nhận hoặc từ chối tái chế).

**Xem lại lịch sử đánh giá:**

Đánh giá viên có thể truy cập lịch sử các đánh giá mà họ đã thực hiện.

Giao diện hiển thị thông tin về các sản phẩm đã đánh giá và kết quả của từng sản phẩm.

**Xem danh sách các sản phẩm yêu cầu từ chối hoặc chấp nhận tái chế của khách hàng:**

Đánh giá viên có thể xem danh sách các sản phẩm mà khách hàng đã chấp nhận hoặc từ chối kết quả tái chế.

Cập nhật và theo dõi trạng thái của các sản phẩm trong quá trình xử lý.

**Xem thống kê:**

Đánh giá viên có thể xem thống kê số lượng đánh giá trong tháng, phân phối giá chót và các dữ liệu liên quan khác.

## Hạn chế của đồ án

**Không thể chat realtime với khách hàng:**

Việc không có tính năng chat realtime làm giảm khả năng giao tiếp nhanh chóng giữa khách hàng và đánh giá viên, ảnh hưởng đến trải nghiệm người dùng và hiệu quả xử lý vấn đề.

**Load đôi lúc chậm do cloud phản hồi chậm:**

Tốc độ phản hồi chậm từ server hoặc cloud làm giảm hiệu suất hệ thống, gây khó chịu cho người dùng.

**Thống kê chưa đa dạng:**

Hệ thống hiện tại chưa cung cấp đủ đa dạng thông tin thống kê về các hoạt động và dữ liệu liên quan đến quá trình đánh giá sản phẩm tái chế.

Thống kê chỉ tập trung vào một số chỉ số cơ bản mà không thể phản ánh đầy đủ hoặc chi tiết các khía cạnh của quy trình.

Sự thiếu hụt thông tin thống kê đa dạng có thể làm hạn chế khả năng đánh giá và tối ưu hóa hiệu suất của hệ thống.

**Hạn chế trong việc lọc dữ liệu:**

Cơ chế lọc dữ liệu trong hệ thống hiện tại vẫn còn hạn chế và không cung cấp đủ các tùy chọn để người dùng có thể tìm kiếm và lọc dữ liệu một cách hiệu quả.

Việc lọc dữ liệu hạn chế có thể làm giảm tính linh hoạt và khả năng điều chỉnh của người dùng khi họ cần truy xuất thông tin cụ thể hoặc thực hiện các phân tích chi tiết.

**Một số chuyển trang bị chậm:**

Chuyển trang chậm có thể làm giảm trải nghiệm người dùng, gây mất kiên nhẫn và giảm hiệu quả làm việc.

## Hướng phát triển

**Tích hợp Trí Tuệ Nhân Tạo (AI) và Học Máy (Machine Learning):**

Sử dụng AI và Machine Learning để tự động phân loại sản phẩm và dự đoán khả năng tái chế.

Phát triển các mô hình dự đoán để giúp đánh giá viên và khách hàng đưa ra quyết định dựa trên dữ liệu lịch sử và các yếu tố khác.

**Mở rộng Tính Năng Giao Tiếp Thời Gian Thực:**

Tích hợp tính năng chat realtime mạnh mẽ hơn, bao gồm cả trò chuyện video, để tăng cường giao tiếp giữa khách hàng và đánh giá viên.

Phát triển chatbot thông minh hơn có khả năng giải quyết các vấn đề phức tạp và cung cấp hỗ trợ tự động.

**Nâng Cấp Hệ Thống Quản Lý Dữ Liệu:**

Tối ưu hóa cơ sở dữ liệu để xử lý lượng dữ liệu lớn một cách hiệu quả hơn.

Triển khai các giải pháp lưu trữ dữ liệu phân tán để giảm độ trễ và tăng tính sẵn sàng của hệ thống.

**Phát Triển Ứng Dụng Di Động:**

Xây dựng ứng dụng di động cho cả khách hàng và đánh giá viên, cung cấp trải nghiệm tiện lợi và linh hoạt hơn.

Tối ưu hóa ứng dụng để hoạt động mượt mà trên các thiết bị di động và kết nối mạng khác nhau.

**Mở Rộng Phạm Vi và Hiệu Quả Tái Chế:**

Mở rộng hệ thống để hỗ trợ đánh giá và tái chế cho nhiều loại sản phẩm và vật liệu khác nhau.

Tạo ra các đối tác và liên kết với các doanh nghiệp tái chế để tối ưu hóa quy trình tái chế và tăng cường tầm ảnh hưởng xã hội và môi trường.

**Cải Thiện Bảo Mật và Quản Lý Quyền Truy Cập:**

Tăng cường bảo mật để đảm bảo an toàn và bảo vệ dữ liệu người dùng.

Phát triển các cơ chế quản lý quyền truy cập để kiểm soát việc truy cập vào thông tin sản phẩm và đánh giá.

**Nghiên Cứu và Áp Dụng Công Nghệ Mới:**

Theo dõi và áp dụng các công nghệ mới như Blockchain để cải thiện tính minh bạch và đảm bảo tính xác thực của dữ liệu.

Sử dụng Internet of Things (IoT) để thu thập dữ liệu từ các thiết bị thông minh và cảm biến để hỗ trợ quá trình tái chế.

# TÀI LIỆU THAM KHẢO

Các tài liệu từ Internet

1. [Sparx Systems - Sparx Enterprise Architect User Guide.Sparx Systems Documentation – Truy cập lần cuối 5/2024.](https://sparxsystems.com/resources/tutorials/)
2. [Visual Paradigm - Visual Paradigm Documentation. Visual Paradigm Documentation – Truy cập lần cuối 5/2024](https://circle.visual-paradigm.com/diagram-examples/)**.**
3. [Amazon Web Services - DynamoDB Documentation. AWS DynamoDB Documentation – Truy cập lần cuối 5/2024.](https://docs.aws.amazon.com/dynamodb/)
4. [Amazon Web Services - S3 Documentation. AWS S3 Documentation – Truy cập lần cuối 5/2024.](https://docs.aws.amazon.com/s3/)