A logo for a university

Description automatically generatedTRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM

**KHOA CÔNG NGHỆ THÔNG TIN**

**TÊN ĐỀ TÀI**

**Xây dựng hệ thống quản lí tái chế rác thải với kiến trúc Phân lớp (Layered Architecture)**

Sinh viên thực hiện : Vũ Huỳnh Minh Nhật - 20070151

Nguyễn Đình Duy -20052471

Lớp : DHKTPM16ETT

Tp. Hồ Chí Minh, tháng 5 năm 2024

**1.Giới thiệu**

Vấn đề về quản lý rác thải trở thành một thách thức ngày càng lớn đối với xã hội. Chúng ta không thể phủ nhận rằng lượng rác thải đang tích tụ và gây ảnh hưởng tiêu cực đến môi trường và sức khỏe con người. Trong bối cảnh này, khái niệm tái chế đã nổi lên như một giải pháp hữu hiệu để giảm thiểu tác động của rác thải đối với môi trường. Tái chế là quá trình chuyển đổi các vật liệu đã qua sử dụng thành sản phẩm mới hoặc tái sử dụng chúng để giảm thiểu sự lãng phí tài nguyên và ô nhiễm môi trường. Thay vì loại bỏ hoặc đốt cháy các vật liệu đã qua sử dụng, tái chế cho phép chúng được tái sử dụng để tạo ra sản phẩm mới hoặc thực hiện các quá trình sản xuất khác. Vì vậy nhóm quyết định chọn chủ đề này để làm về hệ thống quản lí tái chế rác thải.

**2.Mục tiêu**

Mục tiêu của hệ thống quản lí tái chế rác thải là trở thành một công cụ quan trọng trong việc quản lí và tối ưu hóa quá trình tái chế rác thải. Nó giúp tổ chức và theo dõi các hoạt động tái chế, từ việc thu thập thông tin về loại rác thải, phân loại, xử lí, đến việc theo dõi quy trình tái chế và đánh giá hiệu quả. Hệ thống quản lí tái chế rác thải giúp tăng cường hiệu suất và giảm thiểu lãng phí trong quá trình tái chế, đồng thời đảm bảo tuân thủ các quy định và tiêu chuẩn về môi trường.

**3.Yêu cầu chức năng**

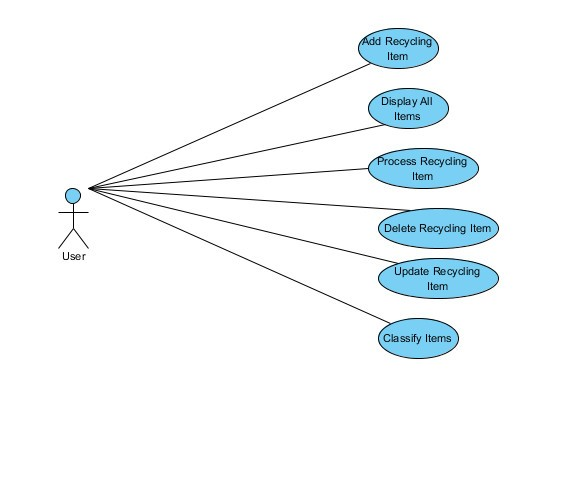
Hệ thống phần mềm quản lý tái chế rác thải có thể có một số chức năng sau:

Quản lý thông tin rác thải: Hệ thống có khả năng thu thập, lưu trữ và quản lý thông tin về loại rác thải, khối lượng, nguồn gốc, và các thông tin khác liên quan.

Phân loại và xử lý rác thải: Hệ thống giúp phân loại rác thải theo loại, như nhựa, giấy, kim loại, thủy tinh, hữu cơ, vv. Nó cũng cung cấp các phương pháp xử lý rác thải như tái chế, tái sử dụng, chế biến nhiệt, và xử lý bằng vi sinh vật.

Đánh giá hiệu quả: Hệ thống cung cấp các công cụ để đánh giá hiệu quả tái chế, bao gồm việc theo dõi lượng rác thải được tái chế, tiết kiệm tài nguyên, giảm lượng khí thải và ô nhiễm môi trường.

Báo cáo và thống kê: Hệ thống giúp tạo ra báo cáo và thống kê về hoạt động tái chế, bao gồm thông tin về loại rác thải, khối lượng, tiến độ tái chế, hiệu quả và các chỉ số khác.



Sơ đồ usecase tổng quát

**4.Kiến trúc hệ thống**

Hệ thống quản lí tái chế rác thải sử dụng kiến trúc phân tầng (Layered Architecture).Các thành phần bên trong kiến trúc phân lớp được tổ chức thành các tầng nằm ngang, mỗi tầng thực hiện một vai trò cụ thể trong ứng dụng

– Tầng Presentation: được dùng để giao tiếp với người dùng, nhiệm vụ chính là hiển thị dữ liệu và nhận dữ liệu từ người dùng.

– Tầng Business Logic: nhiệm vụ chính là cung cấp các chức năng của phần mềm.

– Tầng Persistence: cầu nối với lớp database, có nhiệm vụ xử lí , truy xuất dữ liệu từ các tầng trên xuống database và trả kết quả lên giao diện

– Tầng Data: lưu trữ dữ liệu, cho phép lớp Business Logic có thể tìm kiếm, trích xuất, cập nhật… dữ liệu.

A diagram of a business

Description automatically generated

Mô hình kiến trúc phân lớp (Layered Architecture)

* Ưu điểm của kiến trúc phân lớp

1. Tính tổ chức và dễ hiểu:

- Kiến trúc phân lớp giúp tổ chức mã nguồn một cách rõ ràng và dễ hiểu. Mỗi tầng có một nhiệm vụ cụ thể, giúp các nhà phát triển dễ dàng tìm kiếm và quản lý mã nguồn.

2. Tính tái sử dụng:

- Các tầng có thể được tái sử dụng trong các dự án khác. Ví dụ, tầng dữ liệu có thể được sử dụng lại trong nhiều ứng dụng khác nhau mà không cần thay đổi.

3. Dễ bảo trì và mở rộng:

- Do các tầng được phân chia rõ ràng, việc thay đổi hoặc cập nhật một tầng sẽ ít ảnh hưởng đến các tầng khác. Điều này làm cho việc bảo trì và mở rộng hệ thống trở nên dễ dàng hơn.

4. Dễ kiểm tra và gỡ lỗi:

- Việc kiểm tra và gỡ lỗi trở nên dễ dàng hơn do có sự tách biệt giữa các tầng. Các nhà phát triển có thể kiểm tra và gỡ lỗi từng tầng một cách độc lập.

5. Tăng tính bảo mật:

- Bằng cách cô lập các tầng khác nhau, kiến trúc phân lớp có thể giúp tăng cường bảo mật. Ví dụ, tầng dữ liệu có thể được bảo vệ khỏi truy cập trực tiếp từ người dùng bằng cách sử dụng các tầng trung gian.

* Nhược điểm của kiến trúc phân lớp

1. Hiệu suất:

- Do phải đi qua nhiều tầng, kiến trúc phân lớp có thể làm giảm hiệu suất của hệ thống. Mỗi tầng có thể thêm độ trễ khi xử lý các yêu cầu.

2. Độ phức tạp:

- Việc thiết kế và duy trì các tầng có thể trở nên phức tạp, đặc biệt là khi hệ thống lớn và có nhiều tầng. Điều này đòi hỏi phải có kỹ năng và kinh nghiệm tốt trong việc thiết kế kiến trúc phần mềm.

3. Tính cứng nhắc:

- Kiến trúc phân lớp có thể trở nên cứng nhắc và khó thay đổi khi cần phải thay đổi cấu trúc của hệ thống. Ví dụ, việc thay đổi logic nghiệp vụ có thể đòi hỏi thay đổi ở nhiều tầng khác nhau.

4. Khả năng mở rộng hạn chế:

- Đối với các hệ thống lớn, kiến trúc phân lớp có thể gặp khó khăn trong việc mở rộng. Việc thêm chức năng mới hoặc mở rộng hệ thống có thể đòi hỏi phải điều chỉnh nhiều tầng, gây khó khăn trong việc duy trì và phát triển hệ thống.

5. Tăng khối lượng mã nguồn:

- Việc phân chia hệ thống thành nhiều tầng có thể làm tăng khối lượng mã nguồn, do phải thêm các lớp trung gian và các giao diện để tương tác giữa các tầng.

**5.Kiểu hệ thống phù hợp với kiến trúc phân lớp**

Thông thường, các hệ thống lớn và phức tạp thường được thiết kế theo kiến trúc phân lớp để tăng tính tổ chức và dễ bảo trì. Các hệ thống có yêu cầu mở rộng và thay đổi linh hoạt cũng thích hợp với kiến trúc phân lớp. Ngoài ra, các hệ thống có yêu cầu về tính tái sử dụng cao cũng có thể tận dụng được lợi ích của kiến trúc phân lớp.

Kiến trúc phân lớp cũng giúp tách biệt logic kinh doanh và giao diện người dùng, giúp dễ dàng thay đổi giao diện mà không ảnh hưởng đến logic bên trong. Tuy nhiên, việc sử dụng kiến trúc phân lớp cũng có nhược điểm, như tăng độ phức tạp và chi phí phát triển ban đầu.

**6.Các vấn đề trình bày**

- Việc chọn thiết kế kiến trúc phân lớp là một quyết định quan trọng trong quá trình phát triển phần mềm và có thể đối mặt với một số vấn đề sau:

* Hiệu suất: Kiến trúc phân lớp có thể gây ra độ trễ trong việc truy cập và xử lý dữ liệu do việc đi qua nhiều tầng. Điều này có thể ảnh hưởng đến hiệu suất của ứng dụng, đặc biệt là đối với các ứng dụng có yêu cầu về tốc độ và độ trễ thấp.
* Khả năng mở rộng của phần mềm: trong một vài trường hợp có thể khiến cho việc mở rộng trở nên khó khăn do có xuất hiện yêu cầu mới cần phải thay đổi nhiều phần của hệ thống.
* Tính linh hoạt: Kiến trúc phân lớp có thể không linh hoạt đủ để đáp ứng nhanh chóng các yêu cầu thay đổi của khách hàng hoặc thị trường. Các thay đổi có thể yêu cầu phải thay đổi ở nhiều tầng, gây ra sự phức tạp và chi phí cao.

**7.Kết luận**

Hệ thống quản lí tái chế rác thải của nhóm đã đạt được mục đính chính là giúp người dùng quản lí được lượng rác thải thu thập được, phân loại và thống kê số lượng rác thải. Bên cạnh đó vẫn còn một vài hạn chế trong việc quản lí quy trình quản lí xử lí tái chế, đánh giá mức độ hiệu quả của công việc

Sử dụng kiến trúc phân tầng trong hệ thống quản lý tái chế rác thải théo nhóm là một lựa chọn hợp lý, mang lại nhiều lợi ích về tổ chức, bảo trì, bảo mật và tính tái sử dụng. Tuy nhiên, cần cân nhắc kỹ lưỡng về hiệu suất (việc đi qua nhiều tầng có thể gây ra độ trễ, ảnh hưởng đến hiệu suất của hệ thống.) và độ phức tạp(đòi hỏi kỹ năng cao và sự hiểu biết sâu về kiến trúc phần mềm) khi thiết kế và triển khai hệ thống.