

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC BÁCH KHOA**  
**KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH**



**BÁO CÁO BÀI TẬP LỚN**  
**CÔNG NGHỆ PHẦN MỀM**

---

**Đề tài**  
**Urban Waste Collection Aid - UWC 2.0**

---

**GVHD: Lê Đình Thuận**

**Sinh viên thực hiện:**

- |                        |               |
|------------------------|---------------|
| 1. Nhiếp Thị Bảo Châu  | MSSV: 2012715 |
| 2. Nguyễn Trí Hiếu     | MSSV: 2013153 |
| 3. Huỳnh Ngọc Như      | MSSV: 2010495 |
| 4. Võ Phan Anh Quân    | MSSV: 2014285 |
| 5. Trần Công Minh Quân | MSSV: 2012528 |
| 6. Lê Văn Vỹ           | MSSV: 2010805 |

**TP. Hồ Chí Minh, 11/2022**

# Mục lục

<b>1</b>	<b>Giới thiệu dự án</b>	<b>2</b>
1.1	Bối cảnh của dự án . . . . .	2
1.2	Các bên liên đới đến dự án (Stakeholder) . . . . .	2
<b>2</b>	<b>Phân tích yêu cầu</b>	<b>5</b>
2.1	Yêu cầu chức năng(Functional requirements) . . . . .	5
2.2	Yêu cầu phi chức năng(Non-functional requirements) . . . . .	7
<b>3</b>	<b>Biểu đồ Use-case</b>	<b>10</b>
3.1	Biểu đồ use-case cho toàn hệ thống . . . . .	10
3.2	Biểu đồ use-case cho Task Assignment module . . . . .	10
<b>4</b>	<b>Mô hình hóa hệ thống</b>	<b>14</b>
4.1	Activity Diagram cho Task Assignment module . . . . .	14
4.2	Sequence Diagram và giải pháp khái quát cho Route Planning task . . . . .	16
4.2.1	Assumption . . . . .	16
4.2.2	Workflow . . . . .	17
4.3	Class Diagram cho Task Assignment module . . . . .	18
4.3.1	Route and vehicle planning . . . . .	19
4.3.2	Collector and Janitor Assignment . . . . .	20
4.3.3	Business và UI class . . . . .	21
<b>5</b>	<b>Thiết kế kiến trúc</b>	<b>22</b>
5.1	Mô hình kiến trúc MVC . . . . .	22
5.1.1	Mô tả về kiến trúc MVC . . . . .	22
5.1.2	Lý do lựa chọn MVC . . . . .	23
5.1.3	Nhược điểm của MVC . . . . .	24
5.2	Modules . . . . .	24
5.3	Component Diagram . . . . .	24

# 1 Giới thiệu dự án

## 1.1 Bối cảnh của dự án

Việc quản lý rác thải đô thị là trong những vấn đề mà nhiều thành phố (đặc biệt là các thành phố thuộc các nước đang phát triển) đang gặp phải. Đặc biệt, rác thải chất rắn tốn nhiều chi phí để quản lý và thường phát sinh nhiều vấn đề nan giải. Một trong những giải pháp khả thi cho vấn đề này là phát triển một ứng dụng nhằm quản lý quá trình thu gom rác thải chất rắn ở đô thị nhờ vậy sẽ cải thiện hiệu suất, giảm sức ép lên nguồn nhân lực và cơ sở hạ tầng.

Từ ý tưởng đó, ứng dụng UWC 2.0 đã được phát triển bởi tổ chức X nhằm phục vụ cho dịch vụ thu gom rác của Nhà cung cấp dịch vụ Y. Trước ứng dụng UWC 2.0 đã tồn tại một hệ thống UWC 1.0. Ứng dụng UWC 2.0 được đòi hỏi sẽ tận dụng những dữ liệu đã có ở hệ thống UWC 1.0 đồng thời tính năng “Task Management” trên UWC 2.0 phải tương thích và sử dụng được với UWC 1.0 nhiều nhất có thể.

## 1.2 Các bên liên đới đến dự án (Stakeholder)

- a. *Collectors*: Lái xe chở rác, thu gom rác tại điểm tập kết (MCP).
- b. *Janitors*: Thu gom rác, vận chuyển vào MCP.
- c. *Back officers*: Quản lý, phân chia tuyến đường, khu vực phụ trách, công việc cho collector và janitor.
- d. *Đội ngũ kỹ thuật viên* Cài đặt và bảo trì ứng dụng.
- e. *Đơn vị quản lý MCP (quản lý sensor cập nhật thông số của MCP)*: Đảm bảo MCP hoạt động bình thường, đảm bảo luồng thông tin về MCP luôn được cung cấp tới ứng dụng.
- f. *Bên Y - công ty cung cấp dịch vụ quản lý rác thải*: Sử dụng ứng dụng UWC 2.0 để quản lý quy trình thu gom rác hiệu quả hơn.
- g. *Cơ quan quản lý rác thải đô thị/ UBND thành phố*: Cấp phép cho ứng dụng. Có thể yêu cầu các thông tin quản lý chung lấy từ ứng dụng.

Stakeholder	Nhu cầu/Vấn đề đặt ra	Giải pháp từ UWC 2.0
Collectors Janitors	<p>Khó nắm được thông tin phân công công việc của mình. Không thể làm việc một mình mà cần trao đổi thông tin với các Collectors, Janitors, Back officers đặc biệt là khi thu gom rác tại điểm tập kết MCP.</p> <p>Nếu không nắm được thông tin cập nhật về các MCP kịp thời, sẽ không biết phân phối, đổi hướng lượng rác để không bị quá tải rác hoặc tốn công sức, thời gian cho các địa điểm trống.</p> <p>Báo cáo tiến độ hoàn thành công việc lên hệ thống.</p>	<p>Theo dõi được thông tin phân công công việc hằng ngày, hằng tuần của mình bao gồm: tuyến đường, xe (dành cho collectors), các điểm MCP.</p> <p>Có thể trao đổi thông tin với các Collectors, Janitors, Back officers khác.</p> <p>Nắm được thông tin về các MCP để triển khai công việc.</p> <p>Đánh dấu được các công việc đã hoàn thành.</p>
Back officers	<p>Cần giải pháp để quản lý, lên lịch, phân công collectors và janitors.</p> <p>Biết được các thống kê về công việc, thống kê về MCP và tiến độ hoàn thành công việc của các nhân viên cấp dưới.</p>	<p>Ngoài các chức năng như 2 bên collectors, janitors, còn các chức năng khác đặc thù:</p> <ul style="list-style-type: none"> <li>- Nắm được thông tin về các tuyến đường, MCP, mức độ hoàn thành công việc của nhân viên.</li> <li>- Phân công công việc cho các bên dựa theo tình hình MCP, các tuyến đường và thông báo tự động đến các bên tham gia đó.</li> </ul>
Đội ngũ kỹ thuật viên	<p>Cần 1 ứng dụng dễ sử dụng, tương thích với nhiều nền tảng (trình duyệt web) khi cài đặt hàng loạt cho nhân viên</p>	<p>Ứng dụng website dễ dùng, tương thích nhiều nền tảng.</p> <p>Tự động hóa quy trình gửi thông báo về MCP, thông báo về công việc được phân công.</p>
Đơn vị quản lý MCP (quản lý các sensors cập nhật thông số về các MCP)	<p>Các MCP luôn có bên Collectors hoặc Janitors phụ trách phù hợp, không để xảy ra trường hợp quá tải hoặc trống.</p>	<p>Thông tin từ MCP sẽ được cập nhật real time đến cho quản lý Back officers, janitors để tối ưu hóa tình trạng rác tại các MCP.</p> <p>Các Collectors, Janitor được trao đổi với nhau để nắm bắt thông tin khi lấy rác tại các MCP.</p>

Bên Y - công ty cung cấp dịch vụ quản lý rác thải	Cần quản lý việc phân công, quá trình trao đổi thu gom rác giữa các nhân viên hiệu quả từ đó dịch vụ quản lý rác thải được cải thiện, tiết kiệm chi phí.	Các back officers phân công công việc cho các nhân viên sao cho tối ưu nhất.  Các nhân viên cũng có thể liên hệ với nhau, thông tin từ MCP cập nhật liên tục để tránh trường hợp quá tải, trống, không có người phụ trách.
Cơ quan quản lý rác thải đô thị/UBND thành phố	Cần thông tin lịch sử về phân công công việc của các Janitors, Collectors, Back officers.  Mong muốn cải thiện vấn đề rác thải của thành phố.	Ứng dụng có tính năng xuất thông tin ra dạng file log để xem được thông tin phân công, quản lý việc thu gom rác thải.  Hiện thị thông tin hiện tại về, để cập nhật đc tình hình công việc của các Janitors, Collectors, Back officers.  Xem xét độ hiệu quả của ứng dụng trong việc tối ưu hóa quy trình thu gom rác.

## 2 Phân tích yêu cầu

### 2.1 Yêu cầu chức năng(Functional requirements)

*a. Đối với back officer:*

- Có thể quản lý tài khoản cá nhân trên hệ thống (đăng ký, đăng nhập, thay đổi thông tin cá nhân, thay đổi mật khẩu)
- Có thể giao tiếp với Janitor và Collector (thông qua nhắn tin)
- Có thể xem được danh sách thông tin của tất cả các Janitor và Collector: khung giờ làm việc, lịch trình làm việc (thời gian nào - ở địa điểm nào), tuyến đường làm việc, phương tiện được cấp (cho Collector)
- Có thể xem được danh sách thông số của tất cả các phương tiện thu gom rác: khối lượng, dung tích bể chứa, mức nhiên liệu, tuyến đường di chuyển
- Có thể xem được danh sách thông số của các điểm tập kết rác (MCP): vị trí trên bản đồ, dung tích tối đa của điểm tập kết rác, dung tích hiện thời của điểm tập kết rác
- Có thể xem được tiến độ công việc của Janitor
- Có thể gửi mail thông báo thay đổi lịch trình cho Collector và Janitor
- Có thể chỉnh sửa task (khung giờ, địa điểm, phương tiện thu gom rác) cho Janitor và Collector
- Có thể gửi lịch trình làm việc hằng ngày, hằng tuần cho Janitor và Collector
- Có thể thay đổi tuyến đường thu gom rác cho Janitor và Collector nếu xuất hiện MCP đầy
- Có thể nhận được phương án tối ưu hóa tuyến đường từ hệ thống hỗ trợ bên ngoài (external system)
- Có thể thông báo thông tin về dung tích của điểm tập kết rác (MCP) cho những Janitor và Collector phụ trách
- Có thể thông báo về điểm tập kết rác (MCP) đầy

*b. Đối với Janitor:*

- Có thể quản lý tài khoản cá nhân trên hệ thống (đăng ký, đăng nhập, thay đổi thông tin cá nhân, thay đổi mật khẩu)
- Có thể xem lịch trình làm việc hằng ngày: khung giờ làm việc, lịch trình làm việc, tuyến đường làm việc, phương tiện làm việc
- Có thể xem lịch trình làm việc hằng tuần: tuyến đường, MCP, phương tiện thu gom rác

- Có thể xem thông tin Collector đồng nghiệp được phân công cùng phương tiện thu gom rác
- Có thể xem được danh sách thông số của các điểm tập kết rác (MCP): vị trí trên bản đồ, dung tích tối đa của điểm tập kết rác, dung tích hiện thời của điểm tập kết rác
- Có thể nhận được thông báo nếu điểm tập kết rác (MCP) đầy
- Có thể giao tiếp với Collector, các Janitor khác và Back officer thông qua tin nhắn
- Có thể cập nhật tiến độ công việc lên hệ thống cho Back officer
- Có thể nhận được thông báo thay đổi lịch trình từ Back officer

*c. Đối với Collector:*

- Có thể quản lý tài khoản cá nhân trên hệ thống (đăng ký, đăng nhập, thay đổi thông tin cá nhân, thay đổi mật khẩu)
- Có thể xem lịch trình làm việc hàng ngày: khung giờ làm việc, lịch trình làm việc, tuyến đường làm việc, phương tiện làm việc
- Có thể xem thông tin Janitor đồng nghiệp được phân công cùng phương tiện thu gom rác
- Có thể xem được thông số của phương tiện thu gom rác được điều khiển
- Có thể xem được danh sách thông số của các điểm tập kết rác (MCP): vị trí trên bản đồ, dung tích tối đa của điểm tập kết rác, dung tích hiện thời của điểm tập kết rác
- Có thể gửi thông báo về mức nhiên liệu cho Back Officer
- Có thể nhận thông báo về trình trạng đầy của các điểm tập kết rác (MCP) từ Back officer
- Có thể giao tiếp với Janitor, các Collector khác và Back officer thông qua tin nhắn
- Có thể nhận thông báo thay đổi lịch trình từ Back officer

*d. Đối với hệ thống hỗ trợ bên ngoài (external system):*

- Có thể nhận thông tin từ hệ thống UWC 2.0: tuyến đường, MCP
- Có thể gửi thông tin về phương án tối ưu tuyến đường - MCP cho Back officer

*e. Đối với sensor:*

- Có thể nhận thông tin từ hệ thống UWC 2.0: tuyến đường, MCP, phương tiện thu gom rác
- Có thể gửi thông tin về dung tích của MCP cho Back officer

## 2.2 Yêu cầu phi chức năng(Non-functional requirements)

Ngoài những yêu cầu phi chức năng đã được nêu ở trong đề, nhóm còn bổ sung thêm một số yêu cầu phi chức năng khác, và sắp xếp thành các nhóm yêu cầu sau:

### *a. Security*

- Password của người dùng phải được hash bằng MD5.
- Hệ thống sẽ tạm deactivate tài khoản của người dùng nếu nhập sai password 5 lần liên tiếp.
- Những dữ liệu nhạy cảm của người dùng, như thông tin nhân viên, tài khoản, mật khẩu phải được mã hoá bằng 1024 bit SSL.
- Khi user quên mật khẩu và thực hiện chức năng lấy lại mật khẩu, mật khẩu mặc định của hệ thống phải được gửi về email được dùng để đăng ký tài khoản, hoặc gửi mã OTP về cho số điện thoại được sử dụng để đăng ký tài khoản.
- Yêu cầu password của người dùng khi khởi tạo phải có độ dài tối thiểu 8 ký tự, bao gồm ký tự đặc biệt, chữ cái thường, chữ cái in hoa và số.
- Hệ thống cần được trang bị tường lửa (firewall), ngăn ngừa tấn công DDOS và các giải pháp bảo mật cho doanh nghiệp.

### *b. Performance*

- Tất cả màn hình input và output dữ liệu cần phải được sẵn sàng để hiển thị cho người dùng trong vòng 2 giây, với điều kiện tải và connection giữa client/ server là bình thường.
- Đối với màn hình input: tối đa 20 trường dữ liệu, không tính toán dữ liệu phức tạp, không tương tác với hệ thống ngoài, có thể lưu trữ dữ liệu trực tiếp ngay xuống database, không lưu trữ các tệp nội dung lớn như hình ảnh, video, tập tin quá 3 MB.
- Đối với màn hình output: dữ liệu được truy vấn trực tiếp từ database, hạn chế những câu lệnh truy vấn phức tạp, những truy vấn từ hệ thống ngoài. Hiển thị tối đa 50 dòng dữ liệu, mỗi dòng tối đa 10 cột, có độ dài nhỏ hơn 100 ký tự.
- Giao tiếp giữa các nhân viên (back officers, managers, janitors, collectors) có độ trễ dưới 1 giây.
- Điều kiện tải bình thường: 50 concurrent users (người dùng đồng thời) khi không dùng cân bằng tải.
- Điều kiện server tối thiểu: Intel Core i7, 8GB RAM, 1TB hard disk SSD.
- Client/Server Connection: 1MB/s.

### *c. Usability*



- Đối với collectors và janitors: tất cả thông tin quan trọng phải được hiển thị trong 1 màn hình (không cần phải thực hiện thêm thao tác cuộn).
- Phần mềm được sử dụng trong công ty cần có design và theme phù hợp với môi trường làm việc và tính chất của công ty, đảm bảo tính nghiêm túc, chuyên nghiệp.
- Khi người dùng nhập dữ liệu chưa chính xác, chỉ cần cho phép chỉnh sửa lại dữ liệu bị sai, không bắt buộc phải nhập lại toàn bộ dữ liệu. Kiểm tra kiểu dữ liệu... trên từng trường nhập và đưa ra cảnh báo tức thì nếu việc nhập liệu bị sai.
- Giao diện của hệ thống cần có sự nhất quán, về mặt hình ảnh biểu tượng cũng như vị trí các đối tượng trên màn hình để người dùng làm quen dễ dàng hơn.
- Người dùng có thể thành thạo các thao tác trên màn hình trong 15 phút sử dụng.

#### *d. Audit*

- Dữ liệu audit cần lưu trữ tách biệt trong một database riêng, độc lập hoàn toàn với database chính của hệ thống.
- Dữ liệu của hệ thống phải được sao lưu định kỳ, và tồn tại trong 30 ngày.
- Các dữ liệu audit phải ở chế độ Read Only và không cho phép chỉnh sửa từ giao diện người dùng.

#### *e. Extensibility*

- Có khả năng phát triển thêm tính năng theo dõi vị trí của các collectors và janitors mà không thay đổi cấu trúc dữ liệu cũ.
- Có khả năng phát triển thêm tính năng dẫn đường tới các MCPs để hỗ trợ janitors và collectors mà không thay đổi cấu trúc dữ liệu cũ.
- Có khả năng phát triển ngôn ngữ tiếng Anh trong tương lai mà không thay đổi cấu trúc dữ liệu cũ.

#### *f. Supportability*

- Hệ thống có thể được sử dụng hiệu quả trên các trình duyệt web (Opera, UC Browser, Safari, Microsoft Edge, Google Chrome, Samsung Internet Browser).
- Hệ thống không chạy trên các phiên bản trình duyệt quá cũ.

#### *g. Maintainability*

- Server có khả năng upgrade cấu hình

#### *h. Scalability*

- Cần nâng cấp, bảo trì hệ thống định kỳ 6 tháng một lần, mỗi lần bảo trì không quá 2 tiếng.

- Có khả năng tách database trên một server riêng và backend trên một server riêng.
- Có khả năng áp dụng cân bằng tải chạy bằng HAproxy.
- Có khả năng chia nhỏ database master thành các database con và ngược lại.

*i. Reliability*

- Trung bình số lần truy cập hệ thống thất bại là 1 trong 1000 lần truy cập.
- Xác suất hệ thống không khả dụng là dưới 0.01%
- Tỷ lệ xảy ra lỗi là dưới 0.03%

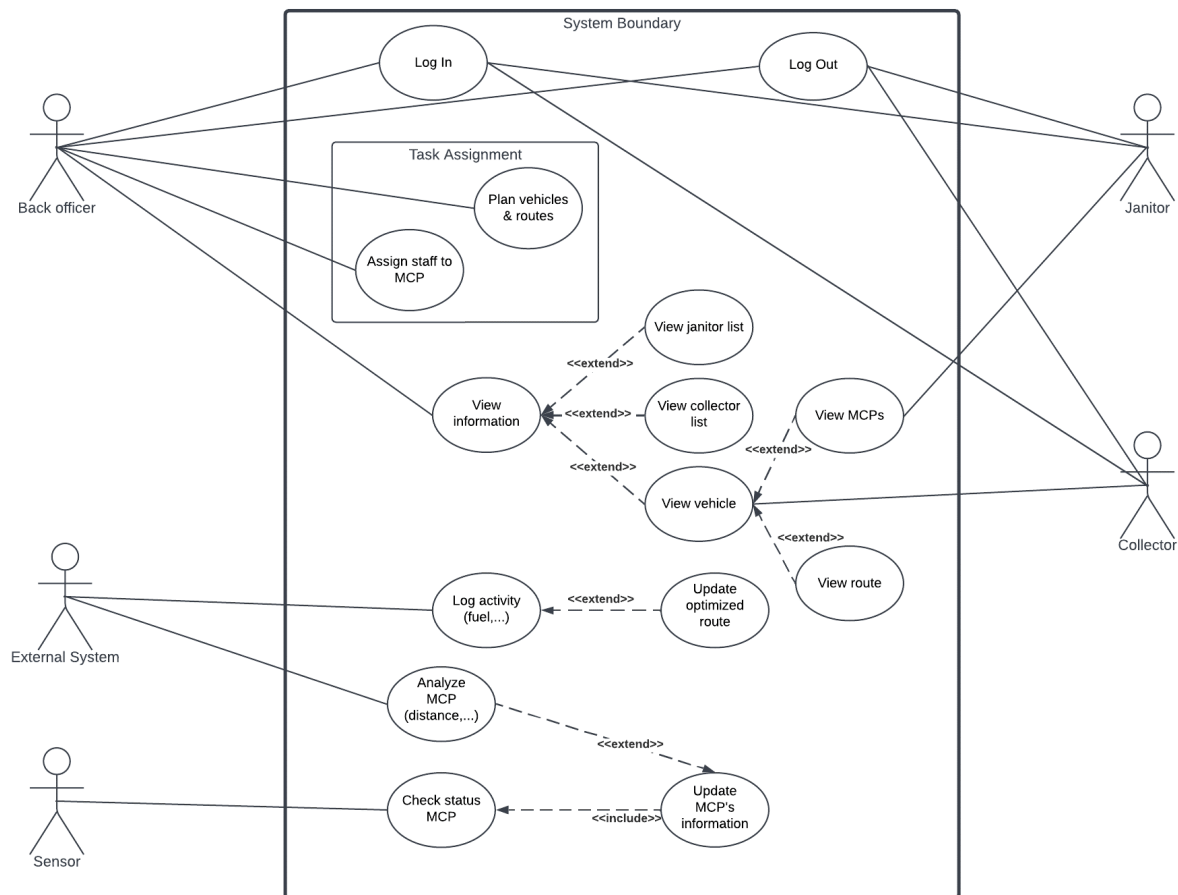
*j. Localization*

- Hệ thống sử dụng ngôn ngữ tiếng Việt.

### 3 Biểu đồ Use-case

#### 3.1 Biểu đồ use-case cho toàn hệ thống

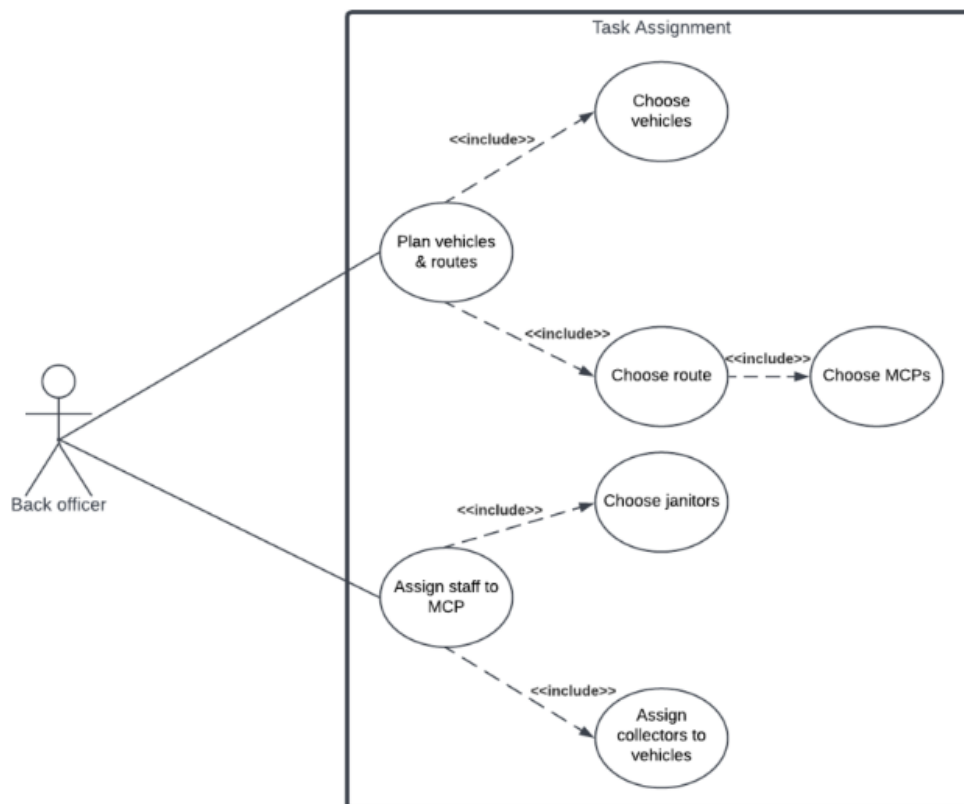
Từ những yêu cầu chức năng được liệt kê ở chương 2, nhóm đã vẽ một biểu đồ diagram tổng quát của dự án, với 5 actor.



Hình 1: Biểu đồ use-case cho toàn bộ hệ thống

#### 3.2 Biểu đồ use-case cho Task Assignment module

Task Assignment module bao gồm 3 chức năng: Lập kế hoạch về tuyến đường và phương tiện, Phân công Janitor đến mỗi MCP và Phân công phương tiện cho các Collector.



Hình 2: Biểu đồ use-case cho Task Assignment module

<b>Use-case name</b>	Plan vehicles & routes
<b>Description</b>	Hàng tháng, Back officer chọn phương tiện, tuyến đường đi qua các MCP được chỉ định.
<b>Trigger</b>	Back officer nhấn vào nút “Vehicle & Route Planning” ở trang “Task Assignment”.
<b>Actor</b>	Back officer
<b>Precondition</b>	Back officer đã đăng nhập vào UWC và phải đang ở trang “Task Assignment”.
<b>Postcondition</b>	Thông tin về vehicle, route được lưu vào database.
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>Hệ thống chuyển hướng sang trang “Vehicle &amp; Route Planning”</li> <li>Back officer gửi bản đồ khu vực hoạt động và danh sách các phương tiện</li> <li>Hệ thống hiển thị danh sách những kế hoạch đường đi (route plan) thỏa mãn điều kiện là bao quát toàn khu vực và tối ưu.</li> <li>Back officer chọn một kế hoạch đường đi phù hợp từ danh sách.</li> <li>Back officer nhấn nút “Confirm”</li> <li>Hệ thống lưu thông tin về kế hoạch đường đi</li> </ol>

**Bảng 1:** Mô tả use-case Plan vehicles routes

<b>Use-case name</b>	Assign janitor to MCP
<b>Description</b>	Hàng tuần, Back officer phân công janitor đến từng MCP.
<b>Trigger</b>	Back officer nhấn vào nút “Janitor Assignment” ở trang “Task Assignment”.
<b>Actor</b>	Back officer
<b>Precondition</b>	Back officer đã đăng nhập vào UWC và phải đang ở trang “Task Assignment”.
<b>Postcondition</b>	Thông tin về janitor được lưu vào database.
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>Hệ thống chuyển hướng qua trang mới “Janitor Assignment”</li> <li>Hệ thống hiển thị danh sách những MCP mà chưa được phân công janitor.</li> <li>Back officer chọn một MCP từ danh sách.</li> <li>Hệ thống hiển thị danh sách các janitor chưa được phân công mà có ca làm việc vào thời gian thực hiện task, bao gồm tên và ID.</li> <li>Back officer chọn các janitor từ danh sách.</li> <li>Back officer nhấn nút “Confirm”.</li> <li>Hệ thống cập nhật thông tin về danh sách janitor sẵn sàng nhận task.</li> <li>Hệ thống lưu thông tin của các janitor vào database và gửi thông báo đến janitor.</li> </ol>
<b>Alternative flow</b>	<p><i>Alternative 1: tại bước 6</i></p> <p>6a. Nếu Back officer nhấn nút “Cancel”, hệ thống chuyển hướng trở lại trang “Task Assignment”</p>
<b>Exceptions</b>	<p><i>Exception 1: tại bước 2</i></p> <p>2a. Nếu không có MCP nào trong danh sách các MCP chưa được phân công nhân sự, hệ thống hiển thị một cửa sổ thông báo tất cả MCP đều đã được phân công nhân sự. <i>Exception 2: tại bước 5</i></p> <p>5a. Nếu không có janitor nào trong danh sách, hệ thống hiển thị một cửa sổ thông báo không còn janitor nào để nhận task.</p>

**Bảng 2:** Mô tả use-case Assign janitors to MCP

<b>Use-case name</b>	Assign collector to vehicle
<b>Description</b>	Hàng tuần, Back officer phân công collector đến các phương tiện.
<b>Trigger</b>	Back officer nhấn vào nút “Collector Assignment” ở trang “Task Assignment”.
<b>Actor</b>	Back officer
<b>Precondition</b>	Back officer đã đăng nhập vào UWC và phải đang ở trang “Task Assignment”.
<b>Postcondition</b>	Thông tin về collector được lưu vào database.
<b>Normal flow</b>	<ol style="list-style-type: none"> <li>Hệ thống chuyển hướng qua trang mới “Collector Assignment”</li> <li>Hệ thống hiển thị danh sách các phương tiện chưa được phân công mà đã được lập kế hoạch để đi qua MCP dưới dạng biển số xe.</li> <li>Back officer chọn một phương tiện từ danh sách.</li> <li>Hệ thống hiển thị danh sách các collector chưa được phân công mà có ca làm việc vào thời gian thực hiện task, bao gồm tên và ID.</li> <li>Back officer chọn các collector từ danh sách cho phương tiện.</li> <li>Back officer nhấn nút “Confirm”.</li> <li>Hệ thống cập nhật thông tin về danh sách collector sẵn sàng nhận task.</li> <li>Hệ thống lưu thông tin của các collector vào database và gửi thông báo đến collector.</li> </ol>
<b>Alternative flow</b>	<p><i>Alternative 1: tại bước 6</i></p> <p>6a. Nếu Back officer nhấn nút “Cancel”, hệ thống chuyển hướng trở lại trang “Task Assignment”</p>
<b>Exceptions</b>	<p><i>Exception 1: tại bước 2</i></p> <p>2a. Nếu không có phương tiện nào trong danh sách, hệ thống hiển thị cửa sổ thông báo chưa có phương tiện nào để lập kế hoạch. <i>Exception 2: tại bước 5</i></p> <p>5a. Nếu không có collector nào trong danh sách, hệ thống hiển thị một cửa sổ thông báo không còn collector nào để nhận task.</p>

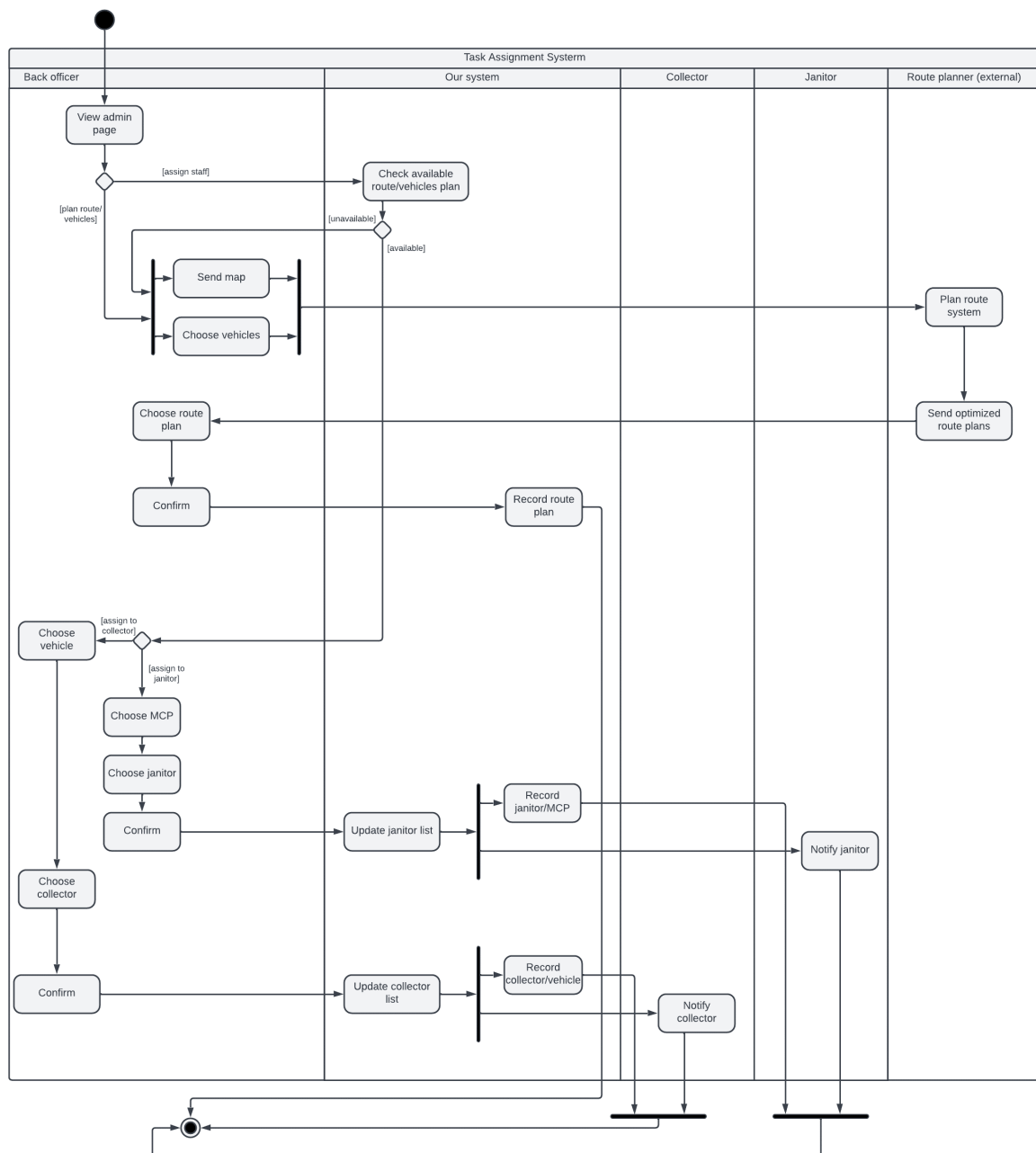
**Bảng 3:** Mô tả use-case Assign collector to vehicle

## 4 Mô hình hóa hệ thống

### 4.1 Activity Diagram cho Task Assignment module

Link activity diagram:

[https://lucid.app/lucidchart/c98eb3b7-8ad1-4e03-91fa-df26aeb52eb5/edit?beaconFlowId=6779E7ED4904ECB6&invitationId=inv\\_10d09e71-c759-4040-9d79-8ca9fb2f1e36&page=vBgYH~9PM.f6](https://lucid.app/lucidchart/c98eb3b7-8ad1-4e03-91fa-df26aeb52eb5/edit?beaconFlowId=6779E7ED4904ECB6&invitationId=inv_10d09e71-c759-4040-9d79-8ca9fb2f1e36&page=vBgYH~9PM.f6)



Hình 3: Activity diagram cho Task Assignment module

### Mô tả:

- Back officer truy cập vào hệ thống và bắt đầu lên kế hoạch Tuyến đường/Phương tiện (Plan Routes/Vehicles) hoặc phân công cho Janitor/Collector (Assign MCP to Janitor/Assign Vehicle to Collector).
- Hệ thống sẽ kiểm tra xem đã có kế hoạch về Tuyến đường/Phương tiện trong tháng này chưa.
  - Nếu chưa có, hệ thống sẽ thông báo Back Officer không đáp ứng được điều kiện để tiến hành phân công cho Janitor/Collector. Back officer chọn lên kế hoạch cho Tuyến đường/Phương tiện.
  - Nếu đã có, hệ thống sẽ thông báo Back Officer đã có kế hoạch về Tuyến đường/Phương tiện trong tháng và được phép phân công cho Janitor/Collector. Back officer chọn phân công cho Janitor/Collector.

### PLANNING ROUTE/VEHICLES

- Back officer tiến hành gửi bản đồ khu vực hoạt động và danh sách các phương tiện đến hệ thống **Route planner** (external system) để tính toán những kế hoạch đường đi khả thi tối ưu bao gồm hệ thống các tuyến đường đảm bảo bao quát toàn bộ khu vực và những phương tiện được phân cho từng tuyến đường.
- Back officer sẽ lựa chọn phương án kế hoạch đường đi phù hợp.
- Tiếp đó, Back officer hoàn tất xác nhận thay đổi.
- Sau khi xác nhận, hệ thống lưu lại thông tin về kế hoạch đường đi.

Back officer thực hiện phân công cho Janitor/Collector nếu đã có dữ liệu từ kế hoạch cho Tuyến đường/Phương tiện.

### ASSIGN MCPS TO JANITOR

- Hệ thống kiểm tra các điểm MCP được lên kế hoạch sử dụng nhưng chưa được phân công cho Janitor để hiển thị cho Back officer.
- Back officer sẽ phân công mỗi điểm MCP cho các Janitor lần lượt vào ca sáng – ca chiều – ca tối.
- Sau khi Back officer hoàn tất xác nhận thay đổi, hệ thống cập nhật danh sách janitor chưa được phân công.
- Cuối cùng, hệ thống lưu lại kế hoạch phân công của Janitor trong ngày hôm đó và gửi thông báo đến janitor.



## ASSIGN VEHICLES TO COLLECTOR

- Back officer lựa chọn Collector để phụ trách các Phương tiện thu gom rác.
- Hệ thống kiểm tra các phương tiện thu gom rác được lên kế hoạch sử dụng nhưng chưa được phân công điều khiển để hiển thị cho Back officer.
- Back officer sẽ phân công phương tiện thu gom rác cho các Collector lần lượt vào ca sáng – ca chiều – ca tối.
- Sau khi Back officer hoàn tất xác nhận thay đổi, hệ thống cập nhật danh sách collector chưa được phân công.
- Cuối cùng, hệ thống lưu lại kế hoạch phân công của collector trong ngày hôm đó và gửi thông báo đến collector.

## 4.2 Sequence Diagram và giải pháp khái quát cho Route Planning task

### 4.2.1 Assumption

Giả sử đã có thuật toán giải quyết được bài toán:

- Đề xuất tuyến đường tối ưu từ điểm  $A \Rightarrow B \Rightarrow A$  (với  $A$ ,  $B$  là điểm đầu điểm đầu, điểm cuối của tuyến đường các vehicle và collector đi qua:  $A = \text{depot}$  (điểm tập kết các vehicle),  $B = \text{Disposal Facility}$  (nhà máy xử lý rác) cuối cùng của tuyến đường).
- Xem xét các kết quả có khả năng, lựa chọn kết quả (chính xác hơn là lựa chọn các điểm MCP, Disposal Facility và tuyến đường sẽ đi qua) sao cho các tuyến đường phải đi đều có tổng khoảng cách đều nhau, và các tuyến đường (sẽ được phân cho các vehicle và collector) sẽ bao phủ đều, đi qua tất cả các MCPs trên thành phố.

Đồng thời, đảm bảo:

1. Mỗi ngày 1 MCPs sẽ được đảm bảo có ít nhất 2 vehicle và collector đi ngang qua, ca sáng 1 lượt + ca chiều/tối 1 lượt.
2. Số lượng xe vehicle luôn nhiều hơn số lượng collectors. Luôn có đủ các vehicle có dung tích  $\geq$  dung tích khối lượng rác trong ngày cao nhất của toàn thành phố (dựa trên báo cáo trong 5 năm gần nhất).
3. Các lựa chọn tuyến đường (route) có thể thay đổi để tối ưu hóa theo định kỳ mỗi tháng. Lịch trình assign task (route và vehicle) cho collector sẽ được thực hiện hằng ngày (bài toán lập trình và tối ưu hóa timetable).

4. Mỗi collector có thể làm 1 hoặc 2 ca (sáng + chiều/tối). Với mỗi ca trong ngày, collector sẽ được phân 1 vehicle cố định nhưng tuyến đường được phân công sẽ có thể khác nhau giữa ca sáng và ca chiều/tối sao cho phù hợp với lịch trình collectors và tối ưu hóa chi phí. Ngoài ca sáng, chiều/tối thì các collectors còn có thể lựa chọn các ngày trong tuần.
5. Khi Back Office thực hiện chức năng giao việc, hệ thống tự đề xuất các phương án tối ưu hóa sao cho công việc phù hợp với nhân sự, nhân lực, cơ sở vật chất và cố gắng đồng đều giữa các collectors với nhau trong 1 ca.
6. Đến 18:00 hằng ngày, các bên stakeholder sẽ cập nhật thông tin về lịch trình làm việc đăng ký, hiện trạng các vehicle để hệ thống cập nhật các thông tin, chuẩn bị đề xuất lịch trình và công việc phù hợp để Back Office đưa ra quyết định cho ngày hôm sau.

#### 4.2.2 Workflow

##### Định kỳ hàng tháng

Back officer, cơ quan quản lý đô thị, Công ty cung cấp dịch vụ rác thải Y cập nhật các thông số đầu vào cho bài toán route planning:

- Các con đường cho phép xe rác vehicle lưu thông (1 chiều hay 2 chiều, quy định về khung giờ).
- Số lượng xe, dung tích mỗi xe.
- Các thay đổi về MCPs, số lượng MCPs, tọa độ của các MCPs, dung tích tối đa MCPs.
- Số lượng, tọa độ của các Disposal Facility.
- Tình hình giao thông, thời tiết dự báo của tháng (tuyến đường nào hay ngập vào khung giờ nào, tuyến đường nào hay kẹt xe vào khung giờ nào).

Hệ thống, thuật toán có sẵn hoặc dịch vụ route planner bên ngoài trả về kết quả là các tuyến đường cho mỗi ca (sáng + chiều/tối) (lịch chung cho các ngày trong tuần) với các ưu tiên:

- Với mỗi ca, tập hợp các tuyến đường trả về đều bao phủ và đi qua đủ các MCPs.
- Tính theo tổng thể cả tuần thì chi phí, khoảng cách của các tuyến đường là tối ưu hóa nhất.
- Khoảng đường đi mỗi ca là không chênh lệch đáng kể.
- Trong từng tuyến đường tối ưu hóa khoảng cách (tránh các đoạn đường bị kẹt xe, bị ngập trong thời gian đó).

Đơn vị quản lý cập nhật số lượng, tình trạng, thể tích các vehicle có thể có của tháng đó. Hệ thống có sắp xếp các vehicle vào các tuyến đường có sẵn sao cho thỏa mãn điều kiện về dung tích tối đa.

### Định kỳ hàng ngày

- Đến 18:00 mỗi ngày, hệ thống ghi nhận các thông tin:
  - Tình trạng, thông tin đột xuất của các phương tiện từ đơn vị quản lý.
  - Tình trạng dự báo thời tiết + giao thông trong ngày (để cảnh báo).
- Hệ thống dựa theo các thông tin đã nhận được và lịch trình tuyến đường cho ngày => route planning.
- Hệ thống tự động gợi ý và Back Officer có thể điều chỉnh tùy chọn chọn giao các vehicle cho từng tuyến đường, các ca thời gian đăng ký sao cho thỏa mãn các điều kiện:
  - Không có sự trùng lặp, 1 vehicle được assign cho 2 tuyến đường của 2 ca.
  - Các xe đi qua số lượng MCPs phải thỏa mãn điều kiện về dung tích chứa chất thải tối đa của tất cả MCPs trên tuyến đường đó.
  - Tất cả các ca sáng và chiều đều thỏa mãn đi qua hết các MCPs.
  - Hệ thống báo lỗi khi các điều kiện trên không được thỏa mãn.

Tuyến đường	Ca(sáng hoặc chiều tối)	Xe
Depot => MCP 1 => MCP 2 => MCP 3 => Disposal Facility => Depot	Sáng	Truck 1 (6 khối)

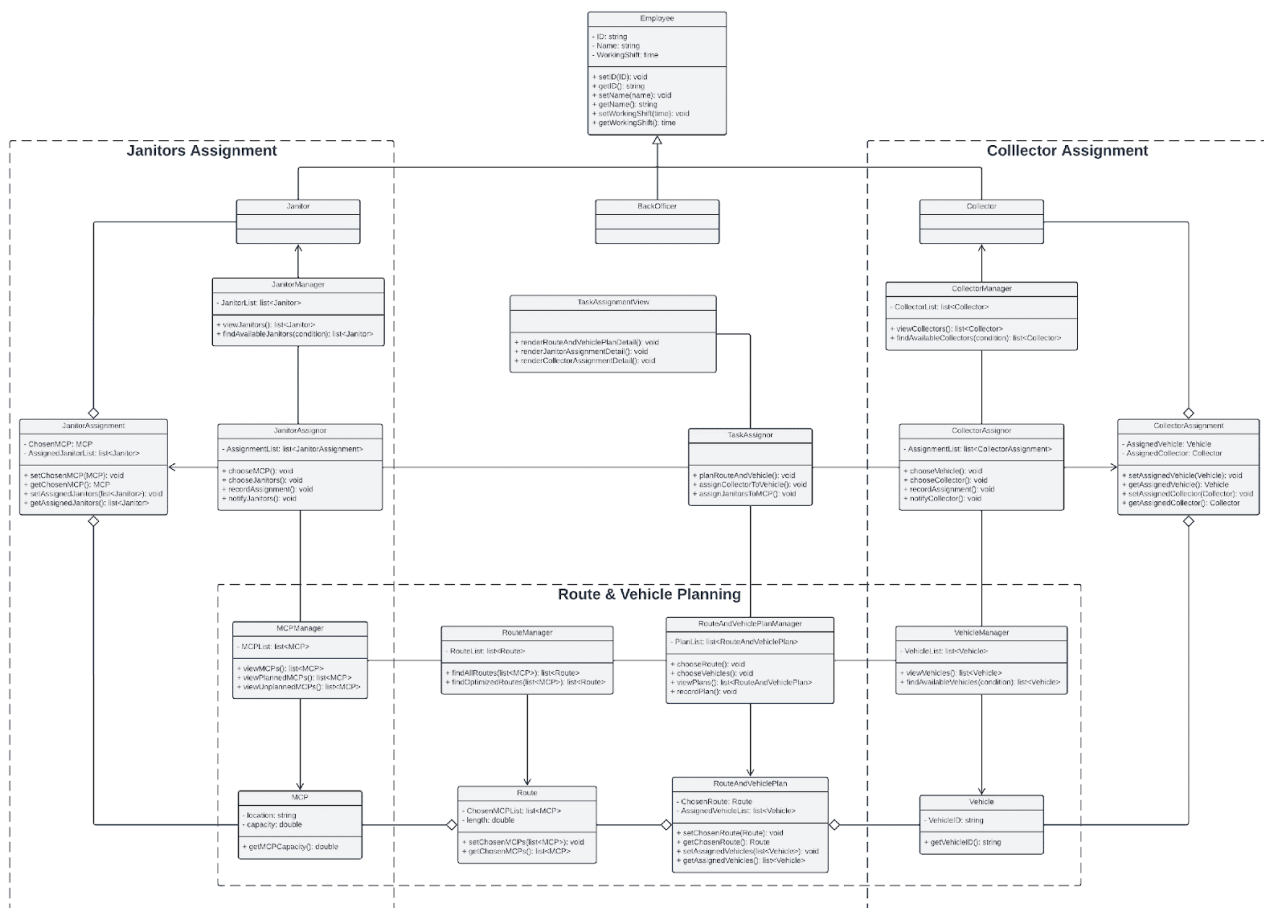
### Workflow

1. Back officer thực hiện các việc phân công khác ko đòi hỏi route planning.
2. Back Officer kiểm tra lại kết quả 1 lần nữa sau khi thực hiện phân công. => Lựa chọn thông báo đến các Collectors/ Janitors các thông tin phân việc.
3. Hệ thống gửi thông tin về công việc ngày tiếp theo cho các collectors, janitor.
4. Các Collectors, Janitors nhận được thông báo và xác nhận phân công hoặc phản hồi lại để Back Officer điều chỉnh (lặp lại bước 2).

## 4.3 Class Diagram cho Task Assignment module

Link class diagram:

[https://lucid.app/lucidchart/ac6ea127-dae9-4dd8-8624-e864bb32d7ba/edit?viewport\\_loc=153%2C-87%2C4561%2C2166%2C.J8YxEfRWXpa&invitationId=inv\\_937ae699-8d2c-458c-9084-c59c17764710](https://lucid.app/lucidchart/ac6ea127-dae9-4dd8-8624-e864bb32d7ba/edit?viewport_loc=153%2C-87%2C4561%2C2166%2C.J8YxEfRWXpa&invitationId=inv_937ae699-8d2c-458c-9084-c59c17764710)

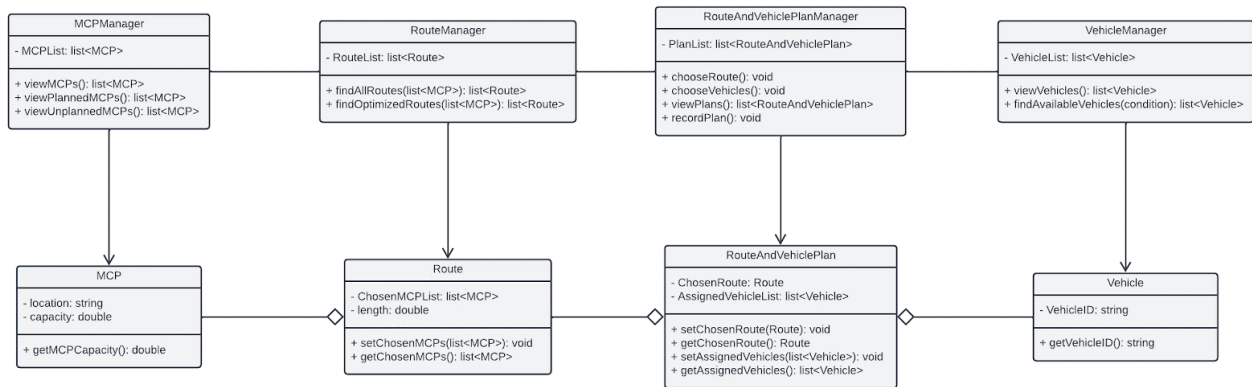


Hình 4: Class diagram cho Task Assignment module

Task Assignment bao gồm các thao tác Lập kế hoạch về tuyến đường và phương tiện (hàng tháng), giao phương tiện cho collector sử dụng và giao các janitor phụ trách mỗi MCP (hàng ngày). Từ đó, nhóm xác định các Entity class đầu tiên là Janitor, Collector, MCP, Vehicle, kèm theo các class quản lý tương ứng của chúng. Class diagram trên có thể được chia ra thành các diagram nhỏ hơn như sau:

#### 4.3.1 Route and vehicle planning

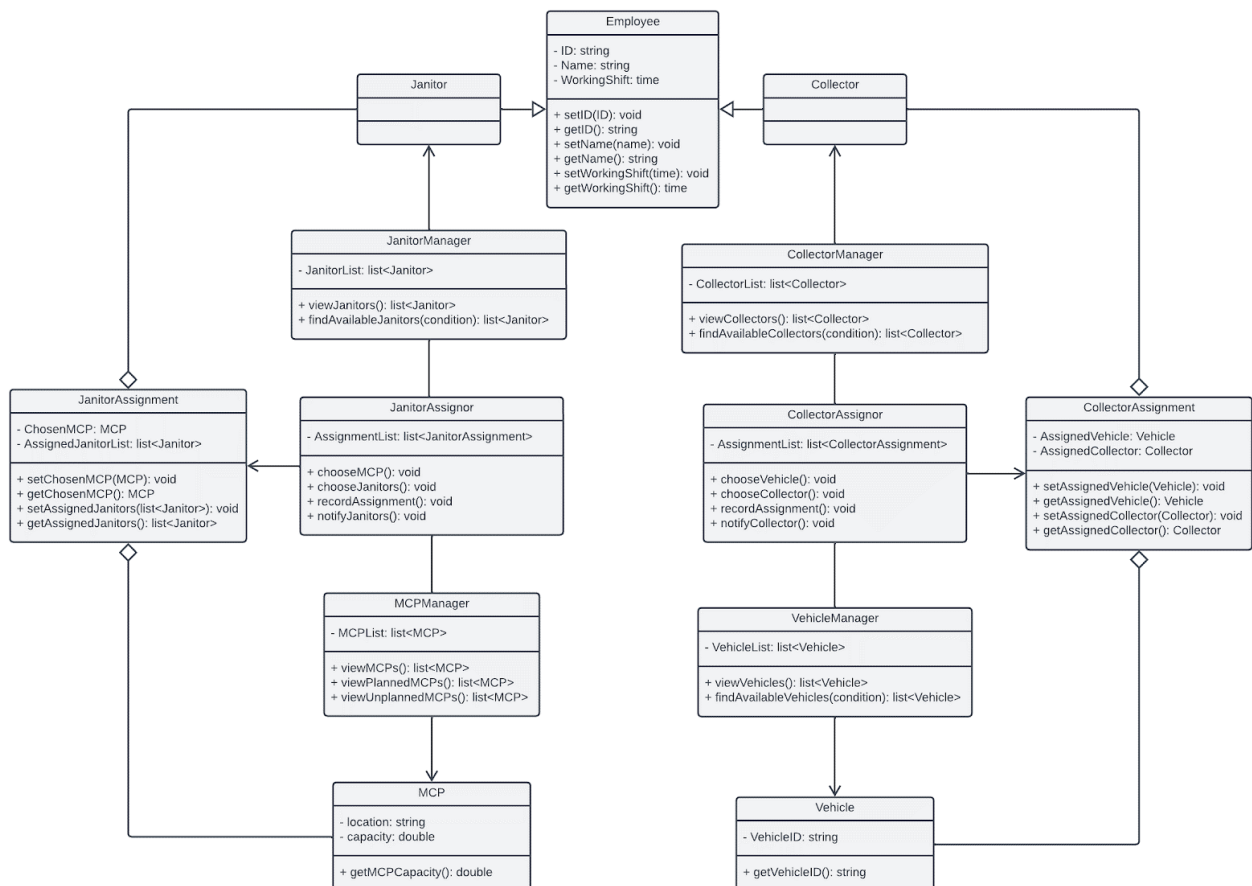
Một tuyến đường chứa một danh sách các MCP nằm trên tuyến đường đó, được biểu diễn bằng Entity class Route. Một bản kế hoạch bao gồm một tuyến đường và một danh sách các phương tiện sẽ được chỉ định đi trên tuyến đường đó, được mô hình hóa bằng class RouteAndVehiclePlan.



Hình 5: Class diagram cho thao tác lập kế hoạch tuyến đường và phương tiện

### 4.3.2 Collector and Janitor Assignment

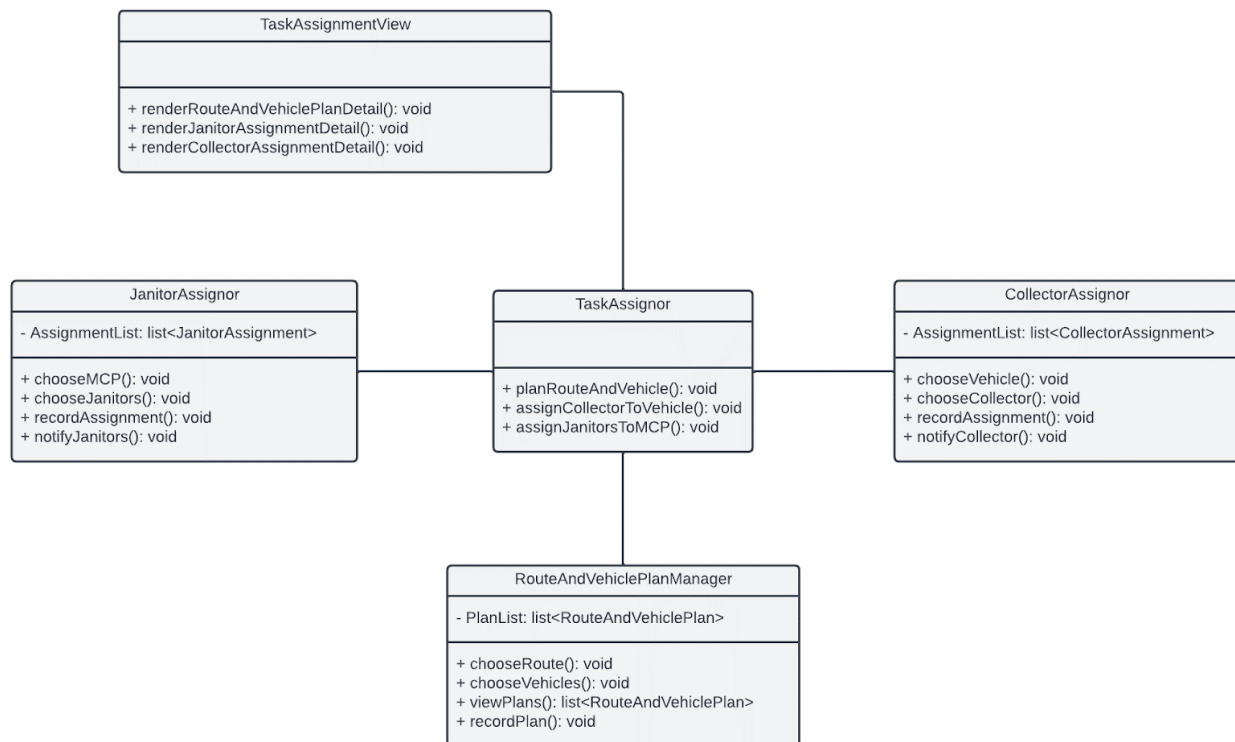
Mỗi MCP sẽ được giao cho một số Janitor, do đó class JanitorAssignment được tạo ra để chứa một đối tượng MCP và một danh sách các Janitor phụ trách MCP đó. Tương tự, một phương tiện sẽ được giao cho một Collector, vì thế class CollectorAssignment chứa một đối tượng Vehicle và một Collector điều khiển phương tiện.



Hình 6: Class diagram cho thao tác giao việc cho Janitor và Collector

### 4.3.3 Business và UI class

Class TaskAssignnor đóng vai trò như là một Controller nhận yêu cầu trực tiếp từ bên ngoài. Class TaskAssignmentView chịu trách nhiệm kết xuất kết quả và gửi ra màn hình.



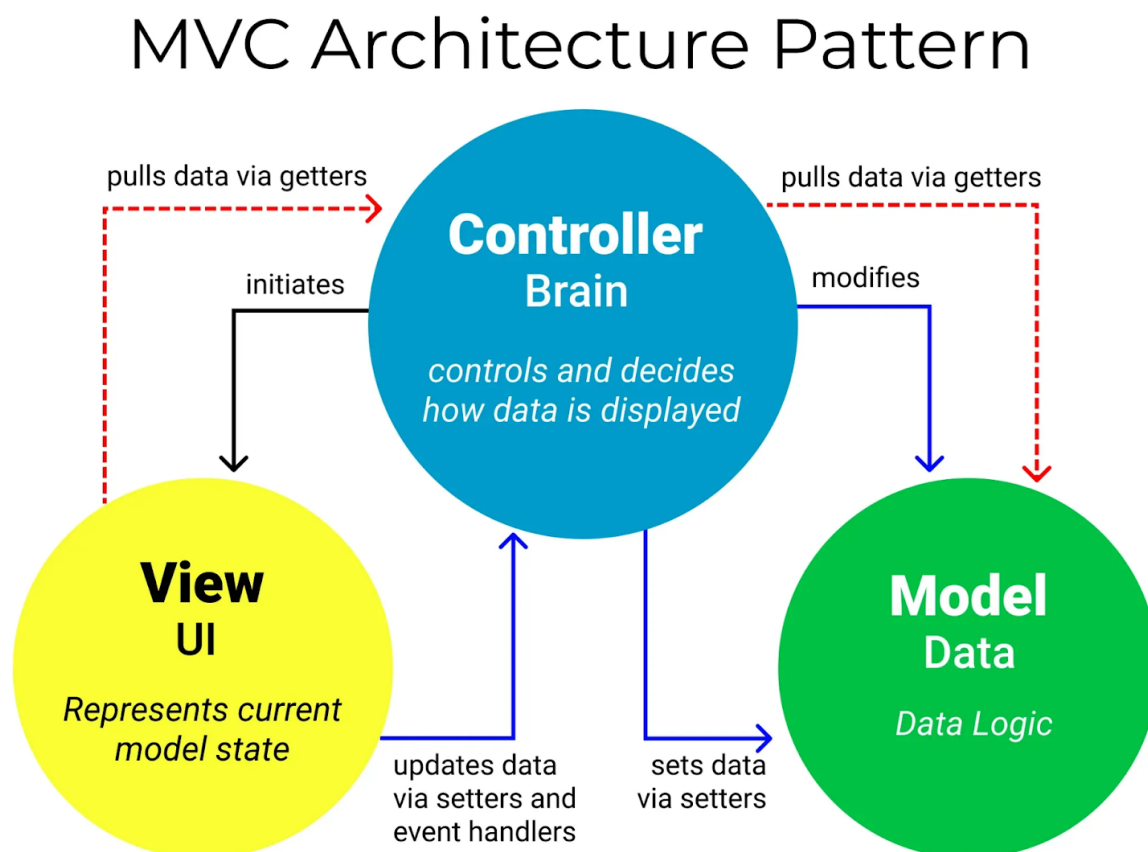
Hình 7: Controller và View

## 5 Thiết kế kiến trúc

### 5.1 Mô hình kiến trúc MVC

Nhóm lựa chọn kiến trúc MVC (Model-View-Controller) để hiện thực hệ thống UWC.

#### 5.1.1 Mô tả về kiến trúc MVC



Hình 8: Mô hình kiến trúc MVC

Model - View - Controller (MCV) là một kiến trúc thiết kế thường được ứng dụng trong phát triển website hoặc ứng dụng di động. Các hệ thống với kiến trúc MCV thường được chia thành 3 phần kết nối với nhau, mỗi phần có một nhiệm vụ riêng, độc lập. 3 phần trong kiến trúc MVC:

- Model: bao gồm các dữ liệu và logic liên quan đến xử lý dữ liệu.
- View: hiển thị dữ liệu đến người dùng hoặc xử lý tương tác người dùng.
- Controller: Điều khiển sự tương tác của hai thành phần Model và View.

Chi tiết hơn về từng bộ phận, ta có:

- Model: là bộ phận lưu trữ toàn bộ dữ liệu và cung cấp các logic, phương thức để tương tác với dữ liệu. Đối với hệ thống UWC, thành phần model:
  - Lưu trữ các dữ liệu về thông tin tuyến đường, ca trực, vehicle, lịch làm việc collectors và janitors,...
  - Cung cấp các phương thức để xử lý dữ liệu và tương tác với hệ cơ sở dữ liệu như việc đăng ký, đăng nhập tài khoản back officers, janitors, collectors, chỉnh sửa công việc phân công cho các janitors, collectors.
  - Thông báo các thay đổi về dữ liệu để view refresh và hiển thị bản cập nhật.
- View: Giao diện tương tác với người dùng, trình bày các dữ liệu tới người dùng.
  - Trình bày các yếu tố giao diện của hệ thống: menu, form, button...
  - Ghi nhận và truyền tải các tương tác của người dùng:
    - Ví dụ người dùng click vào dropdown menu => hiển thị menu.
    - Gửi các sự kiện về cho controller: ví dụ khi người dùng click submit form, view gửi thông báo đến controller.
    - Truyền các dữ liệu người dùng nhập vào: email, mật khẩu, tên đăng nhập cho controller.
  - Khi được thông báo các thay đổi trạng thái từ model, thực hiện truy xuất dữ liệu từ model để thực hiện refresh lại view và update view mới.
- Controller: Xử lý các yêu cầu người dùng nhận được thông qua view:
  - Hiển thị, trả về view các dữ liệu đã được định dạng, sắp xếp theo yêu cầu tương tác của người dùng.
  - Gọi các hàm từ model hiện thực chỉnh sửa, thao tác trên dữ liệu:
    - Cập nhật lịch trình, công việc phân công của janitors, collectors
    - Xóa tài khoản janitors, back officers
  - Xử lý xác thực dữ liệu: kiểm tra tính hợp lệ của dữ liệu nhập vào từ người dùng: email, tài khoản, mật khẩu.

### 5.1.2 Lý do lựa chọn MVC

- MVC tách biệt các phần Model, Controller và View
  - Dễ dàng cho việc kiểm thử, chỉnh sửa và mở rộng hệ thống
  - Nhiều lập trình viên làm việc đồng thời trên từng bộ phận khác nhau giúp quá trình phát triển hệ thống được nhanh chóng
- Đồng thời cung cấp nhiều khung View cho Model
- Hỗ trợ SEO: dễ dàng tạo các mã SEO URL để thu hút lượng truy cập.



### 5.1.3 Nhược điểm của MVC

- Đòi hỏi độ chính xác và phức tạp trong phần code hiện thực để 3 thành phần model - view - controller hoạt động, tương tác hiệu quả.

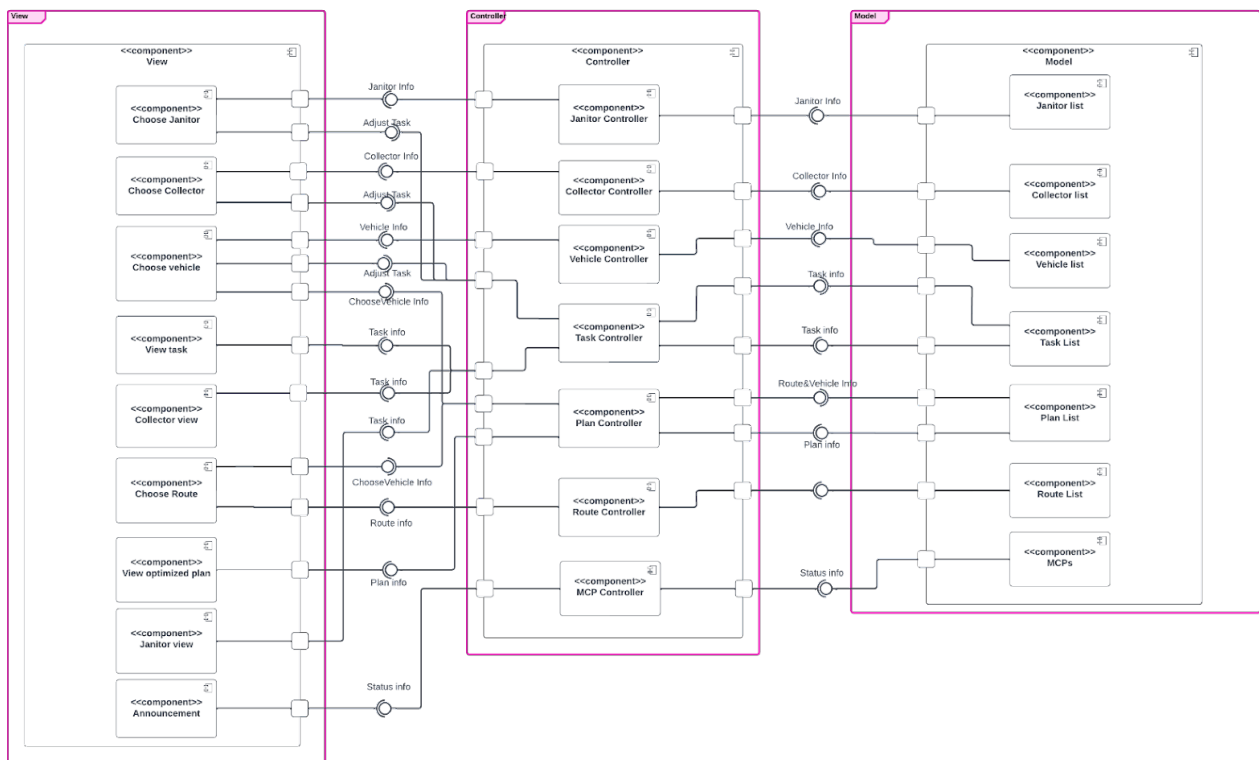
## 5.2 Modules

Module Name	Input	Output	Function
Authentication	Thông tin người dùng	Xác thực thông tin và quyền hạn của người dùng	Xác thực danh tính người dùng
Task Assignment	Số lượng nhân viên có mặt và kết quả hoạch định tuyến đường	Chi tiết công việc cho nhân viên	Phân chia công việc
Route Planning	Số lượng xe có sẵn, thông tin về vị trí của các MCPs trong khu vực	Những tuyến đường được tối ưu hóa và lịch trình cụ thể cho nhân viên phụ trách tuyến đường	Hoạch định tuyến đường
Vehicle Management	Tình trạng của các xe	Số lượng phương tiện có sẵn, số lượng phương tiện đang được sửa chữa, bảo dưỡng	Quản lý phương tiện
HR Management	Tình trạng nghỉ, vắng của nhân viên	Số lượng nhân viên hiện có, số lượng nhân viên vắng mặt	Quản lý nhân sự
Status Report	Thông tin nhân viên	Tình trạng công việc tại thời điểm báo cáo	Báo cáo tình trạng công việc

## 5.3 Component Diagram

Link component diagram:

[https://lucid.app/lucidchart/ac6ea127-dae9-4dd8-8624-e864bb32d7ba/edit?view\\_items=nr7YkndaIsiG&invitationId=inv\\_937ae699-8d2c-458c-9084-c59c17764710](https://lucid.app/lucidchart/ac6ea127-dae9-4dd8-8624-e864bb32d7ba/edit?view_items=nr7YkndaIsiG&invitationId=inv_937ae699-8d2c-458c-9084-c59c17764710)



Hình 9: Component diagram cho Task Assignment module

### Mô tả:

Trong giao diện của Back officer, có các chức năng View task, Choose vehicle, Choose route, Choose collector, Choose janitor.

Trong trang này, dữ liệu về Collector trong Collector List (Database) được tổng hợp bởi khối Collector Controller và xuất ra giao diện thông qua khối Choose Collector. Sau đó back officer thực hiện chọn Collector và Task Controller sẽ nhận được thông tin lựa chọn Collector từ khối Choose Collector. Tương tự với Choose Janitor và Choose Vehicle, sau đó Task Controller sẽ xử lý và gửi thông tin của Collector, Janitor, Vehicle tương ứng tới Task List.

Khối View Task sẽ yêu cầu thông tin của task và Task Controller lấy thông tin từ Task List (Database) và trả về cho khối View task hiển thị các task trong Task List.

Tương tự ta lựa chọn Route từ khối Choose Route và kết hợp với thông tin chọn Vehicle, Plan Controller sẽ nhận thông tin từ 2 khối Choose Route và Choose Vehicle, thực hiện xử lý và gửi thông tin của Route và Vehicle tương ứng tới Plan List.

Các dữ liệu về MCPs trong Database sẽ được tổng hợp qua khối MCP Controller và trả thông tin trạng thái (Status info) cho khối Announcement và thực hiện thông báo khi cần thiết.

Trong giao diện của Collector, các dữ liệu về Task List trong Database sẽ được tổng hợp qua khối Collector Controller và trả thông tin về task cho khối Collector view và hiển thị task.

Trong giao diện của Janitor, các dữ liệu về Task List trong Database sẽ được tổng hợp qua khối Janitor Controller và trả thông tin về task cho khối Janitor view và hiển thị task.