

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



CÔNG NGHỆ PHẦN MỀM (CO3001)

Urban Waste Collection Aid

UWC 2.0

GVHD:	Lê Đình Thuận	
Lớp:	L03	
SVTH:	Bùi Đoàn Gia Phong	2010509
	Nguyễn Đắc Hoàng Phú	2010514
	Lý Gia Huy	2010289
	Phan Hoàng Phúc	2010057
	Đặng Đức Danh	2010173
	Kazama Ken I Chi	2012722
	Tạ Lê Đắc Lộc	2010396

Tp. Hồ Chí Minh, Tháng 9/2022

Mục lục

1 Task 1: Requirement elicitation - Tìm hiểu yêu cầu	3
1.1 Xác định bối cảnh dự án - phân tích tình huống	3
1.1.1 Bối cảnh chung	3
1.1.2 Phân tích tình huống	3
1.1.2.a Stakeholders	3
1.1.2.b Những điều họ cần để thực hiện tốt công việc	3
1.1.2.c Những vấn đề gặp phải	4
1.1.2.d Giải pháp phần mềm - lợi ích mang lại cho các bên liên quan	4
1.2 Yêu cầu chức năng và phi chức năng	5
1.2.1 Yêu cầu chức năng	5
1.2.2 Yêu cầu phi chức năng:	5
1.3 Use-case diagram	8
1.3.1 Whole System	8
1.3.2 Task assignment module	9
2 Task 2: System Modelling - Mô hình hóa hệ thống	11
2.1 Activity Diagram của Task Assignment Module	11
2.1.1 Activity Diagram	11
2.1.2 Mô tả	11
2.2 Sequence Diagram của Route Planning Task	12
2.2.1 Sequence Diagram	12
2.2.2 Mô tả	12
2.3 Class Diagram của Task Assignment Module	13
2.3.1 Class Diagram	13
2.3.2 Mô tả	13
3 Task 3: Architecture design - Thiết kế kiến trúc	15
3.1 Architectural approach cho mô hình MVC	15
3.1.1 Diagram	15
3.1.2 Mô tả	15
3.2 Vẽ implementation diagram cho Task Assignment Module	17
3.2.1 Deployment Diagram	17
3.2.2 Component Diagram	17
Tài liệu tham khảo	19



History Log

Task	Thời gian	Nội Dung	Phân công	Thời lượng
1	15/09/2022	Chọn nhóm trưởng	Phong	1 giờ 45 phút
		Thảo luận tổng quát về đề tài. Chọn các công cụ để hiện thực đề tài. Xác định tổng quát về các đối tượng và những vấn đề có thể gặp phải trong các task.	All	
		Task 1.1	Phú, Huy	
		Task 1.2	Phúc, Ken	
		Task 1.3	Danh, Lộc	
		LaTeX	Lộc	
	18/09/2022	Thảo luận về từng phần của từng nhóm nhỏ đã được phân công.	All	2 giờ
		Sửa một số phần như use case, functional, non-functional trong các task nhỏ.		
		Set thời gian cho các deadline tiếp theo.		
	21/09/2022	Hoàn thành các chỉnh sửa cuối cùng của các task nhỏ trong task 1.	All	1 giờ
		Phân công gõ báo cáo hoàn thành task 1.		
		Phân công tìm hiểu các công cụ để chuẩn bị cho các task tiếp theo.		

1 Task 1: Requirement elicitation - Tìm hiểu yêu cầu

1.1 Xác định bối cảnh dự án - phân tích tình huống

1.1.1 Bối cảnh chung

Với những nước đang phát triển nói chung và với Việt Nam nói riêng, vấn đề về môi trường sống trước những tác động từ sự gia tăng dân số và sự phát triển kinh tế luôn được quan tâm hàng đầu. Đặc biệt là tại các khu đô thị lớn, nơi mà mật độ dân cư đông đúc, các hoạt động sản xuất kinh doanh cũng như lượng hàng hóa tiêu dùng đều lên đến những con số khổng lồ thì vấn đề môi trường sống cần được bảo vệ ra sao càng trở nên cấp thiết hơn bao giờ hết. Và đặc biệt là vấn đề chất thải rắn sinh hoạt, nếu không được xử lý, quản lý bằng một chu trình khoa học sẽ gây ra nhiều hệ lụy đến môi trường sống của chúng ta.

Tăng trưởng kinh tế cùng với tốc độ gia tăng dân số nhanh và quá trình đô thị hóa diễn ra trên diện rộng là các nguyên nhân dẫn đến tình trạng ô nhiễm môi trường, trong đó có vấn đề quản lý chất thải rắn ngày càng trầm trọng tại các đô thị lớn - nơi các bãi rác ngày càng gia tăng cả về số lượng, thành phần. Điều đó đã tạo nên những áp lực lớn đối với các cơ quan quản lý nhà nước trong quản lý chất thải rắn sinh hoạt.

Trong ngữ cảnh đó nếu cứ tiếp tục duy trì sử dụng các chu trình quản lý rác hiện tại về lâu dài sẽ không đáp ứng được xu hướng của sự phát triển. Do vậy, vấn đề cấp thiết hiện nay là cải thiện hệ thống quản lý việc thu gom rác thải vốn có để nâng cao hiệu quả tiến trình, giảm bớt tiêu tốn chi phí của cả tiến trình.

Vì vậy, việc phát triển hệ thống UWC 2.0 để cải thiện hiệu quả của việc thu gom chất thải rắn sinh hoạt là phù hợp với xu thế phát triển nhanh chóng của xã hội. Hệ thống UWC 2.0 cần phải đáp ứng được các đặc tính sau:

- Được phát triển cải tiến từ UWC 1.0 với một cơ sở dữ liệu có sẵn;
- Đơn giản, dễ sử dụng, phù hợp với yêu cầu của các bên liên quan;
- Giải quyết được các vấn đề đặt ra của các bên liên quan;
- Tối ưu được lợi ích của các bên liên quan.

1.1.2 Phân tích tình huống

1.1.2.a Stakeholders

Stakeholders hay các bên liên quan có thể được coi là một đối tượng hay là một tổ chức có ràng buộc và bị ảnh hưởng bởi hệ thống. Họ sử dụng hệ thống cho mục đích cá nhân và của tập thể.

Các stakeholders liên quan đến phần mềm này có thể kể đến như sau:

- **Back officer:** giữ vai trò vận hành hệ thống để tạo lịch trình, điều phối và sắp xếp công việc cho janitors và collectors (sẽ giải thích sau).
- **Collector:** giữ vai trò lái các loại phương tiện cơ giới.
- **Janitor:** giữ vai trò thu gom rác từ các điểm thu gom chính (MCP) thông theo cách thủ công qua xe đẩy (trolley).
- **Technician:** giữ vai trò là những chuyên viên kỹ thuật và nghiên cứu, họ dùng dữ liệu của quá khứ để tìm hiểu, phân tích và đưa ra các quyết định giúp cải thiện hiệu suất công việc trong tương lai.
- **Customer:** đóng vai trò là khách hàng sử dụng dịch vụ dọn vệ sinh.

1.1.2.b Những điều họ cần để thực hiện tốt công việc

Back officer: cần có một cái nhìn tổng quan về người vệ sinh và người thu gom, lịch làm việc của họ cũng như tổng quan về các loại xe và các chi tiết kỹ thuật của chúng (trọng lượng, công suất, mức tiêu thụ nhiên liệu, v.v.) hay về các địa điểm MCP và sức chứa của nó. Bên cạnh đó cần có giao diện điều khiển để trực quan phải được cập nhật hàng ngày, có thể kịp thời gửi đến các thông tin cho các nhân viên từ đó điều chỉnh tiến độ cho phù hợp.

Collector và Janitor: cần một cái nhìn tổng quan và chi tiết về nhiệm vụ cơ bản của họ hàng ngày và hàng tuần. Liên lạc với những thành viên khác và nhân viên phụ trách. Họ cũng cần cơ chế checkin/checkout công việc và được thông báo về MCP nếu lượng tải đầy. Có các quyền lợi cơ bản của người lao động: xin nghỉ phép, yêu cầu ngừng làm việc,...

Technician: Cần theo dõi quá trình và thông tin thu gom chất thải một cách tốt nhất và đầy đủ nhất, cũng như có thể liên lạc kịp thời nếu có vấn đề xảy ra với back officer. Cuối cùng Họ dùng dữ liệu của quá khứ để tìm hiểu, phân tích và đưa ra các quyết định giúp cải thiện hiệu suất công việc trong tương lai và tối ưu hóa tuyến đường về mức tiêu thụ nhiên liệu và khoảng cách di chuyển cho mỗi phương tiện.

Customer: Họ cần theo dõi, đánh giá hiệu quả mà dịch vụ mang lại ở khu vực họ sinh sống hay các cảnh quan khác chẳng hạn: vệ sinh sạch sẽ, đổ rác đúng giờ,... và có thể thanh toán các khoản phí liên quan một cách thuận tiện nhất.

1.1.2.c Những vấn đề gặp phải

Back officer: Vì đây là công việc liên quan nhiều đến con người nên họ cần một giao diện bắt mắt và thân thiện để dễ dàng tương tác. Họ sẽ gặp khó nếu cách phân công công việc, lịch trình và nhân sự nếu bị rắc rối mất thông tin, dữ liệu về mặt trình bày, hiển thị. Nếu thông tin được cập nhật chậm hay không thể tương tác trực tiếp với các nhân viên một cách nhanh chóng, công việc của họ và chất lượng của cả dịch vụ chất lượng đều sẽ đi xuống.

Collector và Janitor: Thường thì họ sẽ không giàu nên điện thoại thường thuộc loại thô sơ, đối với những người lớn tuổi sẽ không có nhiều kinh nghiệm về công nghệ. Họ sẽ gặp khó trong việc sử dụng phần mềm nhất là tương tác để cập nhật thông tin liên tục với Back officer.

Technician: Vì đây là công việc nghiên cứu và phát triển nên họ cũng cần thông tin được cập nhật chính xác và liên tục. Họ sẽ gặp rắc rối nếu công cụ thiếu trực quan, thiếu các đánh giá và thiếu các phương pháp hỗ trợ để đưa ra quyết định.

Customer: Làm thế nào mà họ có thể đánh giá chất lượng của dịch vụ như thùng rác được loại bỏ, cảnh quan của khu vực,... làm thế nào mà họ có thể liên lạc trực tiếp với các back officer hay các nhân viên khác nếu có thiệt hại xảy ra và họ cũng cần được biết mức độ thuận tiện khi thanh toán các khoản phí liên quan đến dịch vụ.

1.1.2.d Giải pháp phần mềm - lợi ích mang lại cho các bên liên quan

- Back officer:

- + Thuận lợi để quản lý, thu gom trực tiếp và người dọn dẹp.
- + Sắp xếp lộ trình phù hợp cho xe đi, chuẩn bị phí đổ xăng, sửa chữa.
- + Sắp xếp hợp lý thời gian thu gom rác, theo dõi phương tiện di chuyển, công việc xung quanh đó.
- + Có thể giao tiếp với nhân viên khác một cách dễ dàng hơn.

- Collector và Janitor:

- + Giao diện thân thiện và dễ sử dụng.
- + Hỗ trợ linh hoạt trong việc di chuyển và thực hiện công việc.
- + Định hướng công việc rõ ràng, tối ưu hóa lịch trình đi lại.
- + Cách thức liên hệ và thông báo dễ dàng với backofficer và những thành viên khác.

- Technician:

- + Tiết kiệm được công sức thu thập và xử lý thông tin.
- + Hỗ trợ mạnh mẽ trong việc ra quyết định và linh hoạt trong xử lý tình huống.

- + Các phương pháp thống kê và Tối ưu hóa hiệu quả trong việc vận chuyển.

- **Customer:**

- + Thuận tiện trong việc liên hệ và giao tiếp với back officer.
- + Các quá trình xử lý nhanh chóng và tự động
- + Tính năng bình luận và đánh giá chất lượng dịch vụ.

1.2 Yêu cầu chức năng và phi chức năng

1.2.1 Yêu cầu chức năng

- **Customer:**

- + Khách hàng có thể quản lý tài khoản cá nhân trên hệ thống (đăng ký, đăng nhập, điều chỉnh thông tin cá nhân).
- + Khách hàng sẽ nhìn thấy tổng quan về tên, địa điểm trực nhật, ca, ngày, giờ làm việc của nhân viên thu gom và nhân viên vệ sinh.
- + Khách hàng có thể viết những bài bình luận, đánh giá cho cán bộ phụ trách.
- + Khách hàng cũng có thể thanh toán tiền rác qua phần mềm.

- **Back Officer:**

- + Cán bộ phụ trách đăng nhập vào hệ thống để làm việc.
- + Cán bộ phụ trách có thể quản lý (xem, thêm, sửa, xoá) thông tin nhân viên.
- + Cán bộ phụ trách có thể quản lý (xem, thêm, sửa, xoá) phương tiện lao động và các chi tiết kỹ thuật của chúng (trọng lượng, công suất, mức tiêu thụ nhiên liệu, v.v.).
- + Có thể điều chỉnh và chỉ định các MCP cho nhân viên thu gom và nhân viên vệ sinh.
- + Tạo các tuyến đường cho nhân viên thu gom.
- + Có thể gửi tin nhắn cho nhân viên và khách hàng.

- **Collector và Janitor:**

- + Nhân viên vệ sinh đăng nhập vào hệ thống để làm việc.
- + Có thể chỉnh sửa được thông tin cá nhân của chính họ.
- + Nhìn thấy tổng quan về lịch làm việc của họ.
- + Nhìn thấy chi tiết nhiệm vụ hàng ngày cũng như hàng tuần của họ.
- + Có thể giao tiếp với những nhân viên khác, các cán bộ phụ trách và khách hàng.
- + Check in/ check out công việc mỗi ngày.
- + Được thông báo về địa điểm MCP và các tuyến đường cần phải làm việc.

- **Technician:**

- + Kỹ thuật viên đăng nhập vào hệ thống để làm việc.
- + Phân tích, trực quan hoá để đưa ra những quyết định cho công việc hiệu quả trong tương lai (sắp xếp lại gốc (rescheduling root), khoảng cách, đường đi của các điểm MCP,...).
- + Có thể giao tiếp và liên lạc với những nhân viên khác, các cán bộ phụ trách và khách hàng.

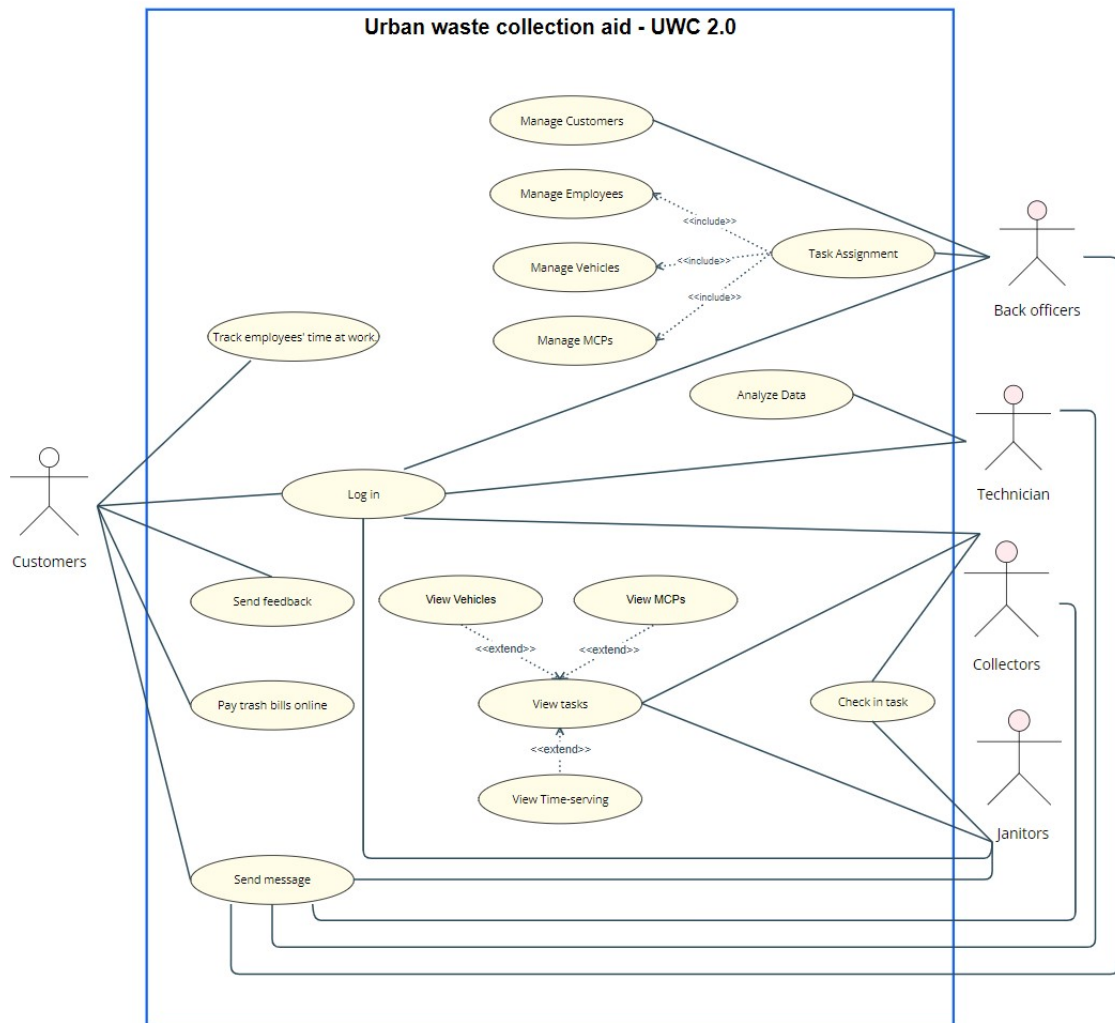
1.2.2 Yêu cầu phi chức năng:

STT	Tên chức năng	Mô tả chức năng
1	Tốc độ, hiệu năng (Performance)	<ul style="list-style-type: none">- Các quy trình của ứng dụng web load không quá 3 giây.- Các tin nhắn phải được truyền đạt theo cách thức thời gian thực với độ trễ ít hơn 1 giây.- Thông tin phải được cập nhật từ MCP sau mỗi 15 phút với tính khả dụng của ít nhất 95% thời gian hoạt động của chúng.- Tuyến đường được chỉ định được tối ưu hóa về mức tiêu thụ nhiên liệu và quãng đường di chuyển.
2	Bảo mật truy cập (Access Security)	<ul style="list-style-type: none">- Khi người dùng tham gia vào website bị quên mật khẩu, người dùng sẽ được yêu cầu lấy lại mật khẩu từ email đăng ký, một link khôi phục mật khẩu sẽ được gửi tới email đăng ký của người dùng.
3	Khả năng tiếp cận (Accessibility)	<ul style="list-style-type: none">- Các tính năng trên web có thể âm thanh, hình ảnh minh hoạt để phục vụ người dùng.
4	Tính sẵn có (Availability)	<ul style="list-style-type: none">- Người dùng có thể vào website trực tuyến 24/7 ngoại trừ các thời gian bảo trì hệ thống sẽ thông báo trước đến qua email và thông báo tại website.
5	Bảo mật (Confidentiality)	<ul style="list-style-type: none">- Khi thanh toán khóa điện tử thông qua website sẽ không lưu lại các thông tin quan trọng (số thẻ, số cccd/cmnd...) của khách hàng được nhập trong quá trình thanh toán.- Cảnh báo nếu như có IP máy chủ khác xâm nhập.- Trang Web có thể ngăn ngừa tấn công DDOS.- Chức năng thanh toán phải áp dụng hoặc sử dụng lại hệ thống có các cơ chế bảo mật: SSL, SET, PCI DSS.
6	Tính toàn vẹn (Integrity)	<ul style="list-style-type: none">- Lịch sử người dùng được lưu lại.- Tài khoản đăng nhập được lưu lại để lần tiếp theo không cần nhập mật khẩu.
7	Độ tin cậy (Reliability)	<ul style="list-style-type: none">- Xác suất trang web bị lỗi khi người dùng đăng nhập vào học là 1/10000.- Tỷ lệ xảy ra lỗi là dưới 0.05% .

8	Tính đa dụng (Usability)	<ul style="list-style-type: none">- Người dùng dễ dàng gửi câu hỏi đến các cán bộ phụ trách qua ứng dụng chat trực tuyến có sẵn tại website, có thể dễ dàng chụp ảnh màn hình, ghi lại các thắc mắc để gửi các cán bộ phụ trách.- Giao diện thân thiện, gần gũi. Người dùng dễ dàng thành thạo việc sử dụng các tính năng của website.
9	Đa nền tảng (Multi-platform)	<ul style="list-style-type: none">- Hệ thống có thể sử dụng hiệu quả trên điện thoại di động (Android, IOS), máy tính bảng hay máy tính bàn, laptop (Windows, Linux, Mac) với các trình duyệt (Chrome, Firefox, Safari, Opera).
10	Tính bảo trì (Maintainability)	<ul style="list-style-type: none">- Mỗi lần nâng cấp, bảo trì hệ thống định kỳ (theo quý 3 tháng) với thời gian tùy thuộc vào mức độ bảo trì/ nâng cấp.
11	Tính mở rộng của hệ thống (Scalability)	<ul style="list-style-type: none">- Trong tương lai, hệ thống có thể mở rộng ở nhiều nơi khác.
12	Kích thước (Size)	<ul style="list-style-type: none">- Tổng dung lượng của các file load về thiết bị của người dùng không quá 300MB.
13	Tính bản địa (Localization)	<ul style="list-style-type: none">- Hỗ trợ cả tiếng Anh và tiếng Việt.
14	Tính linh hoạt (Flexibility)	<ul style="list-style-type: none">- Có thể đăng nhập ở mọi nơi trên thế giới.
15	Hỗ trợ trực tuyến người dùng	<ul style="list-style-type: none">- Nút help xuất hiện trong màn hình. Khi người dùng bấm vô thì mở ra một trang online documentation, lưu trữ toàn bộ tài liệu hướng dẫn để người dùng tham khảo ngay tức thì.- Khi người dùng thực hiện một thao tác lỗi, hệ thống hiển thị một popup message hướng dẫn cách khắc phục ngay.
16	Khả năng quản lí (Manageability)	<ul style="list-style-type: none">- Trang web sẽ duy trì hoạt động ngay cả khi các cán bộ phụ trách chỉnh sửa thông tin nhân viên hoặc thông tin người dùng.

1.3 Use-case diagram

1.3.1 Whole System



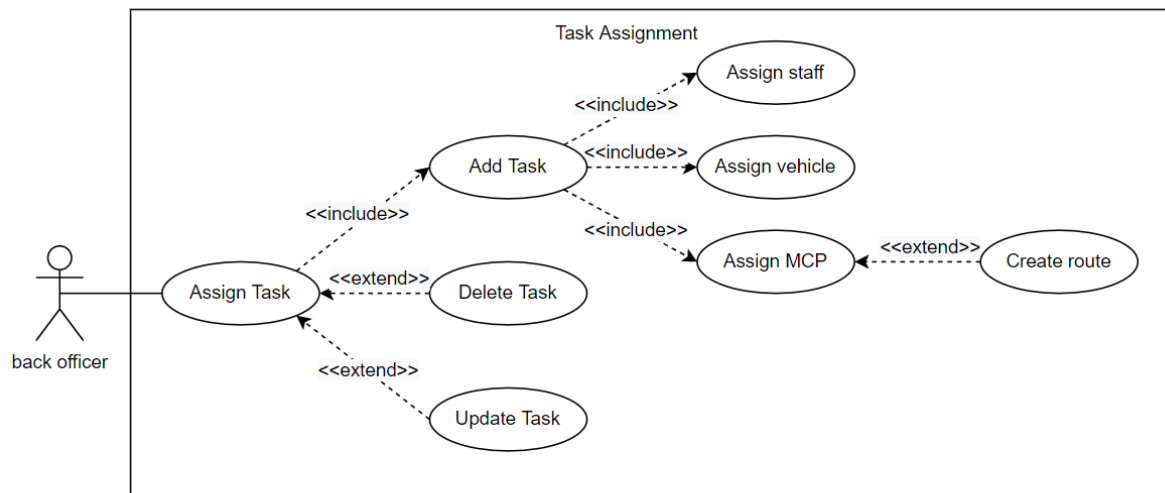
Actor ID	Tên Actor
1	Customers (Khách hàng)
2	Back Officers (Cán bộ phụ trách)
3	Collectors (Nhân viên thu gom)
4	Janitors (Nhân viên vệ sinh)
5	Technician (Kỹ thuật viên)

Bảng 1: Bảng danh sách các Actor

Use case ID	Tên use case	Mô tả
1	Log in	Dùng để đăng nhập tài khoản người dùng.
2	Track Employees' time at work	Theo dõi giờ, ngày, ca làm việc, địa điểm làm việc của nhân viên thu gom và nhân viên vệ sinh.
3	Send feedback	Gửi phản hồi và đánh giá dịch vụ công ty.
4	Pay trash bills online	Thanh toán hoá đơn tiền rác trực tuyến.
5	Send message	Nhắn tin cho nhau.
6	Task assignment	Phân công nhiệm vụ bao gồm: + Quản lý phương tiện lao động (Manage Vehicles) + Quản lý nhân viên (Manage Customers) + Quản lý các MCP (Manage MCPs)
7	Manage customers	Quản lý khách hàng.
8	View task	Xem công việc của họ. Tại đây họ có thể xem được phương tiện lao động cần sử dụng, các địa điểm MCP và thời gian làm việc của họ.
9	Check in task	Bắt đầu nhận việc, nhận công việc.
10	Analyze Data	Phân tích các dữ liệu và số liệu của phần mềm.

Bảng 2: Bảng mô tả các usecase chính của hệ thống

1.3.2 Task assignment module



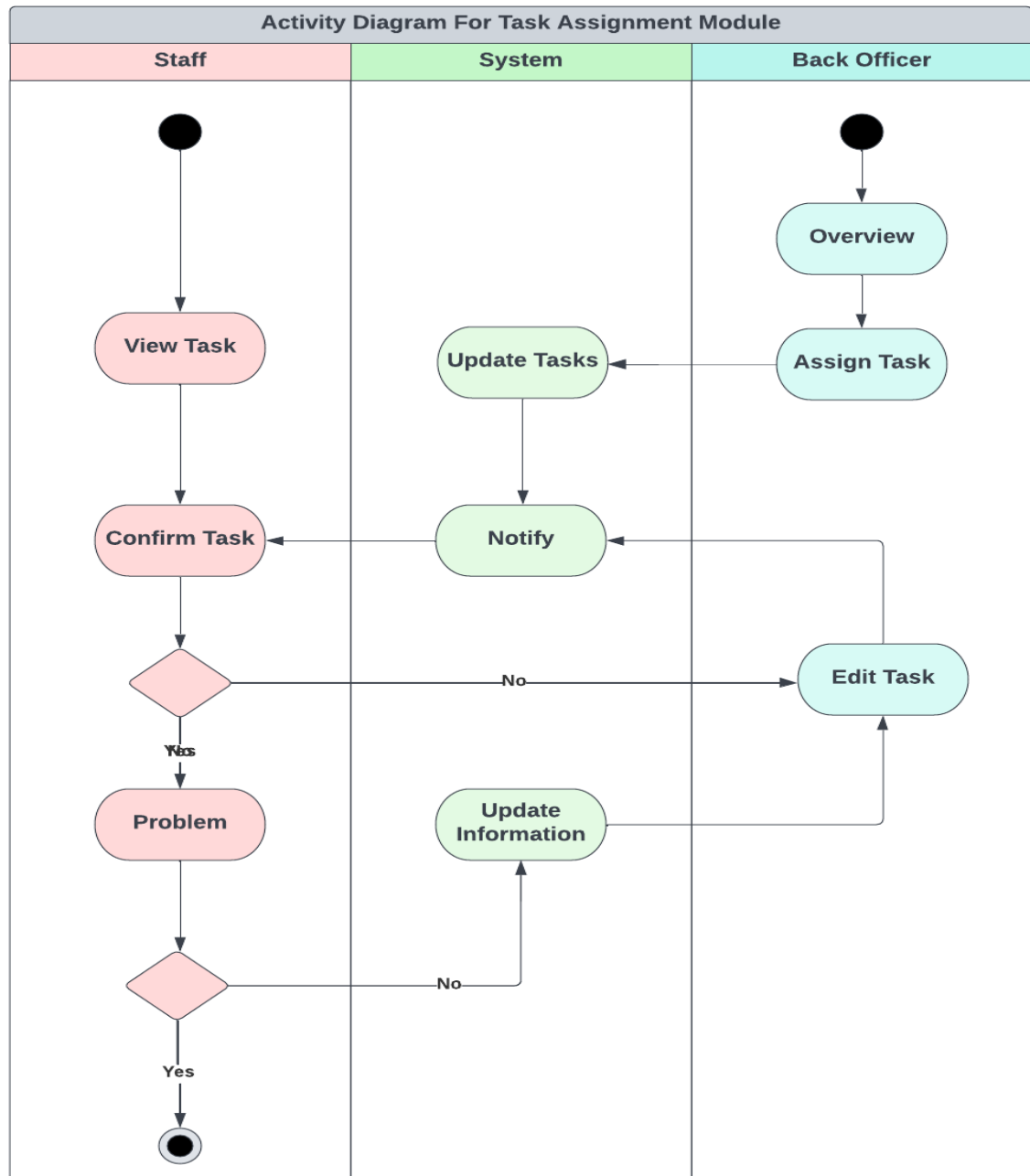
Use-case Name	Task Assignment
Actor	Back officer
Description	Back officer có thể dùng chức năng này để thực hiện thao tác bao gồm: thêm, xóa, sửa cho công việc của nhân viên
Trigger	Nhấn vào "Assign Task"
Precondition	+ User phải là back officer + User có quyền truy cập vào hệ thống Task Assignment và cơ sở dữ liệu của hệ thống
Normal Flow	1. Hệ thống hiển thị giao diện chính với lựa chọn thêm việc mới. Mỗi công việc có thêm 2 lựa chọn là xóa và sửa. 2. Back officer nhấn 1 trong 3 lựa chọn trên: + Nếu chọn vào "Add Task", hệ thống yêu cầu 3 hành động "Assign staff", "Assign vehicle" và "Assign MCP" cho back officer. Nếu chọn nhiều MCP, hệ thống yêu cầu back officer "Create route" cho chúng. + Nếu chọn vào "Delete Task", công việc sẽ bị xóa. + Nếu chọn "Update Task", hệ thống chuyển sang giao diện để back officer điều chỉnh thông tin công việc. 3. Back officer nhấn hoàn thành. 4. Hệ thống trở về giao diện chính.
Postcondition	Những thay đổi công việc được lưu vào hệ thống và gửi thông tin về cho nhân viên tương ứng
Alternative Flow	Khi chọn "Assign staff" mà nhân viên đó không sẵn sàng, hệ thống sẽ trở lại và yêu cầu giao việc cho nhân viên khác.
Exception Flow	Khi chọn "Assign vehicle" mà không có xe nào sẵn sàng, hệ thống sẽ báo lỗi.

Bảng 3: Bảng mô tả use-case dạng bảng

2 Task 2: System Modelling - Mô hình hóa hệ thống

2.1 Activity Diagram của Task Assignment Module

2.1.1 Activity Diagram



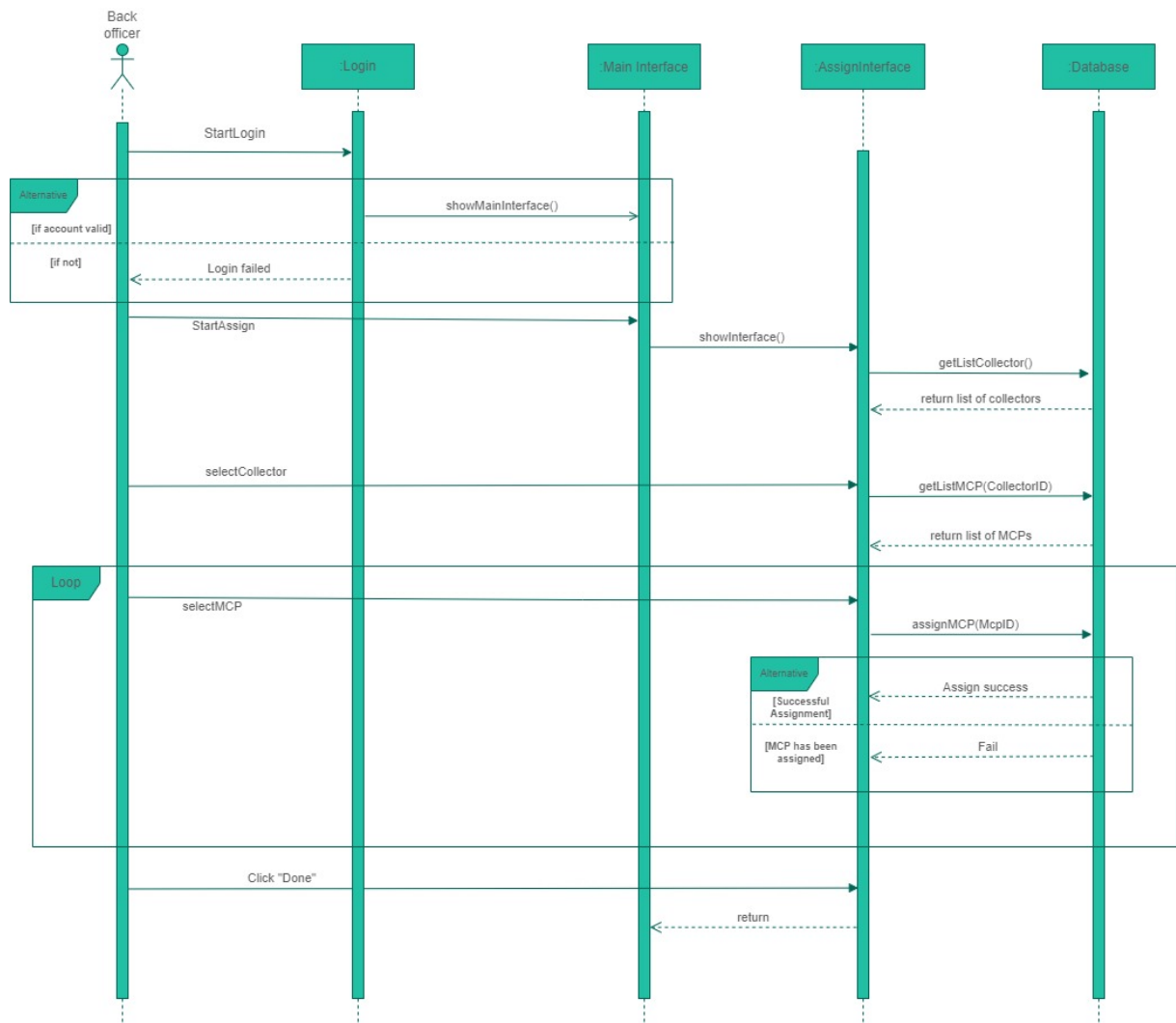
2.1.2 Mô tả

- Back Officer vào xem danh sách tổng quan.
- Sau đó, Back Officer bắt đầu phân công cho từng janitor và collector.

- Hệ thống sẽ kiểm tra và trả về những nhân viên đang “available”.
- Dựa vào danh sách nhận được từ hệ thống, họ bắt đầu phân công công việc cho từng nhân viên (thời gian, địa điểm, phương tiện,...).
- Sau khi phân công, danh sách phân công được cập nhật trên hệ thống và thông tin được gửi về cho những nhân viên liên quan.
 - + Nếu nhân viên gặp vấn đề về ca làm việc thì có thể gửi thông báo đến hệ thống.
 - + Back Officer xác nhận thì họ có thể được tinh chỉnh để làm có ca làm việc phù hợp.
- Trong trường hợp đang trong ca làm việc mà họ có gặp một số vấn đề thì gửi thông báo đến hệ thống.
 - + Hệ thống sẽ xử lý tổng quan thông tin sau đó gửi đến Back Officer.
 - + Thông tin cuối cùng được xử lý hoàn tất, cập nhật và thông báo cho các bên liên quan.

2.2 Sequence Diagram của Route Planning Task

2.2.1 Sequence Diagram



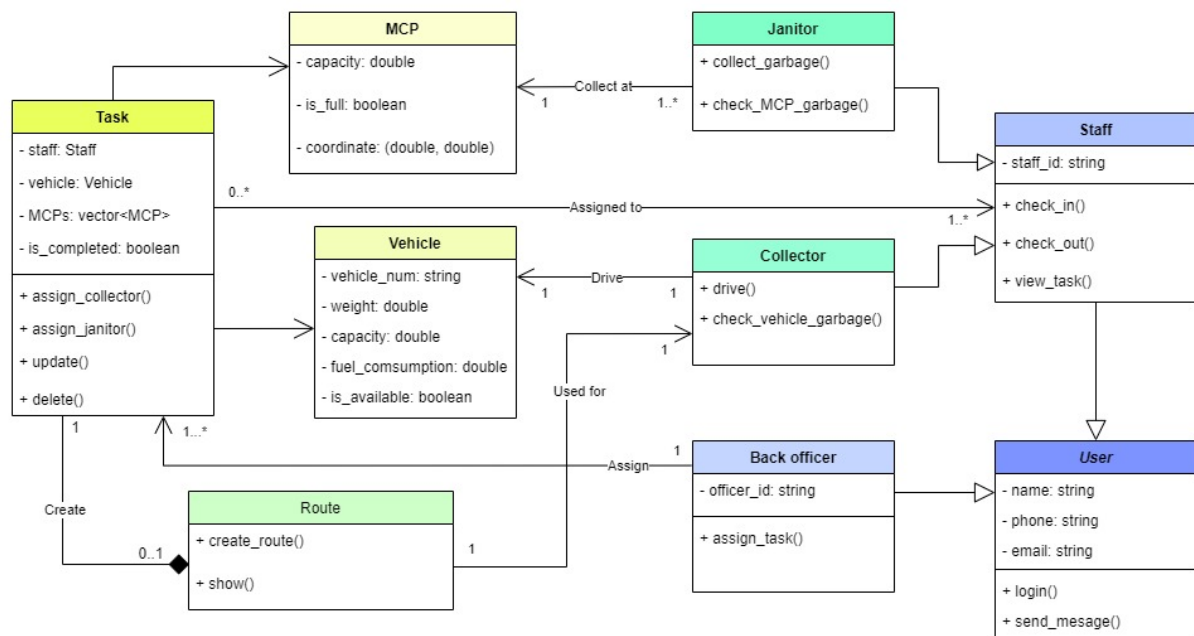
2.2.2 Mô tả

- Back officer bắt đầu đăng nhập vào hệ thống:

- + Nếu đăng nhập thành công thì hệ thống sẽ chuyển sang giao diện chính
- + Nếu không thành công, hệ thống yêu cầu người nhập đăng nhập lại
- Tại giao diện chính, back officer nhấn chọn bắt đầu phân việc và hệ thống sẽ chuyển người dùng qua giao diện phân việc và trả về danh sách các collector
- Back officer chọn collector mà mình muốn phân việc và hệ thống sẽ trả về danh sách các MCP.
- Back officer lần lượt gán các MCP theo thứ tự cho collector để hình thành tuyến đường:
 - + Nếu MCP được gán thành công, hệ thống sẽ hiện thông tin MCP lên giao diện
 - + Nếu MCP đã được gán, hệ thống sẽ báo lỗi và yêu cầu người dùng gán lại
- Sau khi đã gán xong các MCP, back officer nhấn nút “Done” và hệ thống sẽ chuyển người dùng qua giao diện phân việc

2.3 Class Diagram của Task Assignment Module

2.3.1 Class Diagram



2.3.2 Mô tả

- Back officer có thể tiến hành giao việc (Task) cho nhân viên bằng cách sử dụng method `assign_task()`. Khi đở method này sẽ tạo ra một đối tượng thuộc class Task.
- Đối tượng của Task sẽ có các thuộc tính để cung cấp thông tin cho nhiệm vụ đó, bao gồm nhân viên sẽ thực hiện Task (Task.staff), phương tiện được sử dụng trong Task (task.vehicle), các MCP sẽ được thu gom trong Task (task.MCPs) và đặc biệt là có thuộc tính để thể hiện trạng thái của Task đã được hoàn thành chưa (Task.is_completed).
- Trong class Task cũng sẽ có các method hỗ trợ cho quá trình back officer kiểm soát, phân công và chỉnh sửa nhiệm vụ.
 - + `assign_collector()`, `assign_janitor()` sẽ giúp back officer phân việc cho collector và janitor.
 - + Các method `update()`, `delete()` sẽ giúp back officer có thể điều chỉnh Task cho phù hợp với tình hình thực tế.
- Class Janitor và Collector sẽ phải có các method lần lượt là `check_MCP_garbage()`, `check_vehicle_garbage()` để theo dõi tiến độ và gửi tin nhắn cho back officer khi cần thiết.

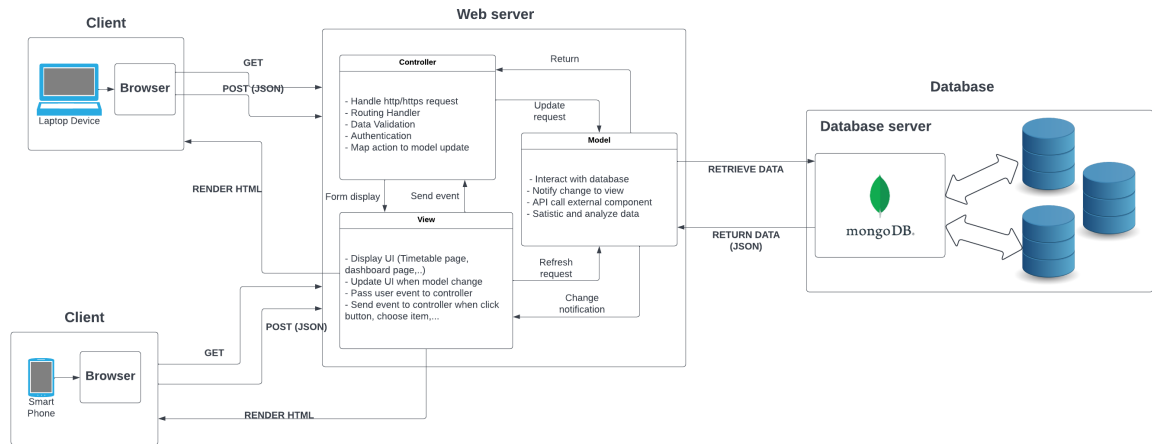


- Bên cạnh đó, class MCP và class Vehicle có các thuộc tính hỗ trợ cho bài toán định tuyến đường đi nhằm tối ưu hiệu suất làm việc.

3 Task 3: Architecture design - Thiết kế kiến trúc

3.1 Architectural approach cho mô hình MVC

3.1.1 Diagram



3.1.2 Mô tả

Mô hình MVC được chia làm 3 thành phần Model, View, Controller.

- View
 - Render các thành phần UI của từng trang (timetable, dashboard,...)
 - + Input: các file HTML, CSS, JS và các resources cần thiết để trình bày được 1 trang cụ thể.
 - + Output: Một UI thể hiện toàn bộ trang cho người dùng tương tác và làm việc
 - Cập nhật UI khi Model thay đổi
 - + Input: Thông báo thay đổi của Model
 - + Output: Một UI mới cập nhật các thành phần dữ liệu bị thay đổi trong Model
 - Gửi input (dữ liệu khi user điền form) của user cho controller
 - + Input: dữ liệu được người dùng cung cấp qua các form
 - + Output: gửi toàn vẹn phần input cho controller kiểm tra và xử lý
 - Gửi event thao tác với UI của user cho controller
 - + Input: các thao tác của user (click button, choose item, ...)
 - + Output: gửi event để thông báo cho controller biết
- Model
 - Tương tác với database
 - + Input: Các lệnh truy vấn dữ liệu phục vụ cho View
 - + Output: Dữ liệu được trích xuất từ database dưới dạng JSON.
 - Thống kê và phân tích dữ liệu
 - + Input: Các lệnh của Controller mà cần có sự phân tích từ dữ liệu
 - + Output: Các số liệu phân tích thống kê được (biểu đồ, số liệu, ...)
 - Thông báo thay đổi cho View
 - + Input: các yêu cầu thay đổi được gửi của Controller
 - + Output: Thay đổi các thành phần, thông báo tín hiệu cho View

- Controller

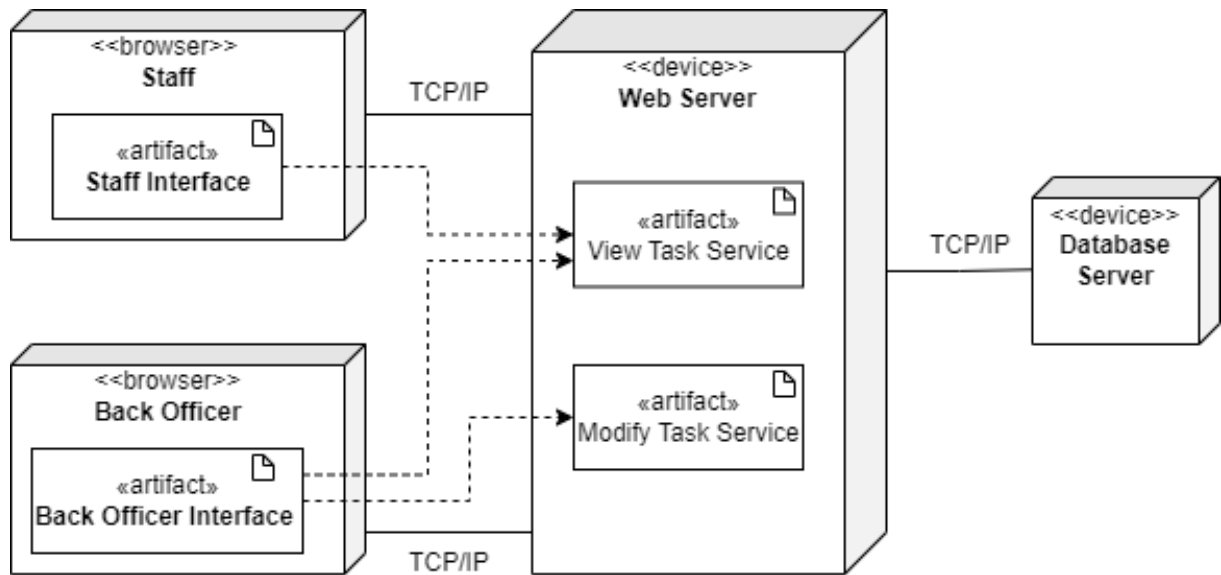
- Bắt và xử lý HTTP/HTTPS request
 - + Input: HTTP/HTTPS request
 - + Output: Gửi thông báo request cho Model thực hiện
- Xử lý các routing của website
- Xác thực dữ liệu
- Authentication
- Trả về response cho phía client với request tương ứng
 - + Input:
 - + Output: các resources cần thiết để trình bày trang web(HTML, CSS, JS, ...)

	MVC	Sever-side Rendering
Lợi ích	<ul style="list-style-type: none"> • Phát triển nhanh chóng nhờ được chia thành 3 phần. Mỗi thành phần có thể được thực hiện song song và hoàn thành riêng lẻ. Chứ không như các mô hình khác cần thực hiện tuần tự các thành phần. • Các thành phần độc lập với nhau nên việc chỉnh sửa, thay đổi, nâng cấp, bảo trì một thành phần sẽ không ảnh hưởng đến toàn bộ kiến trúc. • Dễ dàng thêm các requirement về tương tác với dữ liệu một cách độc lập với việc hiển thị ở phía người dùng. 	<ul style="list-style-type: none"> • Dễ tối ưu, vì toàn bộ dữ liệu đã được xử lý ở server, client chỉ việc hiển thị UI.
Khó khăn	<ul style="list-style-type: none"> • Tốn thời gian chuyển dữ liệu giữa các thành phần • Hiện thực code phức tạp • Cần đảm bảo tương tác hiệu quả giữa model - view - controller. 	<ul style="list-style-type: none"> • Nặng server vì server phải xử lý nhiều logic và dữ liệu • Mỗi lần người dùng chuyển trang là site phải load lại nhiều lần, gây khó chịu • Tốn băng thông vì server phải gửi nhiều dữ liệu thừa và trùng (HTML, header, footer)

Bảng 4: Bảng so sánh khó khăn và lợi ích giữa việc sử dụng MVC và Sever-side Rendering

3.2 Vẽ implementation diagram cho Task Assignment Module

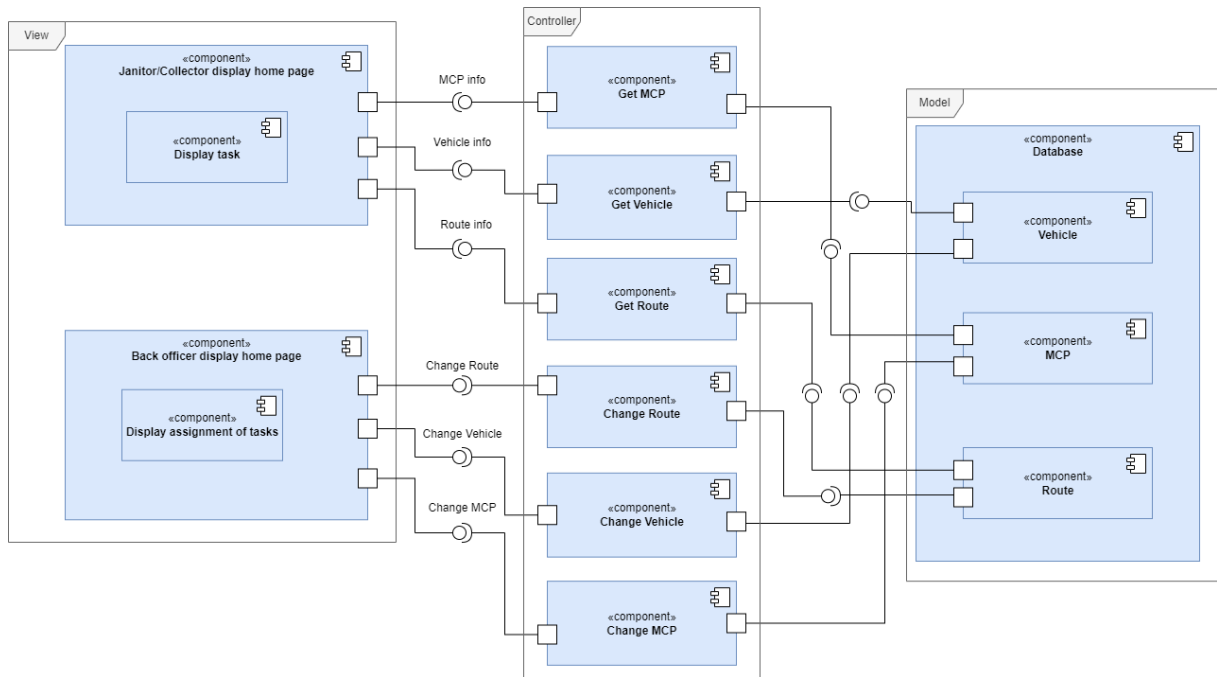
3.2.1 Deployment Diagram



Mô tả

- Nhân viên (Collector và Janitor) và back officer sẽ kết nối với Web server từ web browser qua giao thức TCP/IP.
- Server sẽ cung cấp dịch vụ và giao diện View Task cho cả nhân viên và back officer. Đối với riêng back officer, server cũng sẽ cung cấp thêm dịch vụ Modify Task để phân công cũng như chỉnh sửa công việc cho nhân viên.
- Web Server sẽ giao tiếp với Database Server qua giao thức TCP/IP. Database Server sử dụng MongoDB và chứa các dữ liệu về Route, Vehicle, MCP và nhân viên.

3.2.2 Component Diagram



Mô tả

- Nếu user là Janitor hoặc Collector, user có thể chọn xem công việc. Khi đó, thông tin về các MCP, Vehicle, Route sẽ được các component Get MCP, Get Vehicle, Get Route lấy từ trong Database và cung cấp cho giao diện qua component Display Task.
- Nếu user là Back Officer, user có thể chọn phân công công việc. Khi đó, các component Change Route, Change Vehicle, Change MCP nhận yêu cầu từ giao diện sẽ thực hiện việc thay đổi (thêm, xóa, cập nhật) các MCP, Vehicle, Route trong Database.



Tài liệu

- [1] Ian Sommerville (2016), Software Engineering (10th ed.), ISBN 978-0-137-03515-1, Addison Wesley.
- [2] S. Bennett, J. Skelton, K. Lunn (2004), UML (2nd ed.), McGraw-Hill Europe.
- [3] E.J. Braude (2001), Software Engineering: An Object-Oriented Perspective, ISBN 978-0-471-32208-5, John Wiley.
- [4] Gamma, E., Helm, R., Johnson, R., Vlissides, J., Design Patterns: Elements of Reusable Object-Oriented Software, ISBN 978- 0201633610, AddisonWesley Professional (November 10, 1994).