

HO CHI MINH CITY UNIVERSITY OF TECHNOLOGY
FACULTY OF COMPUTER SCIENCE AND ENGINEERING



SOFTWARE ENGINEERING PROJECT REPORT

Instructor: Lê Đình Thuận

Members: Hoàng Minh Hiền - 2010262
Tăng Văn Minh - 2013787
Võ Công Thành - 1814038
Nguyễn Duy Hòa - 2010276
Võ Trường Chinh - 1710678
Nguyễn Văn Bảo Nguyên - 2013930

HCMC, 09/2022



MỤC LỤC

1. Task 1: Requirements elicitation	3
1.1. Identify the context of this project. Who are relevant stakeholders? What are their current needs? What could be their current problem? In your opinion, what benefits UWC 2.0 will be for each stakeholder?	3
a. Context	3
b. Stakeholders	3
c. Stakeholders' needs and problems	4
d. Benefits	5
1.2. Describe all functional and non-functional requirements that can be inferred from the project description. Draw a use-case diagram for the whole system	6
a. Functional requirements	6
b. Non-functional requirements	6
c. Use-case diagram	7
1.3. For the Task assignment module, draw its use-case diagram and describe the use-case using a table format	8
Communication use case	8
a. Use-case diagram	8
b. Describe use-case	9
Check in and out use case	10
a. Use-case diagram	10
b. Describe use-case	11
MCP's notification use case	13
a. Use-case diagram	13
b. Describe use-case	14
View all tasks use case	15
a. Use-case diagram	15
b. Describe use-case	16
Provide overview of MCPs use case	18



a. Use-case diagram	18
b. Describe use-case	19
Manage calendar	21
a. Use-case diagram	21
b. Describe use-case	22
Manage route	23
a. Use-case diagram	23
b. Describe use-case	24
View calendar	25
a. Use-case diagram	25
b. Describe use-case	26
View detailed tasks	27
a. Use-case diagram	27
b. Describe use-case	28
2. Task 2	29
2.1. Draw an activity diagram to capture the business process between systems and the stakeholders in Task Assignment module	29
2.2. Proposal a conceptual solution for the route planning task and draw a sequence diagram to illustrate it.	30
a. General Diagram	30
b. Create route diagram	31
c. Delete route diagram	32
d. Modify route diagram	33
e. Login diagram	34
2.3 Draw a class diagram of Task Assignment module as comprehensive as possible	35
a. Routes and Vehicles Planning	35
b. Collectors and Janitors Assignment	36



3.1 Describe an architectural approach you will use to implement the desired system. How many modules you plan for the whole WMC 2.0 system? Briefly describe input, output and function of each module	37
a. Architectural approach	37
c. Input, output and function of each module	38
3.2 Draw an implement diagram for Task Assignment module	40

1. Task 1: Requirements elicitation

1.1. Identify the context of this project. Who are relevant stakeholders? What are their current needs? What could be their current problem? In your opinion, what benefits UWC 2.0 will be for each stakeholder?

a. Context

Dense population also means an insurmountable amount of urban waste. To deal with this problem, many countries agree to a goal called Sustainable Development Goal (SDG) which requires sustainable cities and communities and water and sanitation.

Current typical waste collection process here in Vietnam is rather obsolete and inefficient. So we decide to turn it up a notch by replacing traditional collector only model with back officer-collector-janitor for efficiency called UWC 2.0

Instead of moving around the city, from now on, janitors will take responsibility for gathering the garbage in a small neighborhood and bring it manually to the major collecting points. Collectors will have a specific route to come through the major collecting points and pick up the garbage. This is similar to public transportation but we will just need a few cars every day. UWC 2.0 is a better solution, however, we will mainly focus on the communication between back officers, collectors and janitors and some functions that we will show you below. We cannot make any adjustments about the capacity of collecting points.

b. Stakeholders

- + Janitors
- + Collectors
- + Back officers
- + Organization X
- + Service provider Y



Ho Chi Minh City University of Technology
Faculty of Computer Science and Engineering



c. Stakeholders' needs and problems

Current needs:

+ Back officers:

- create calendar
- coordinate front collectors and janitors
- create routes through MCPs
- know information about teams of janitors and collectors
- choose vehicle and assign routes to them (every month)
- sent messages with information about collecting route and plan to collectors and janitors (everyday)
- manage team's schedule

+ Collectors:

- know about which vehicle to use
- know about the collecting route and time
- know about calendars
- receive messages from back officers
- need best route, best way of waste collecting to save time and resource

+ Janitors:

- trollers
- assign tasks and calendar (weekly)
- receive messages from back officers
- minimize workload and time by arrange tasks efficient

+ Organization X:

- use resources efficiently to develop interoperable UWC 2.0 from UWC 1.0 which satisfies the contract's requirements.
- need service provider Y 's resource (current system with database, technique, document).



+ Service provider Y

- organizes X 's resources (labor, technique).
- an efficient management and arranging waste collection based on technology and smart sensors in order to cut expenses and reduce time.

Current problems:

- Poor time management, heavy but inefficient workload for the collector. Not optimized way for communicating
- Obsolete 1.0 system
- High cost for maintenance
- Back officers need sufficient technical skills
- More money to pay for employee
- Vietnamese system interfaces and Eng-Vi capability in the future
- Larger scale in collecting site also means larger system
- Task Management 's expected to be interoperable with UWC 1.0
- System should handle real-time data from at least 1000 MCPs now and 10 000 MCPs in 5 years.

d. Benefits

In my opinion, UWC 2.0 brings different benefits to different stakeholders. As mentioned before, back officers will have a good platform that provides everything they need for management which provides higher performance, instant feedback and easy to communicate with others. From the perspective of collectors and janitors, they can save time and reduce a lot of work that needs to be done.

In the point of view of the government, this project brings a lot of benefits. The most clear to see is the cost reduction as we can reduce many many trucks. And next is that the improvement in management structure, more data about the situation of the city will be sent back to the database of our government and it will improve the life quality of our citizens.

UWC 2.0 is the best solution for collecting garbage that will convert the collecting garbage industry from traditional to digital. This can be a big step to bring our country nearer to a digital world.



1.2. Describe all functional and non-functional requirements that can be inferred from the project description. Draw a use-case diagram for the whole system

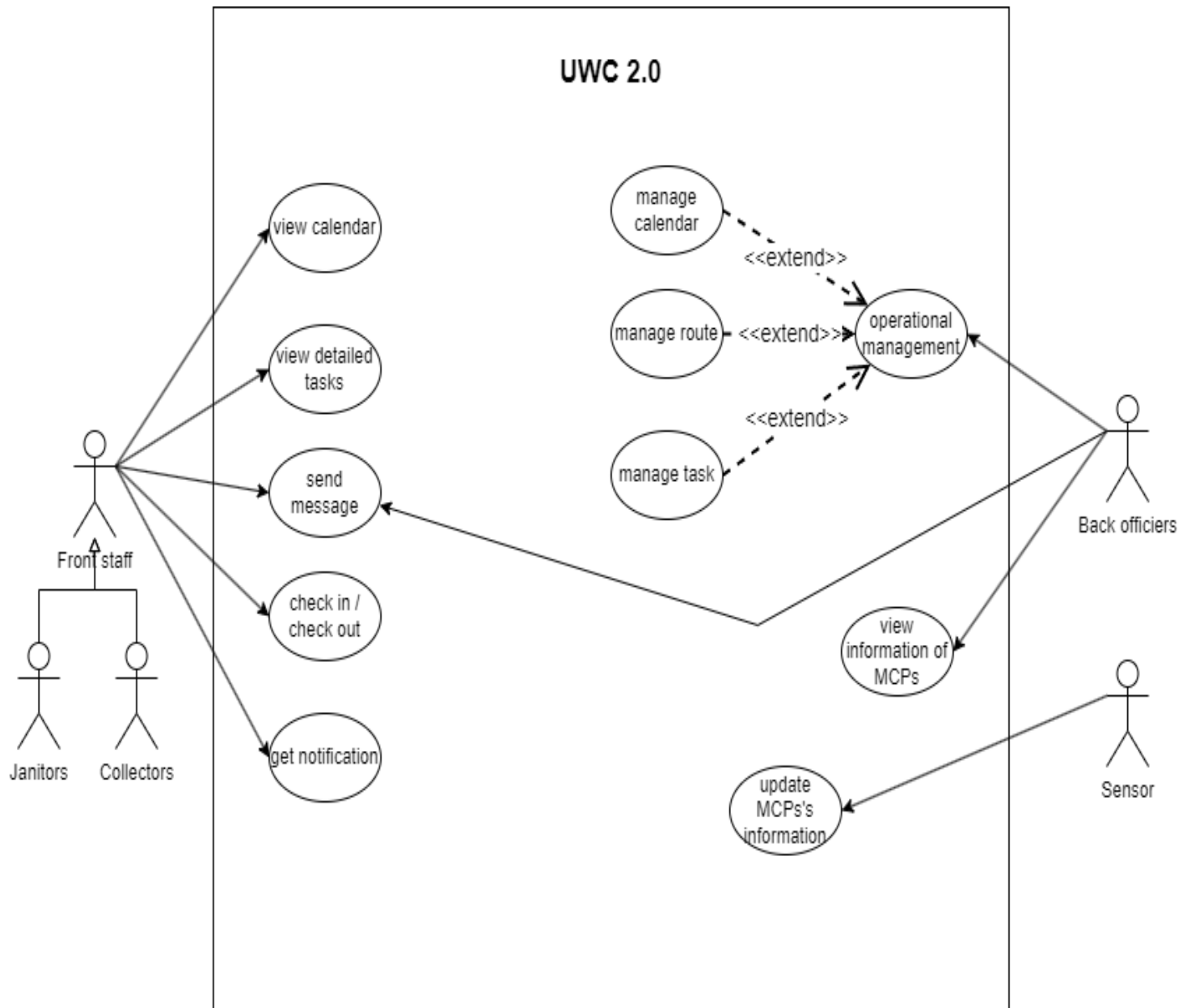
a. Functional requirements

- + Back officers:
 - Be able to operate a central system to **create calendar**
 - Have possibility to **create** and **manage (assign) tasks** among the team of janitors
 - Be able to **plan vehicles** to use for collectors
 - **Create routes** (and **optimize**) for each vehicle
 - **Send messages** (with information about route and time) to staff
- + Front staff (Collectors and Janitors)
 - **View** their **calendar**
 - **Get** information about the **vehicle** to use
 - **Get** information about the **tasks**
 - **Receive messages** (with information about route and time) from Back officers

b. Non-functional requirements

- + Usability:
 - All important information should be displayed in one view (without scrolling down).
 - System interfaces should be in **Vietnamese**, with an opportunity to switch to **English** in the future.
- + Reliability:
 - MCPs' information should be updated from MCPs every 15 minutes with the availability of at least 95% of their operating time.
- + Compatibility:
 - **Import** and **use** the existing data from UWC 1.0. Inter-operable with UWC 1.0
- + Performance:
 - Handle real-time data from at least **1000 MCPs** at the moment
 - Messages sent between members should be in a real-time manner with delay **less than 1 seconds**
- + Scalability:
 - The system must be scalable enough to support 10000 MCPs in 5 years while maintaining optimal performance.

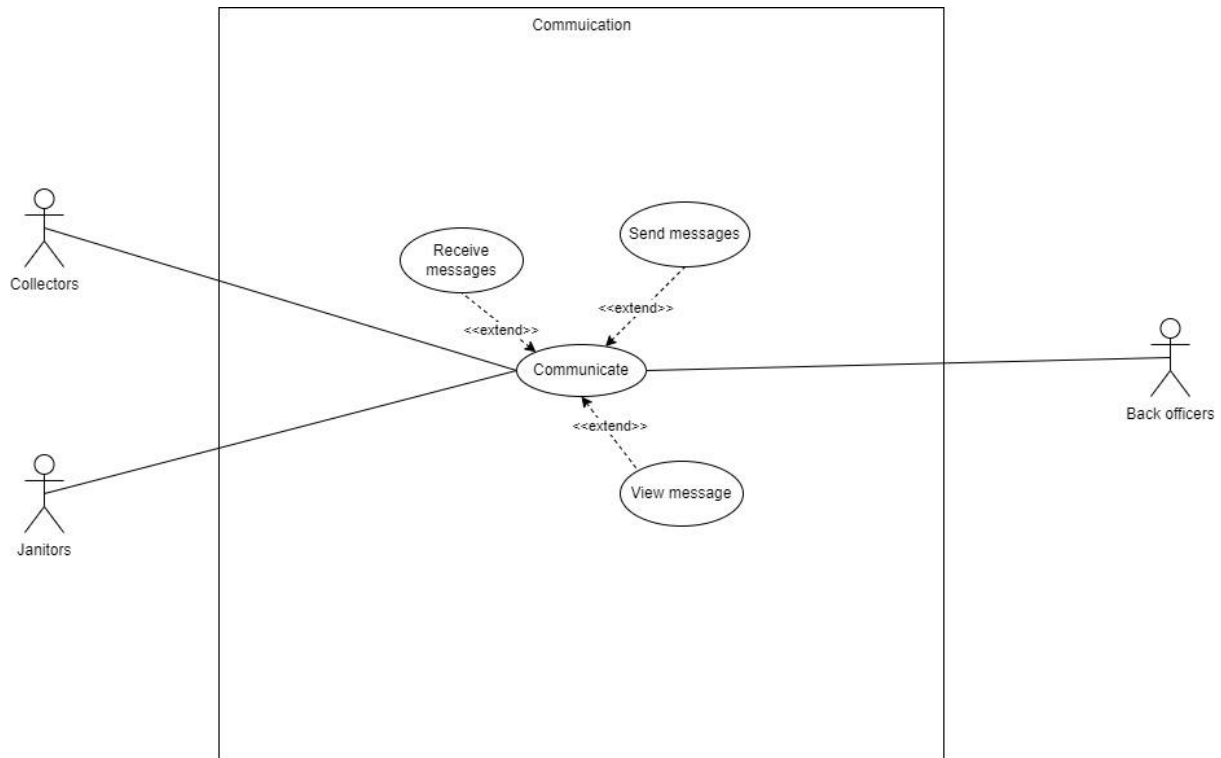
c. Use-case diagram



1.3. For the Task assignment module, draw its use-case diagram and describe the use-case using a table format

Communication use case

a. Use-case diagram



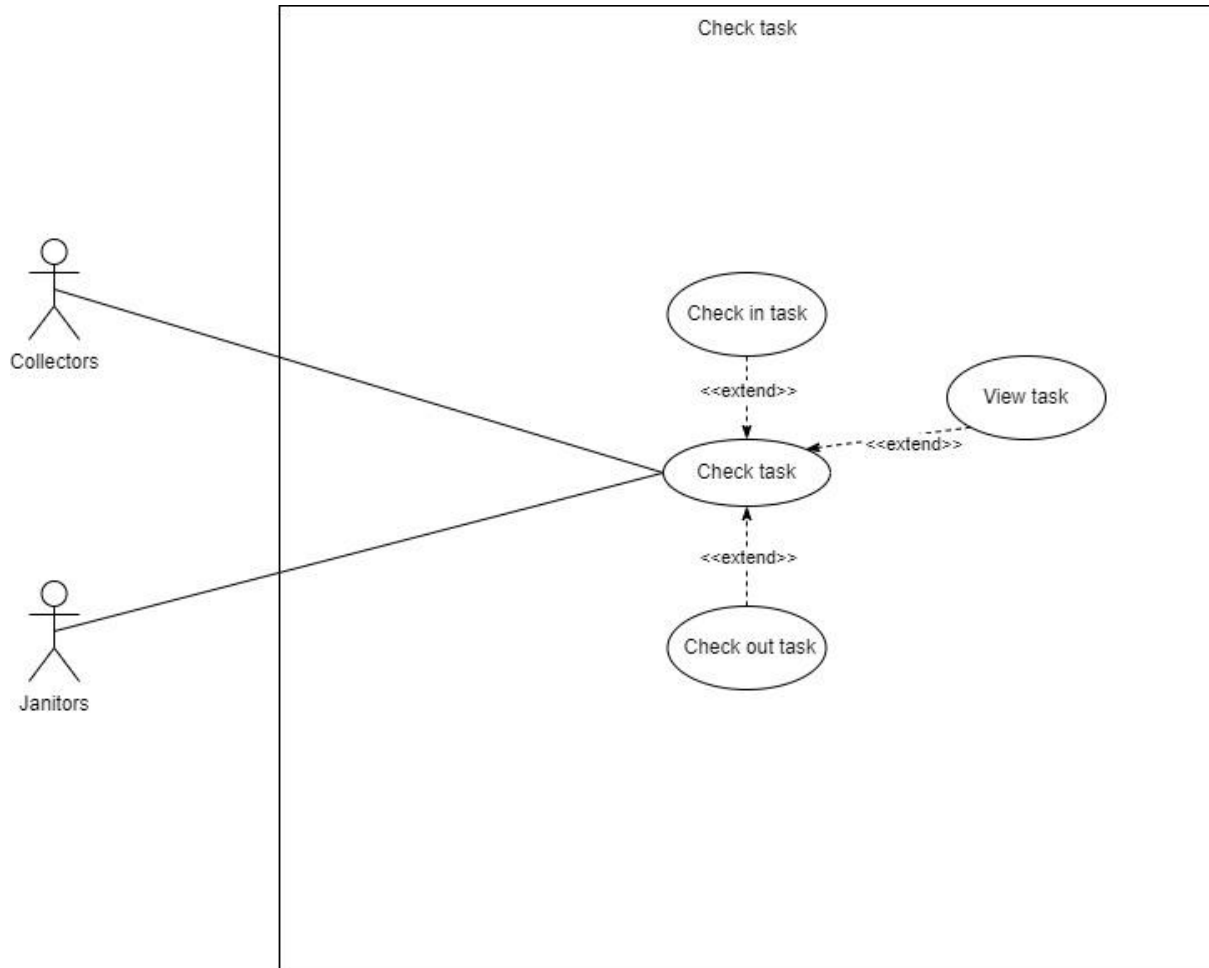
b. Describe use-case

Table 1. Communication use case

Use-case	Communication		
Created by	Vo Cong Thanh	Last updated by:	Vo Cong Thanh
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Collectors, janitors, back officers		
Description	Communicating between collectors, other janitors and back officers		
Trigger	Actors communicate with each other in the form of messages		
Precondition	1/ Have an internet connection to access the system. 2/ Actors have logged into the system. 3/ The communication system must be installed before		
Post-condition	1/ Message has been sent, received, viewed successfully		
Normal Flow	1/ Actors move to communication section 2/ Perform the operation of sending and receiving messages to each other 3/ View messages that have been sent from others 4/ The communication process is complete		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 1s.		

Check in and out use case

a. Use-case diagram





b. Describe use-case

Table 2. Check in use case

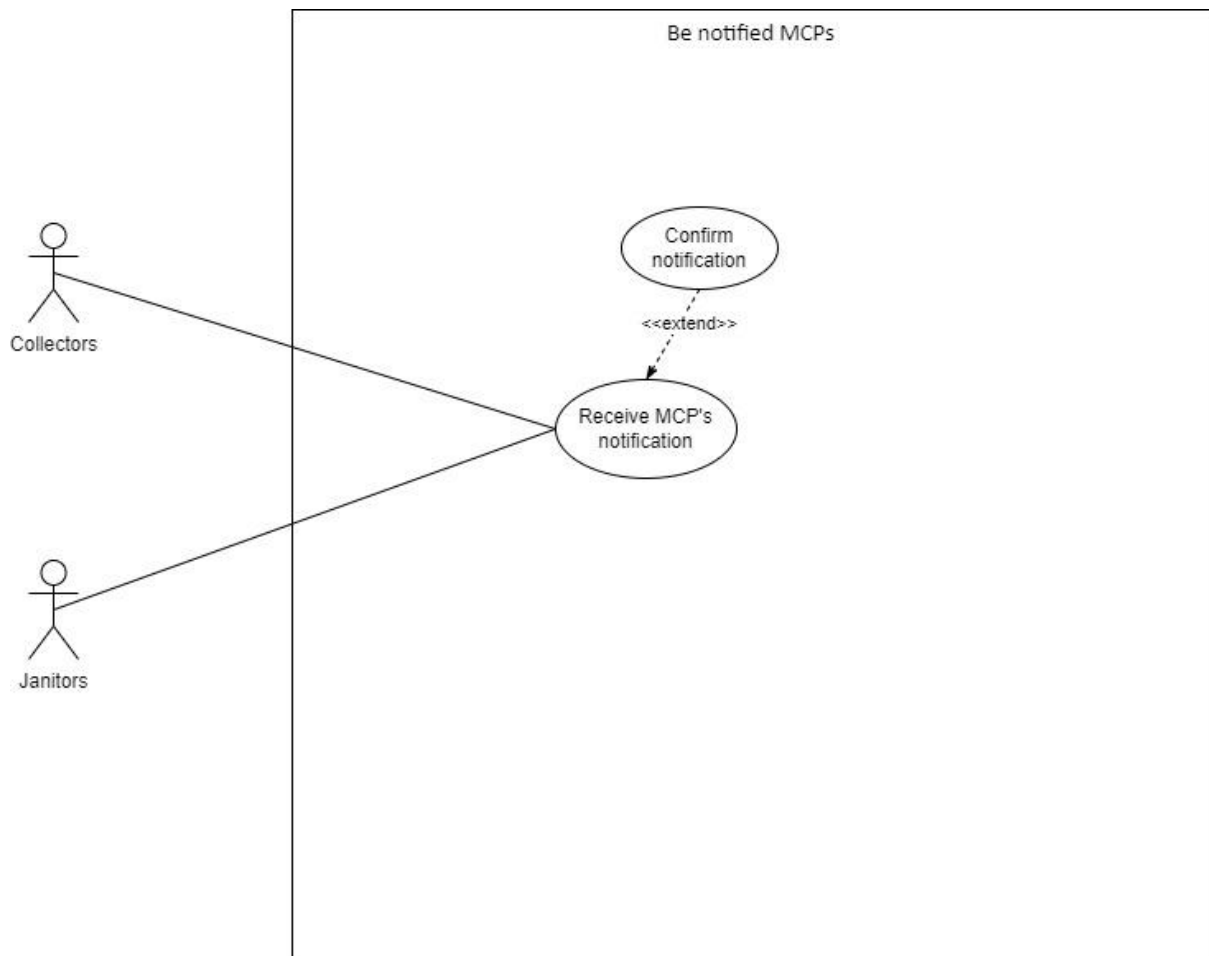
Use-case	Check in task everyday		
Created by	Vo Cong Thanh	Last updated by:	Vo Cong Thanh
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Collectors, janitors		
Description	Check in tasks before starting work every day		
Trigger	Actors perform check in task on system		
Pre-condition	1/ Have an internet connection to access the system. 2/ Actor has logged into the system. 3/ Actor has been assigned the job		
Post-condition	1/ The check in task has been updated successfully		
Normal Flow	1/ Actor chooses item to check in daily work 2/ Perform view, check-in task 3/ Update the work status on the system 4/ The update process is complete.		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 3s.		

Table 3. Check out use case

Use-case	Check out task everyday		
Created by	Vo Cong Thanh	Last updated by:	Vo Cong Thanh
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Collectors, janitors		
Description	Check out tasks after finishing work every day		
Trigger	Actors perform check out task on system		
Pre-condition	1/ Have an internet connection to access the system. 2/ Actor has logged into the system. 3/ Working time has ended		
Post-condition	1/ The check out task has been updated successfully		
Normal Flow	1/ Actor chooses item to check out daily work 2/ Perform view, check-out task 3/ Update the finished work status on the system 4/ The update process is complete.		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 3s.		

MCP's notification use case

a. Use-case diagram





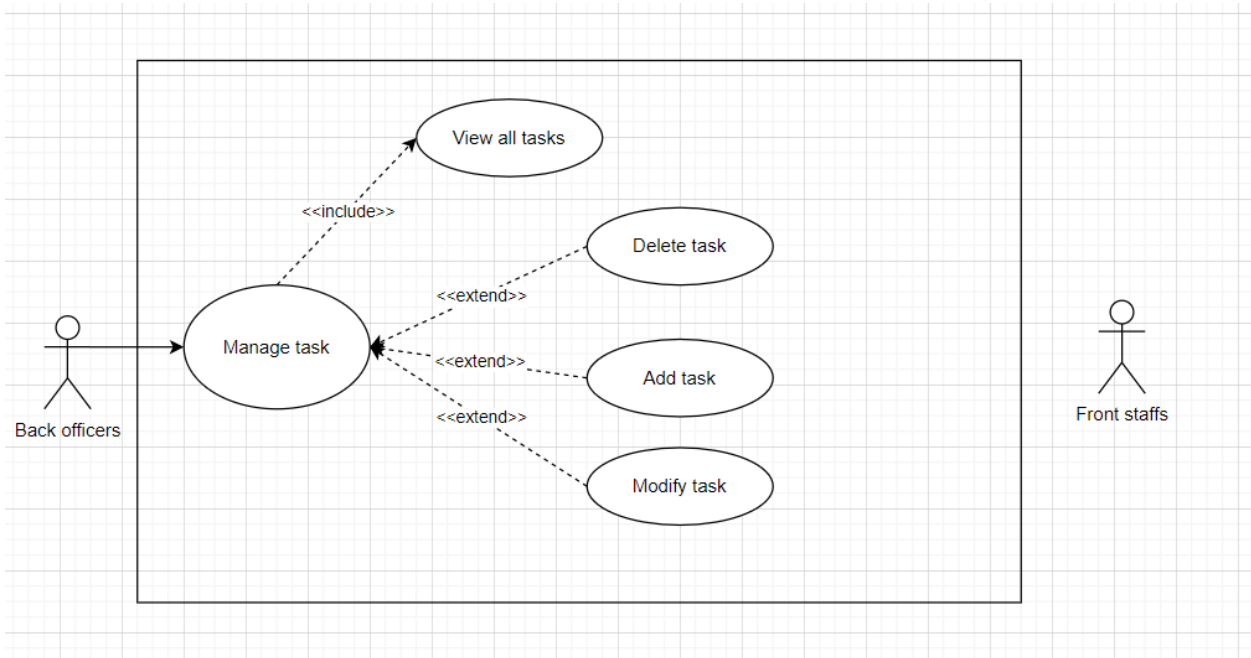
b. Describe use-case

Table 4. MCP's notification use case

Use-case	Receive MCP's notification		
Created by	Vo Cong Thanh	Last updated by:	Vo Cong Thanh
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Collectors, janitors		
Description	Actor receives notification of MCP when they are fully loaded		
Trigger	Be notified about the MCPs		
Pre-condition	1/ Have an internet connection to access the system. 2/ Manager has logged into the system. 3/ Status MCP is fully loaded		
Post-condition	1/ MCP's notification has been received successfully		
Normal Flow	1/ Actors receive notifications from the system 2/ MCP's notification has been received 3/ The update process is complete.		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 3s.		

View all tasks use case

a. Use-case diagram





b. Describe use-case

Table 5. View all tasks use case

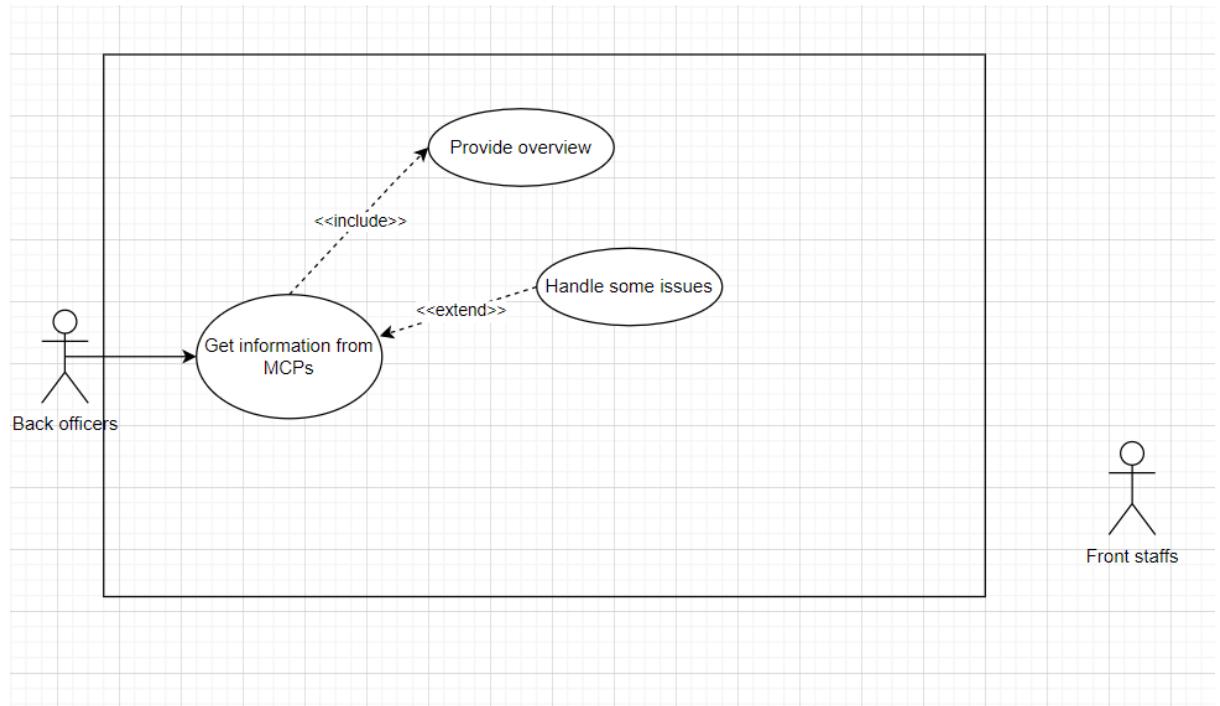
Use-case	View all tasks		
Created by	Nguyen Bao Nguyen	Last updated by:	Nguyen Bao Nguyen
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Back officers		
Description	Use-case allow Back Officers view all tasks		
Trigger	Back Officers want to view all tasks		
Pre-condition	1/ Have an internet connection to access the system. 2/ Back Officers have logged into the system.		
Post-condition	Back Officers can see all tasks		
Normal Flow	1/ Back Officers click into Task Manager 2/ Task Manager displays all the tasks 3/ Back Officers can click to each task to check task status.		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 1s.		

Table 6. Add tasks use case

Use-case	Add task		
Created by	Nguyen Bao Nguyen	Last updated by:	Nguyen Bao Nguyen
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Back officers		
Description	Use-case allow Back Officers add tasks		
Trigger	Back Officers want to add tasks		
Pre-condition	1/ Have an internet connection to access the system. 2/ Back Officers have logged into the system.		
Post-condition	Back Officers can add tasks		
Normal Flow	1/ Back Officers click into Task Manager 2/ Back Officers click into the Add Task button . 3/ Add information about tasks to create a new task. 4/ Click the Save button to save the task.		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 1s.		

Provide overview of MCPs use case

a. Use-case diagram





b. Describe use-case

Table 7. Provide overview of MCPs use case

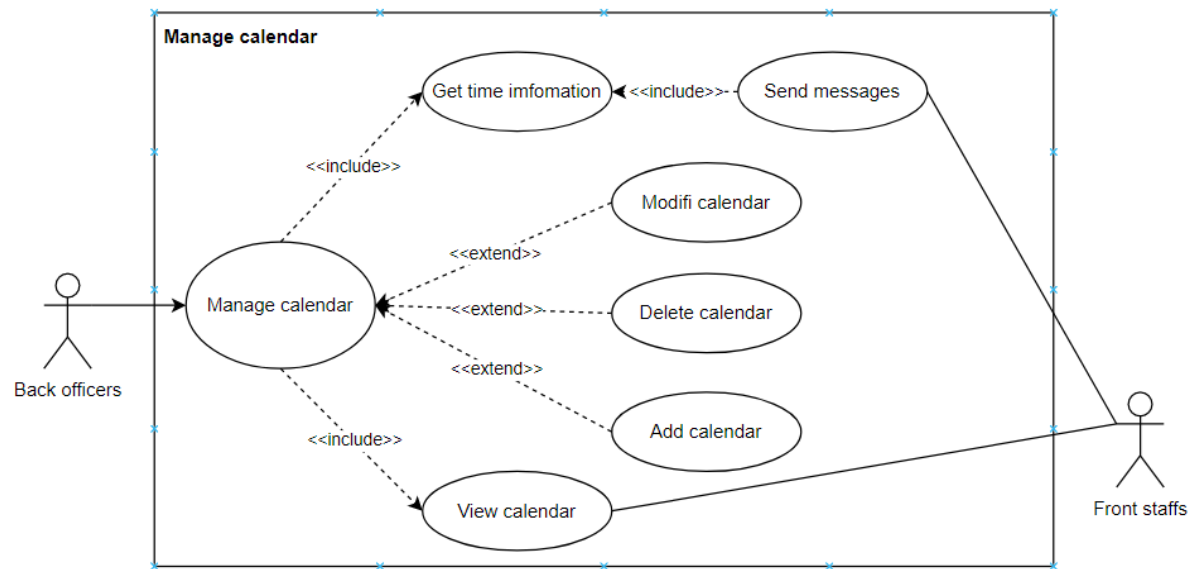
Use-case	Provide overview of MCPs		
Created by	Nguyen Bao Nguyen	Last updated by:	Nguyen Bao Nguyen
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Back officers		
Description	Use-case allow Back Officers see overview of MCPs		
Trigger	Back Officers want to see overview of MCPs		
Pre-condition	1/ Have an internet connection to access the system. 2/ Back Officers have logged into the system.		
Post-condition	Back Officers can see overview of MCPs		
Normal Flow	1/ Back Officers click into MCPs information 2/ Back Officers click into the Overview button		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 1s.		

Table 7. Provide method to handle arising issues

Use-case	Provide method to handle arising issues		
Created by	Nguyen Bao Nguyen	Last updated by:	Nguyen Bao Nguyen
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Back officers		
Description	Use-case allow Back Officers handle arising issues		
Trigger	Back Officers want to handle arising issues		
Pre-condition	1/ Have an internet connection to access the system. 2/ Back Officers have logged into the system. 3/ Have some problems with the system.		
Post-condition	Help the system run smoothly		
Normal Flow	1/ Back Officers click into MCPs information. 2/ Back Officers click into the Modify MCPs By Hand button. 3/ Modify data of MCP to fix the error. 4/ Confirm the work and save.		
Alternative	No		
Exception	No		
Note and issues	No		
Non-funtional	The delay is less than 1s.		

Manage calendar

a. Use-case diagram



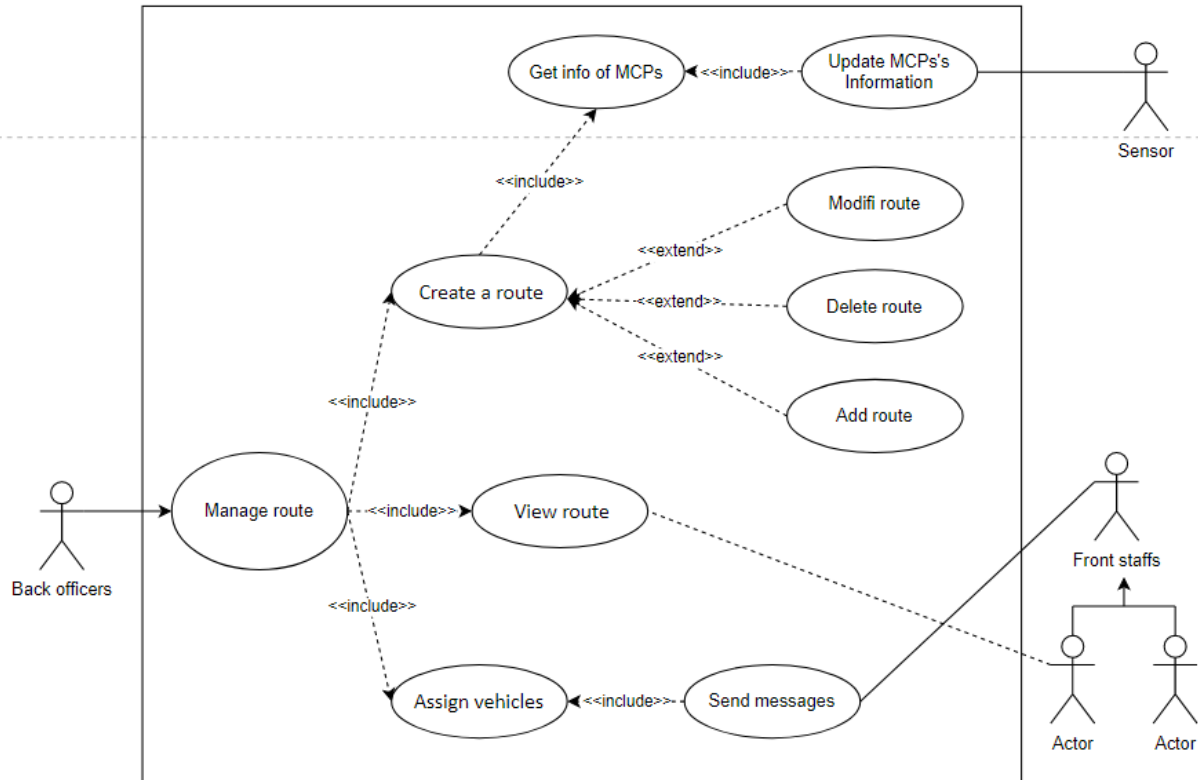
b. Describe use-case

Table 8. Manage calendar use case

Use-case Name	Manage Calendar		
Created by	Vo Truong Chinh	Last updated by:	Vo Truong Chinh
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Back Officers, Front Staffs		
Description	The Back Officers contact to the Front Staffs to know their available time, then create a calendar.		
Trigger	Actors communicate with each other in the form of messages		
Pre-conditions	Front Staffs and Back officers must login into the system using valid authorised account for customers management		
Post-conditions	The manage calendar task has been updated successfully		
Normal Flow	<ol style="list-style-type: none"> 1. System represents interface and 5 options to Back officer: <ul style="list-style-type: none"> + Get time information + Modify calendar + Delete calendar + Add calendar + View calendar 2. Back officer choose 1 in 5 options 3. The system shifts interface corresponding to the option selected 4. Back officer sequentially process and finish the options selected 5. System returns to the main interface 		
Alternative	No		
Exceptions	No		
Note and issue	No		
Non-functional	The delay less than 3s		

Manage route

a. Use-case diagram



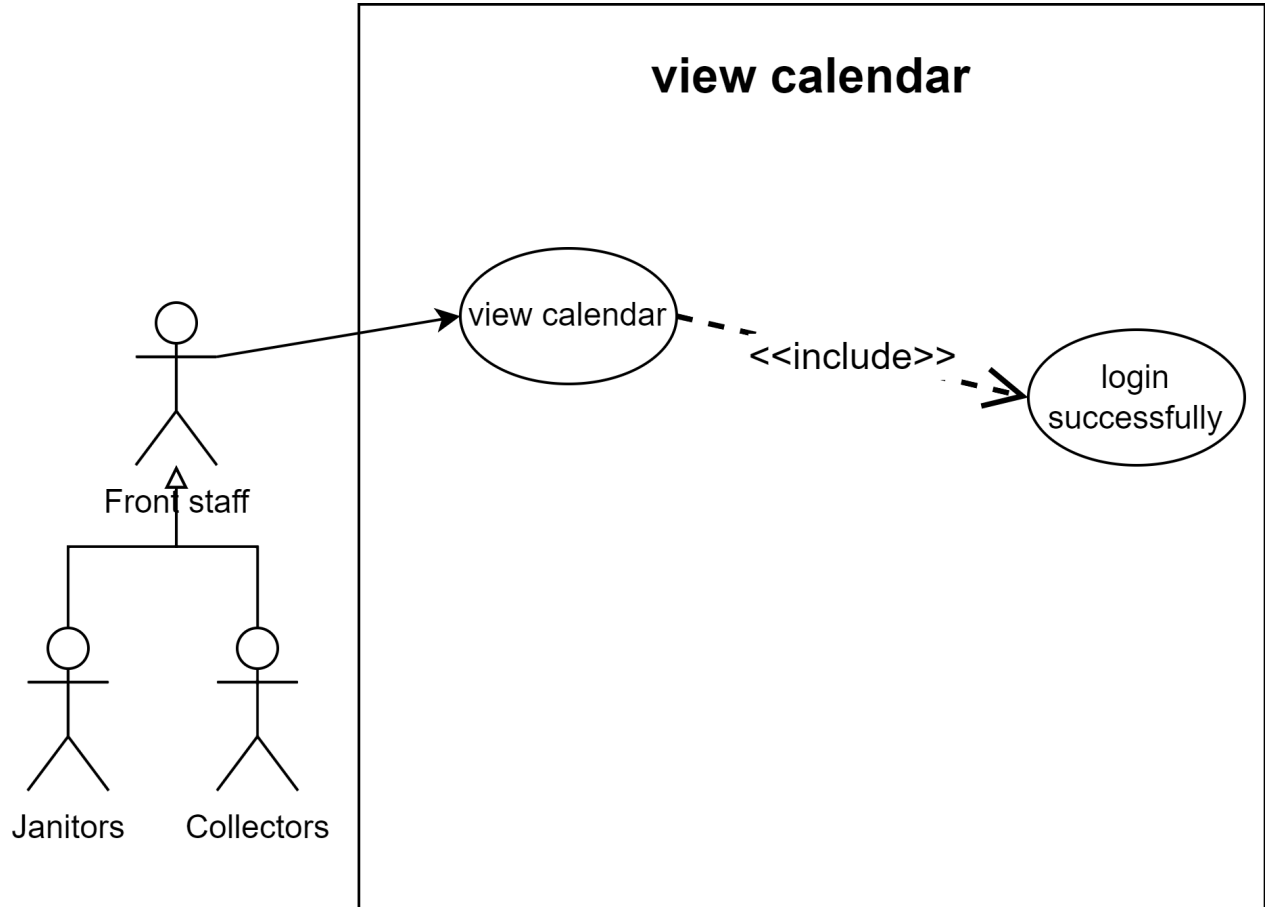
b. Describe use-case

Table 9. Manage route use case

Use-case Name	Manage Route		
Created by	Vo Truong Chinh	Last updated by:	Vo Truong Chinh
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Back Officers, Sensor, Front Staffs		
Description	The Back Officers get information of MCPs from Sensor, then create route with the best effective manually. After that give the route and assign vehicles to janitors and collectors by message		
Trigger	Actors communicate with each other in the form of messages		
Pre-conditions	Front Staffs and Back officers must login into the system using valid authorized account for customers management		
Post-conditions	The manage route task has been updated successfully		
Normal Flow	<ol style="list-style-type: none"> 1. System represents interface and 3 options to Back officer: <ul style="list-style-type: none"> + Create route + View route + Assign vehicles 2. Back officer choose 1 in 3 options 3. The system shifts interface corresponding to the option selected 4. Back officer sequentially process and finish the options selected 5. System returns to the main interface 		
Alternative	No		
Exceptions	No		
Note and issue	No		
Non-funtional	The delay less than 3s		

View calendar

a. Use-case diagram





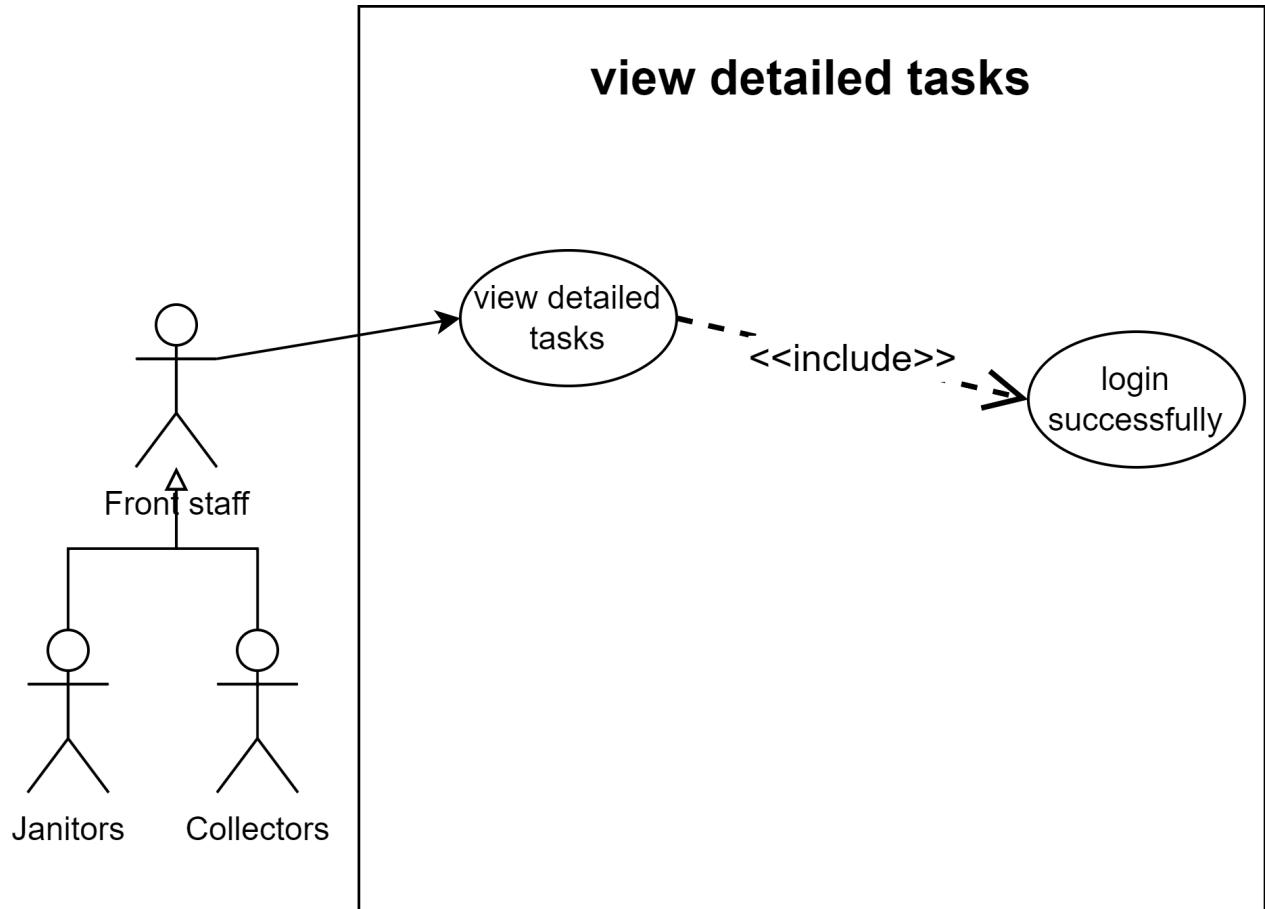
b. Describe use-case

Table 10. View calendar use case

Use-case	View calendar		
Created by	Hoàng Minh Hiền	Last updated by:	Hoàng Minh Hiền
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Collectors, janitors		
Description	Have an overview of calendar		
Trigger	Actors select “View calendar” on system		
Pre-condition	1, Have an internet connection to access the system. 2, Actor has logged into the system.		
Post-condition	System perform actor’s calendar		
Normal Flow	1, System displays the actor’s main screen. 2, Actor select view calendar on system 3, System display actor’s calendar		
Alternative	No		
Exception	3a, System notify error due to maintain system or update database <i>Use case return step 1</i>		
Note and issues	No		
Non-funtional			

View detailed tasks

a. Use-case diagram



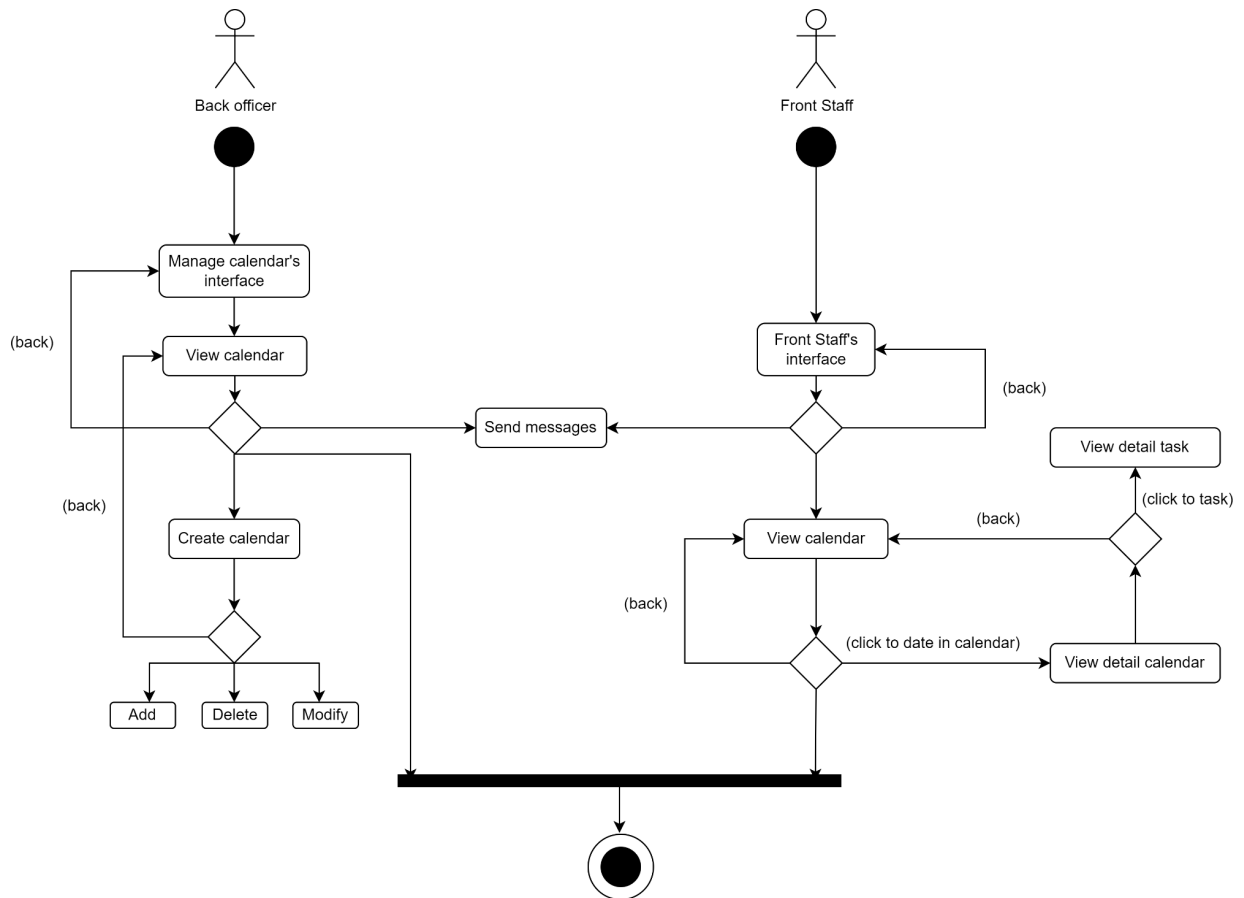
b. Describe use-case

Table 11. View detailed tasks use case

Use-case	View detailed tasks		
Created by	Hoàng Minh Hiền	Last updated by:	Hoàng Minh Hiền
Day created	25/09/2022	Date last updated:	25/09/2022
Actor	Collectors, janitors		
Description	Have an overview of detailed task (vehicles, technical details)		
Trigger	Actors select “View detailed tasks” on system		
Pre-condition	1, Have an internet connection to access the system. 2, Actor has logged into the system.		
Post-condition	System perform actor’s calendar		
Normal Flow	1, System display actor’s main screen. 2, Actor select view detailed tasks on system 3, System display actor’s detailed tasks on a daily and weekly basic		
Alternative	No		
Exception	3a, System notify error due to maintain system or update database <i>Use case return step 1</i>		
Note and issues	No		
Non-funtional	All important information should be displayed in one view (without scrolling down)		

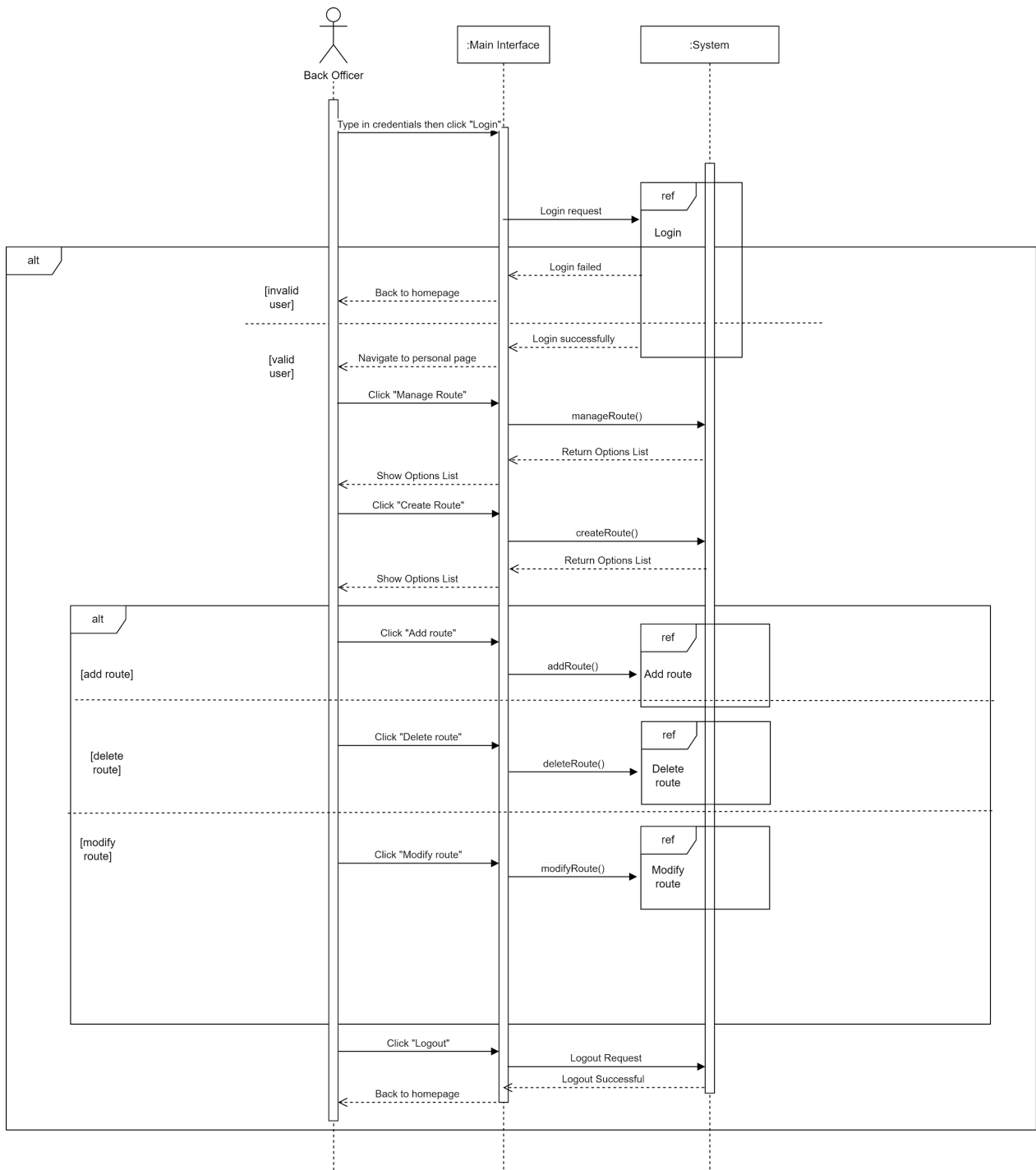
2. Task 2

2.1. Draw an activity diagram to capture the business process between systems and the stakeholders in Task Assignment module

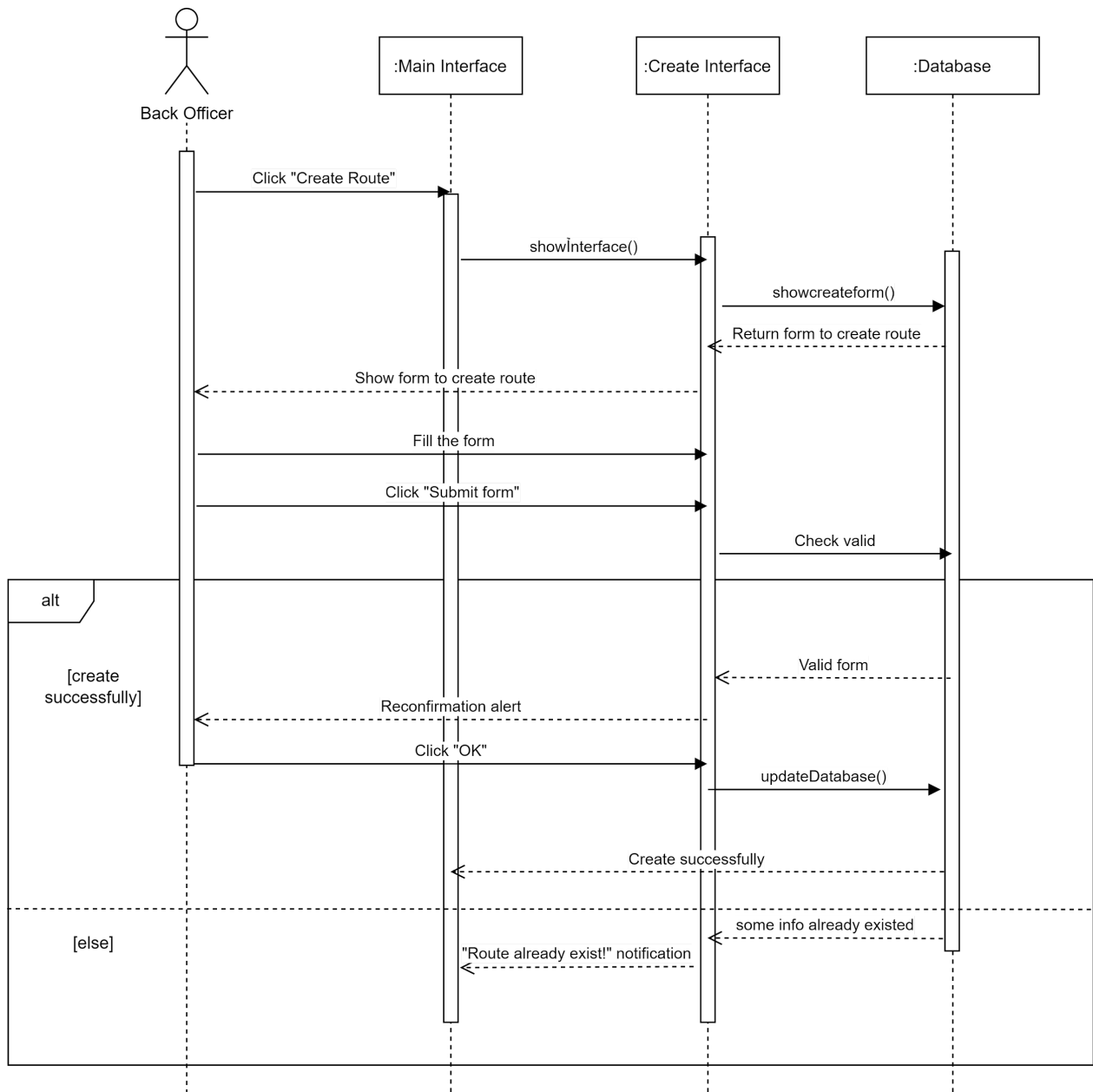


2.2. Proposal a conceptual solution for the route planning task and draw a sequence diagram to illustrate it.

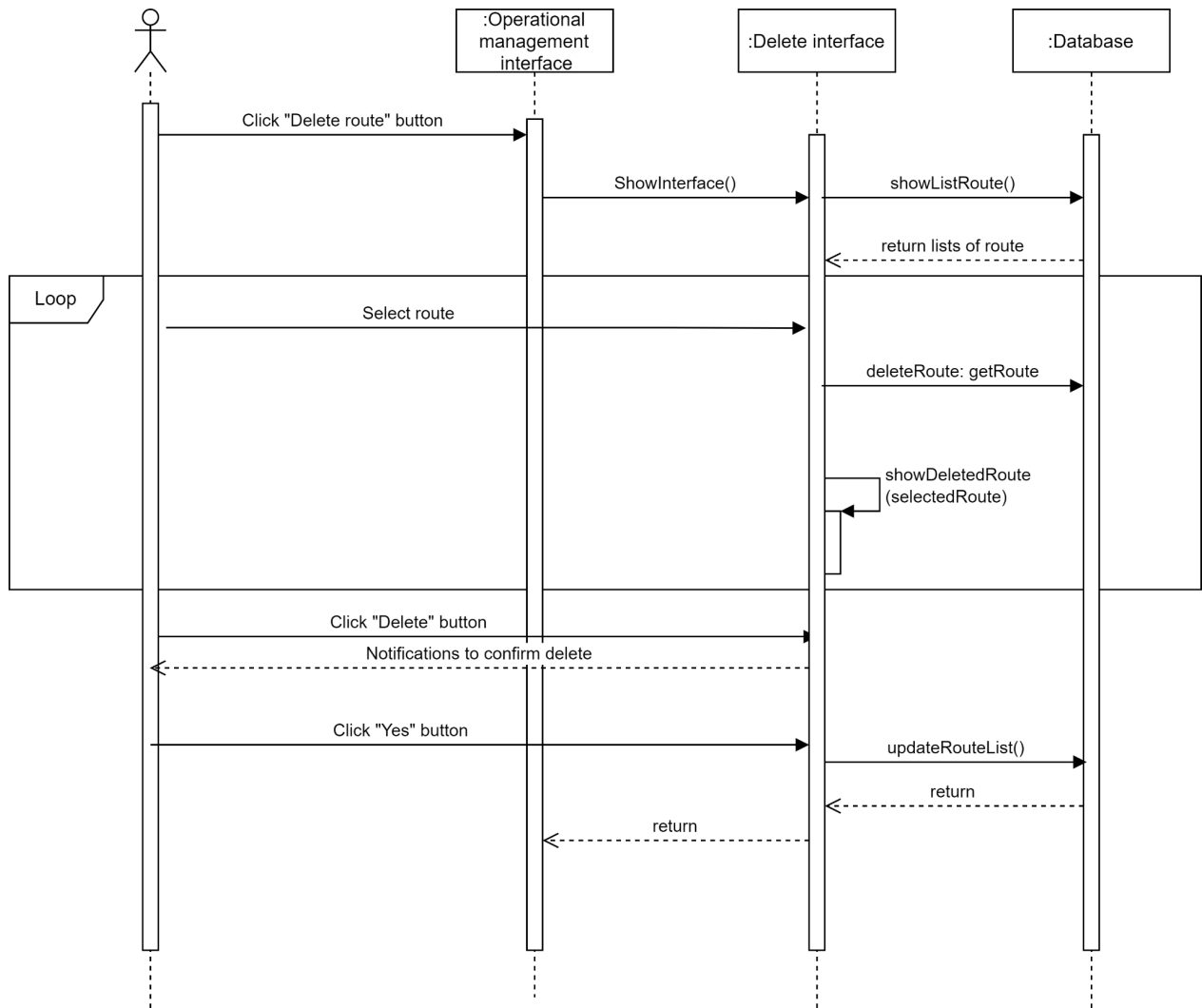
a. General Diagram



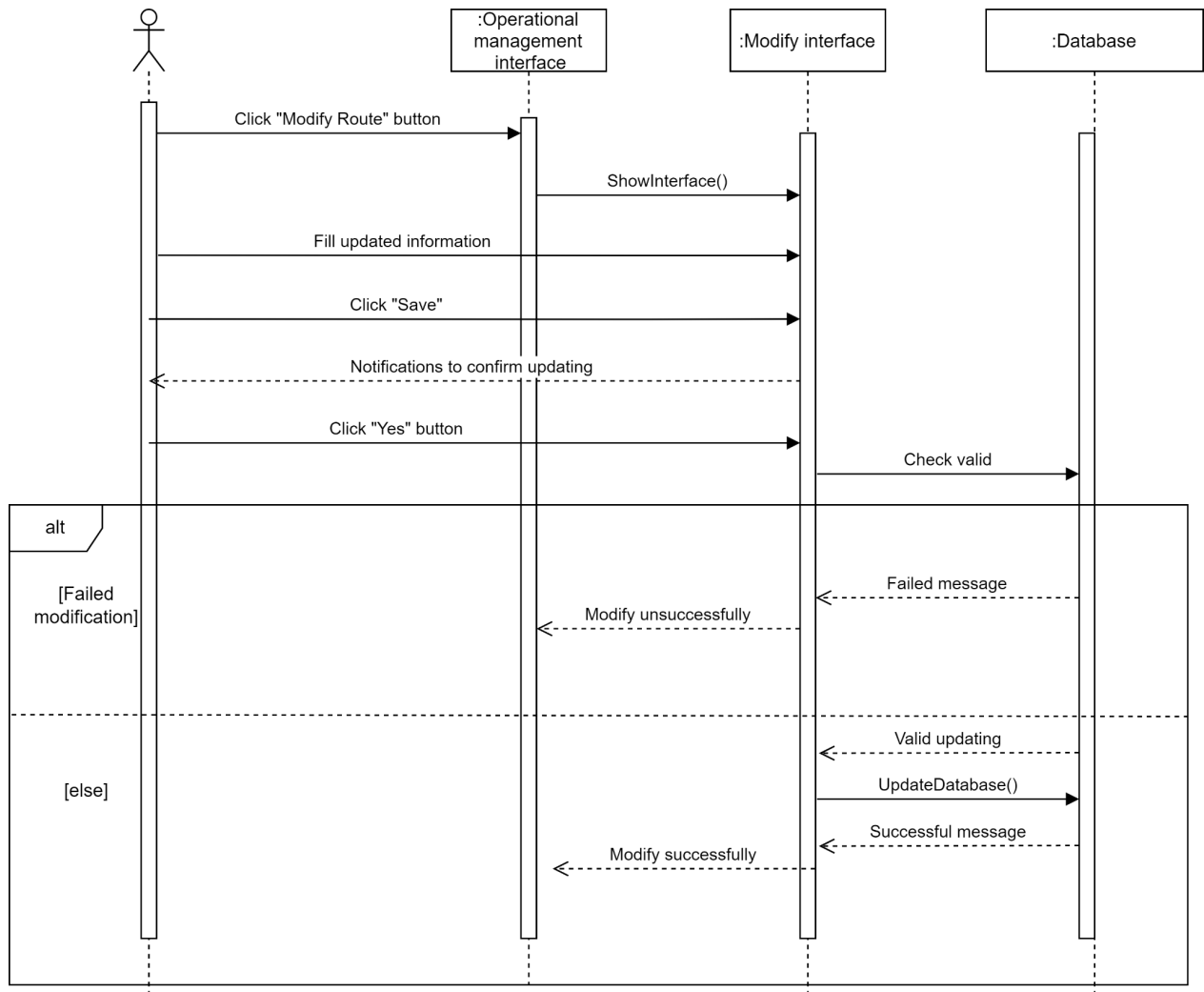
b. Create route diagram



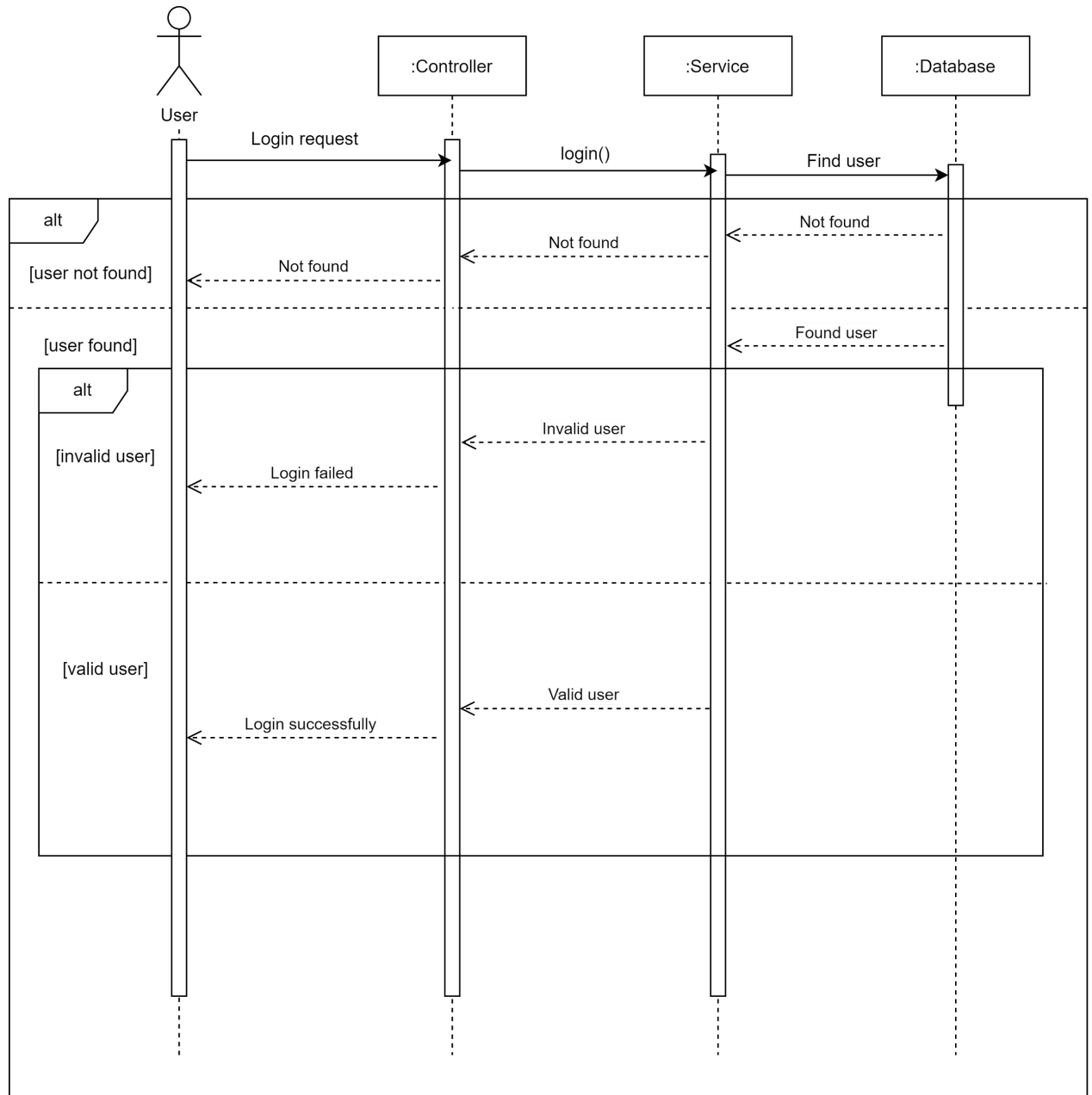
c. Delete route diagram



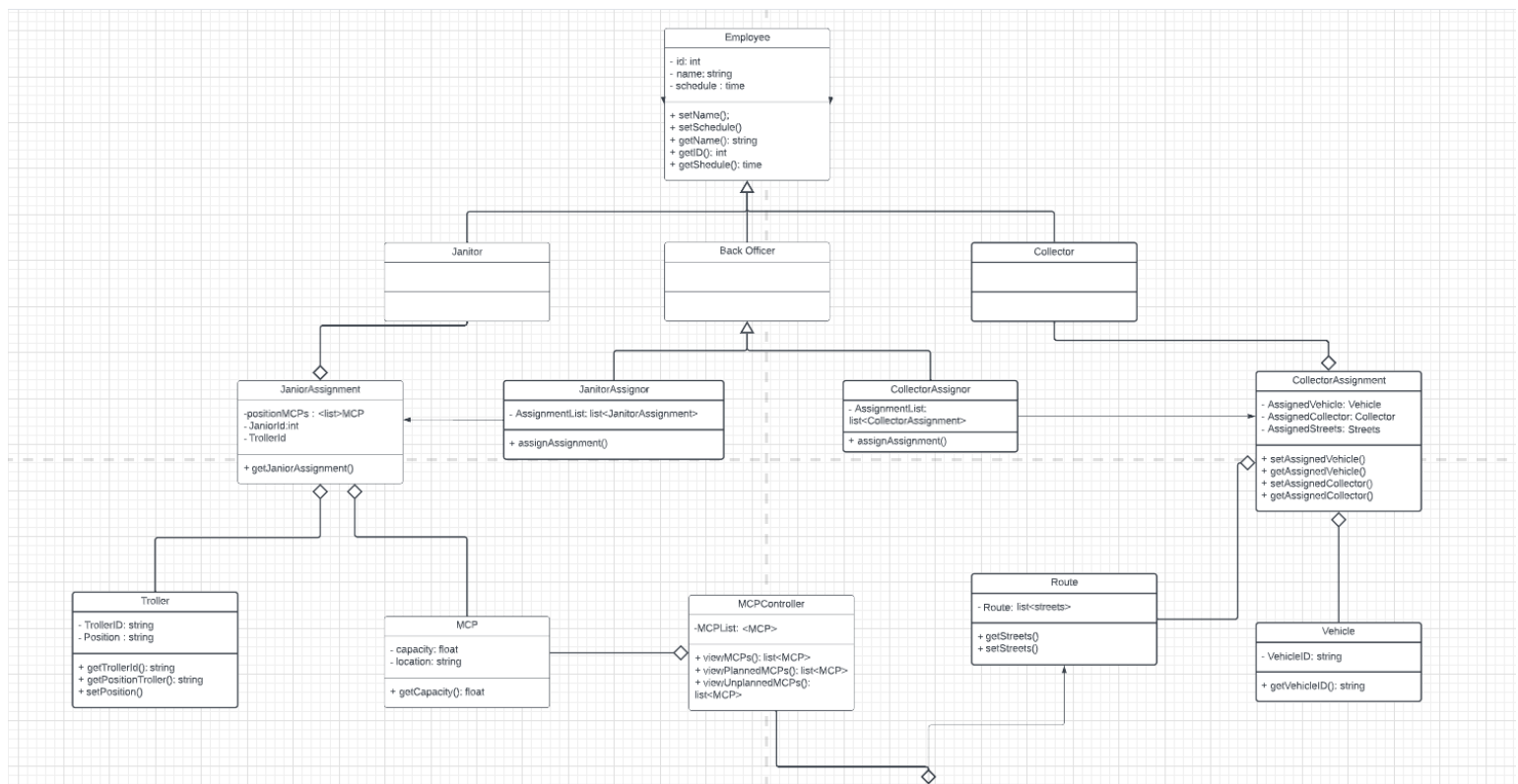
d. Modify route diagram



e. Login diagram



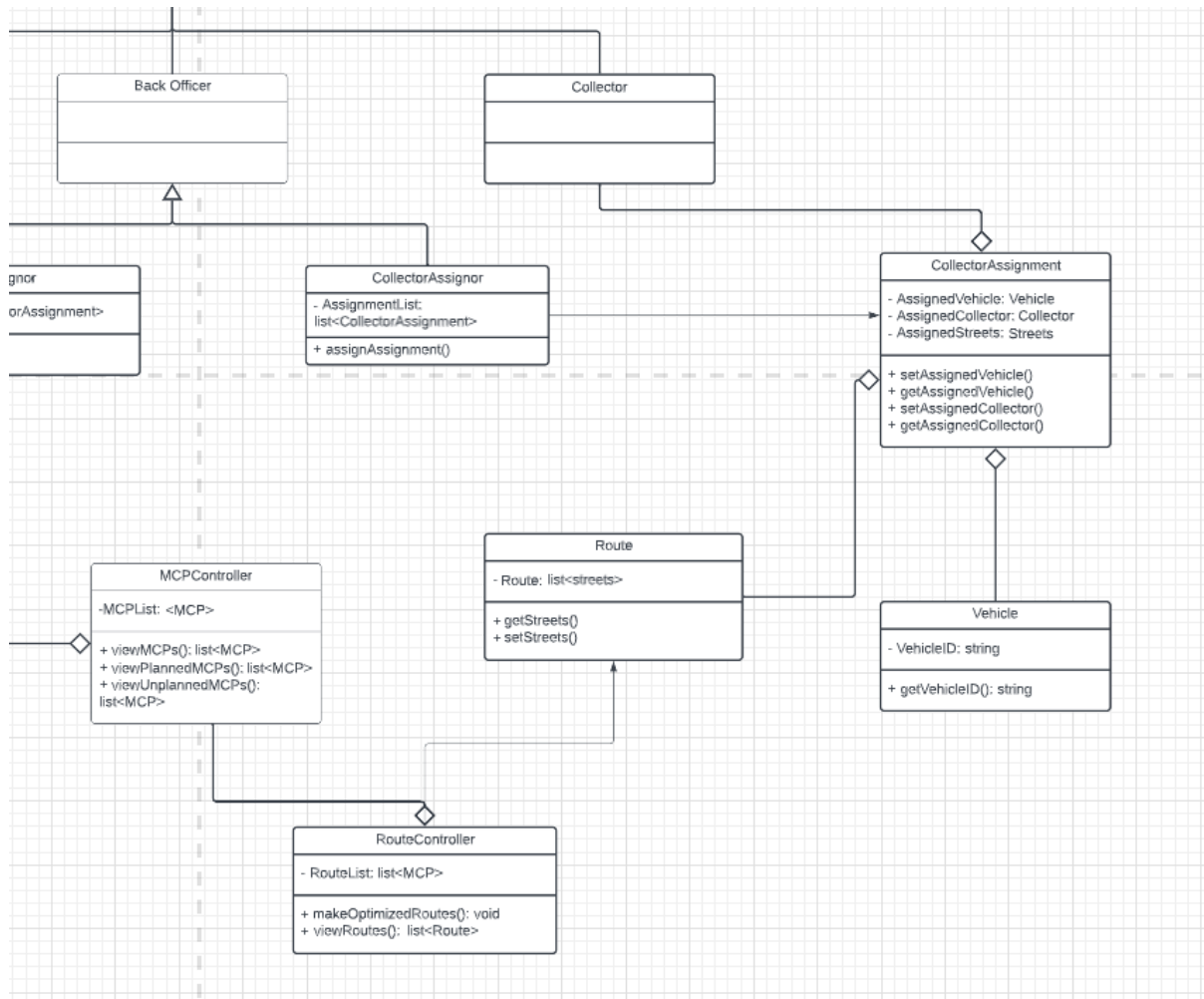
2.3 Draw a class diagram of Task Assignment module as comprehensive as possible



Task Assignment includes routes and vehicles planning (monthly), assigning vehicles to collectors and MCPs to janitors (daily). Therefore we decided that the first Entity classes are Janitor, Collector, MCP, Vehicle and those managing classes corresponded to them. Class diagram can be divided into according smaller sections:

a. Routes and Vehicles Planning

Each route contains a list of available MCPs on that route, as represented in Entity class **Route**. A plan which includes a route, a list of assigned vehicle and collector is modeled by the **CollectorAssignment** class.

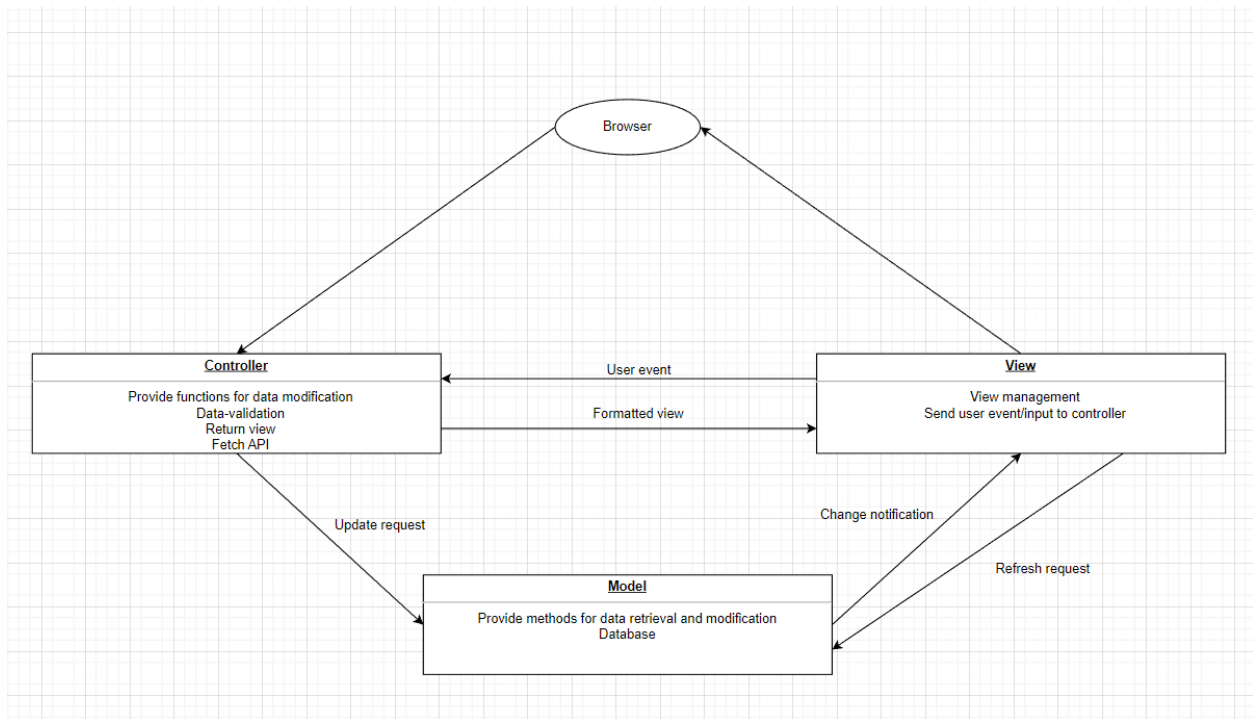


b. Collectors and Janitors Assignment

Each janitor is assigned to a number of MCPs, hence the JanitorAssignment class is made to contain a janitor, a list of MCPs and of course - a troller. Similarly, each vehicle is assigned to a collector, so the CollectorAssignment class will include a vehicle and a corresponding collector to drive it on the assigned route.

3.1 Describe an architectural approach you will use to implement the desired system. How many modules you plan for the whole WMC 2.0 system? Briefly describe input, output and function of each module

a. Architectural approach



- Controller:
 - Call functions from the model to perform modifications on data (to display routes, manage routes, communication, return detailed information)
 - Check for the legitimacy of the credentials provided by the user
 - Return a view depending on user request
 - Processing into instruction
 - Fetch API from the database
- View:
 - Render different UI depending on user request (management UI, search bar,...)
 - Send user event to the controller (click, drag, submit form)
 - Send user input to the controller for application-specific logic

- Send user message, user's modification on the data to the controller
- Send credentials provided by the user to the controller
- Model:
 - Provide API invocation (data provided by sensors from MCPs)
 - Provide data for controller
 - Provide methods for data modification (manage route, task planning, send message, manage vehicles, janitors and collectors information).
 - Calculate and provide optimized routes
 - Provide methods for user authentication (log in, log out).
 - Database provides storage for data (chat history, task, route, profile, calendar)

b. We can plan for the whole WMC 2.0 system

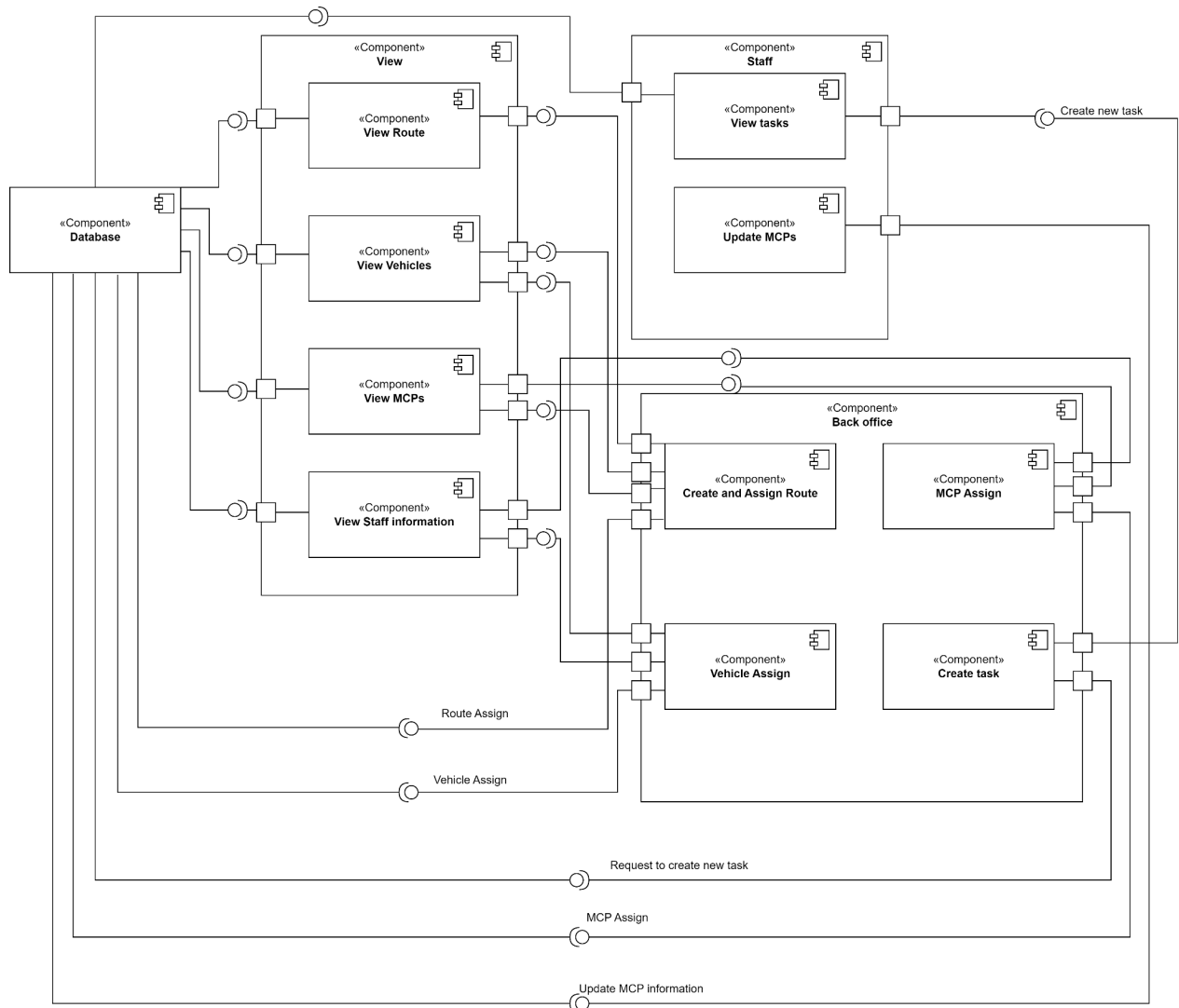
- **9 Modules including** Authentication, Authorization, User profile, MCP information, Route, Communication, Daily task, Check in/Check out task, Receive notifications about MCP status

c. Input, output and function of each module

	Name	Input	Output	Function
Module 1	Authentication	Enter username and password	Create token	System create a protected route to authenticating a user
Module 2	Authorization	Requires an authentication mechanism	User is allowed to manipulate the post	Authorization refers to the process that determines what a user is able to do
Module 3	User profile	User information	Connect and show user information	A user profile is a collection of settings and information associated with a user.
Module 4	MCP information	Enter technical details	Assign task for user	Interact with tasks of MCP

Module 5	Route	Information on the location of MCPs in the area and the number of available Vehicles,	Optimized routes	Create route direction for optimizing fuel and travel
Module 6	Communication	Messages and receiver	Receiver gets messages	Connect users in department
Module 7	Daily task	Task Assignments	Employees receive notifications about work every morning	Notify information about detail task
Module 8	Check in/Check out task	The employee presses the button to accept the job or complete the job	The status of the job is updated on the system	Employee job status management
Module 9	Receive notifications about MCP status	MCPs information	Staff receive notification of MCP status	Notify information about MCP they in charge of

3.2 Draw an implement diagram for Task Assignment module



Describe component diagram:

System has 3 major components:

- View component
 - + Include many sub View components
 - + View components has function as described in task 3.1
 - + View components provide interfaces from the Back office to update data and using interfaces from the database to complete requests.
- Back office component:
 - + Include many sub Back office components
 - + Back office components has function as controller described in task 3.1
 - + Back office components provide interfaces to view to complete requests, using interfaces from Staff and database to modify data.
- Staff component:



- + Include 2 sub Staff components
- + Staff components has function as model described in task 3.1
- + Staff components provide interfaces to complete requests, provide interfaces for Back office to update data

In addition, there are database components:

- Directly interact with database
- Provide interfaces for Back office and Staff to directly access to data on database