TASK 1: REQUIREMENT ELICITATION

TASK 1.1: CONTEXT

Context of this project:

This project is aimed to be applied on an organization scale, based on a similar in-used project with an existing database. The goal is to improve efficiency of service provider Y's waste collection process.

Stakeholders:

Service provider Y's employee, especially their:

- Back officers
- Collectors
- Janitors

Stakeholders' current needs:

They need a more efficient way to improve the efficiency of their garbage collection process.

Their possible current problem:

Their old UWC 1.0 mìght not include a Task Management module or contains a deprecated, inefficient module that needs to be replaced.

Benefits for:

- Back officers: Helps them control and manage the overall waste collection process easily and remotely with real time updates on tasks progress, routes and MCP's capacity.
- Collectors & Janitors: Improve work efficiency with real time updates on MCPs' capacity, online check-in/check-out and chat functions. They can also keep track of their assigned tasks as well as their progress on a daily or weekly basis.

TASK 1.2: REQUIREMENTS

Functional Requirements

1. HIGH PRIORITY

- Every end-user shall login to the system for personalized information from / actions to the system
- Each user shall be able to message other users in real-time with delay less than 1 second
- The system shall be able to access all information (vehicle, MCPs, staff list) from the UWC 1.0 database, or update new information given by back officers

For back officers

- Each back office user shall be able to see all the tasks in the system, including their time, MCPs, status, and person in charge

- Each back office user shall be able to have an overview of vehicles and their technical details
- Each back office user shall be able to have an overview of all MCPs, their max capacity and real-time load (updated every 15 minutes)
- Each back office user shall be able to assign collector to their vehicles, and their distance optimized routes
- Each back office user shall be able to assign janitors to MCPs

For janitors and collectors (called staffs)

- Each staff user can view their task daily and weekly, with all important information displayed in one view
- Each staff user can check in and check out every new shift, this status will be send to back officers

2. MEDIUM PRIORITY

- Each back office user shall be able to see a calendar view of every tasks of every staffs and their status (In queue, Pending, Done)
- Each staff user shall be able to see a calendar view of their own tasks and their status (In queue, Pending, Done)
- The system will notify back officers to assign tasks to staffs (on 1st of every month for collectors and on Monday for janitors)
- The system shall send daily messages (x hour prior to shift time) to staffs to summarize staffs tasks
- The system shall notify back officer when assigned staff does not check-in at the beginning of shift time

3. LOW PRIORITY

- The system shall generate monthly report on the performance of the whole waste collection system

Non-functional Requirements

USABILITY

- The system interface should be in Vietnamese, with the ability to switch to English in the future
- All users should be able to operate the system after 2 hours of training

PERFORMANCE

- The system should be able to store and operable on 1000 MCPs, x staffs at the moment, and 10000 MCPs in five years
- The system should be able to handle y simultaneous users

SECURITY

- The system shall provide password protected access to the system

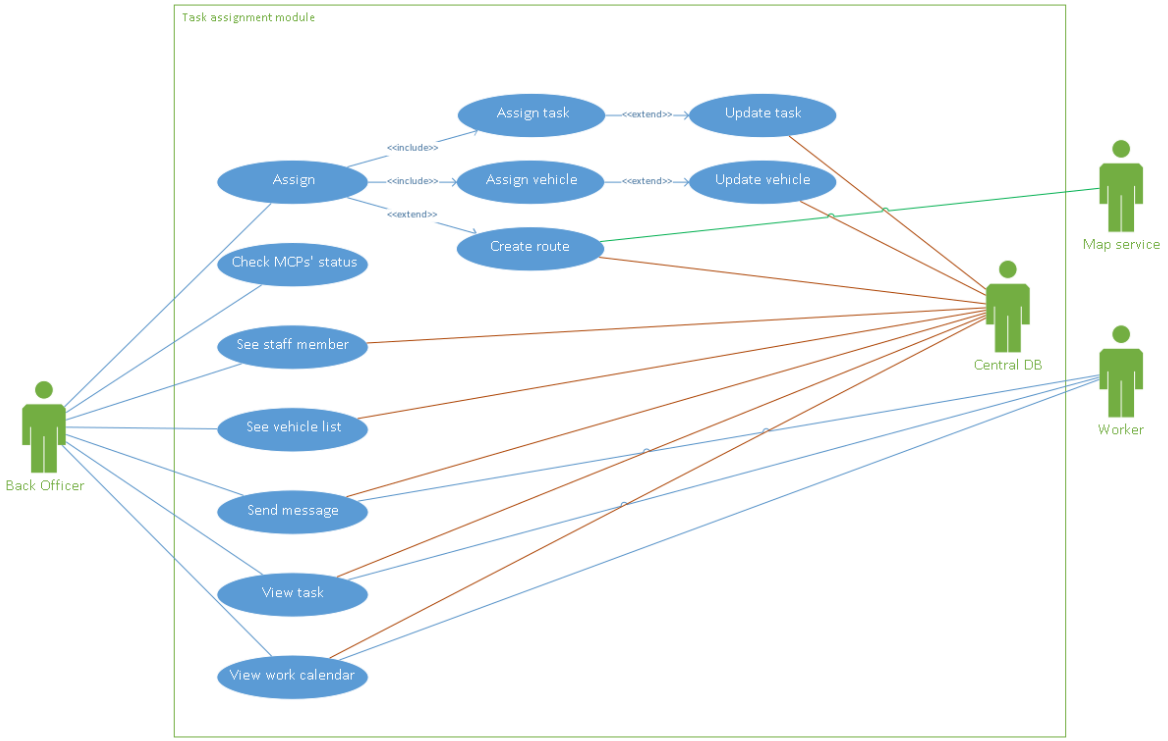
- Only the back officer is granted access to add, modify, remove resources of the organization (database, employees,…)

SUPPORTABILITY

- The system should be accommodate to further changes, improvements in future versions
- The system shall be available on both PC and mobile devices using as web application

TASK 1.3: USE CASE DIAGRAM AND DESCRIPTIONS

Use case diagram



Use case descriptions

ID	TA-1
Name	Assign task
Actor	Back officer, central DB
Description	The Back officer assigns a task by specifying the date and time of task, and staff ID of the personnel to assign to
Trigger	Back officer signals that they want to assign a task

Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	● Action is logged in central DB ● Task and corresponding staff is updated in central DB ● Corresponding staff are notified of the assignment
Normal Flow	**Assign new tasks to staff** 1. System retrieves list of staff from central DB 2. Back officer specifies the date and time of task, and staff to handle the task by choosing from a list 3. Back officer confirms the assignment 4. System logs the action and update the central DB 5. System notifies the assignee(s) of the event 6. System logs the notification event
Alternate Flow	**Update existing tasks** 1. System retrieves list of tasks from central DB 2. Back officer select a task by choosing from a list 3. Back officer modifies the properties of the task 4. Back officer confirms the assignment 5. System logs the action and update the task accordingly 6. System notifies the assignee(s) of the event 7. System logs the notification event
Exception(s)	● Malformed data: Unexpected data type ● Conflicting data: i.e. Requires presence at two location at the same time

ID	TA-2
Name	Assign vehicle
Actor	Back officer, central DB
Description	Back officer assigns a vehicle by specifying vehicle ID, and staff ID of the personnel to assign to
Trigger	Back officer signals that they want to assign a vehicle
Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	● Action is logged in central DB ● Corresponding vehicle and staff is updated in central DB ● Corresponding staff are notified of the assignment
Normal Flow	**Assign vehicle to staff** 1. System retrieves lists of vehicle, staff and route from central DB 2. Back officer specifies a vehicle, staff and route by choosing from the

	list 3. Back officer confirms the assignment 4. System logs the action and update the central DB 5. System notifies the assignee(s) of the event 6. System logs the notification event
Alternate Flow	**Update existing vehicle assignment** 1. System retrieves list of vehicle from central DB 2. System retrieves list of staff from central DB 3. Back officer select a vehicle or staff from a list 4. Back officer modifies the properties of the entity selected 5. Back officer confirms the assignment 6. System logs the action and update the task accordingly 7. System notifies the assignee(s) of the event 8. System logs the notification event
Exception(s)	● Malformed data: Unexpected data type ● Conflicting data: i.e. Requires presence at two location at the same time

ID	TA-3
Name	Create route
Actor	Back officer, central DB
Description	Back officer creates a route by specifying route properties and retrieves an optimized route with said properties from a map service
Trigger	Back officer signals that they want to create a route
Precondition	● User is authenticated ● User is authorised ● Central DB online
Postcondition	● Action is logged in central DB ● Resulting route is stored in central DB
Normal Flow	1. System retrieves a map model from map service 2. Back officer specifies a start point and end point for the new route 3. Back officer specifies intermediary nodes for the route 4. Back officer confirms the request 5. System sends route info to map service requesting for optimized route 6. Map service sends back optimized route 7. System logs the action and update the central DB
Alternate Flow	None
Exception(s)	● Malformed data: Unexpected data type ● Map service offline

ID	TA-4
Name	See staff list
Actor	Back officer, central DB
Description	Back officer queries system about list of staff
Trigger	Back officer signals they want to see the list of all staff
Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	Action is logged in the central DB.
Normal Flow	1. System retrieves list of staff member from central DB and display 2. System logs the interaction
Alternate Flow	None
Exception(s)	None

ID	TA-5
Name	See vehicle list
Actor	Back officer, central DB
Description	Back officer queries system about list of vehicle
Trigger	Back officer signals they want to see the list of all vehicles
Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	Action is logged in the central DB.
Normal Flow	1. System retrieves list of vehicles from central DB and display 2. System logs the interaction
Alternate Flow	None
Exception(s)	None

ID	TA-6

Name	See MCPs' list and status
Actor	Back officer, central DB
Description	Back officer queries system about status of MCPs
Trigger	Back officer signals they want to see the list of all MCPs
Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	Action is logged in the central DB.
Normal Flow	1. System retrieves list of MCPs from central DB and display their status 2. System logs the interaction
Alternate Flow	None
Exception(s)	None

ID	TA-7
Name	View task
Actor	Back officer, staff, central DB
Description	Staff looks up the details of their assigned task(s). Back officer can also looks up the full details of all task(s)
Trigger	User signals that they want to query the status of task(s)
Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	Action is logged in the central DB.
Normal Flow	**Standard user queries for task data** 1. User sends request to system for task data 2. System retrieve task data from central DB and filter according to user's staff ID 3. System sends back filtered task data to user 4. System logs the interaction
Alternate Flow	**Back officer queries for task data** 1. Back officer sends request to system for task data 2. System retrieves task data from central DB 3. System sends back data to Back officer 4. System logs the interaction
Exception(s)	

ID	TA-8
Name	View work calendar
Actor	Back officer, staff, central DB
Description	Staff look up the details of their work calendar. Back officer can also look up the full details of all staffs' work calendar.
Trigger	User signals that they want to query the status of the work calendar
Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	Action is logged in the central DB
Normal Flow	**Standard user queries for work calendar** 1. User sends request to system for work calendar 2. System retrieve user's work calendar from central DB 3. System sends back work calendar to user 4. System logs the interaction
Alternate Flow	**Back officer queries for work calendar** 1. Back officer sends request to system for work calendar 2. System retrieve all users' work calendar from central DB 3. System sends back work calendar to Back officer 4. System logs the interaction
Exception(s)	None

ID	TA-9
Name	Send message
Actor	Back officer, staff, central DB
Description	User sends message to another user
Trigger	User signals that they want to send a message to another user
Precondition	● User is authenticated ● User is authorized ● Central DB online
Postcondition	Interactions are logged in central DB
Normal Flow	1. System retrieves list of staff 2. User selects a recipient

	3. User writes their message 4. User confirms the interaction 5. System logs the message and send to recipient 6. System notifies the recipient 7. System logs the notification event
Alternate Flow	None
Exception(s)	None