

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



SOFTWARE ENGINEERING (CO3001)

Capstone Project

"Urban waste collection aid - UWC 2.0"

GROUP: Lorem ipsum - CLASS: CC01

Lecturers: Nguyễn Đức Anh
Quản Thành Thơ
Mai Đức Trung
Bùi Công Tuấn

Students: Nguyễn Lê Anh Tuấn - 1852837
Hồ Thanh Bình - 2010929
Nguyễn Hà Thanh Hậu - 2052985
Nguyễn Khôi Nguyên - 2152198
Trương Quang Nhã - 2152822

HO CHI MINH CITY, FEBRUARY 2023



Contents

1	Task 1: Requirement elicitation	3
1.1	Task 1.1: Describe the domain context of Urban waste management in Vietnam. Who are relevant stakeholders? What are their current needs? In your opinion, what benefits UWC 2.0 will be for each stakeholder?	3
1.1.1	The domain context of Urban waste management in Vietnam:	3
1.1.2	The relevant stakeholders:	3
1.1.3	The benefits of UWC 2.0 for each stakeholder:	3
1.2	Task 1.2: Describe all functional and non-functional requirements that can be inferred from the project description. Draw a general use-case diagram for the whole system	4
1.2.1	Functional requirements:	4
1.2.2	Non-functional requirements:	5
1.2.3	General use-case diagram for the whole system:	6
1.3	Task 1.3: For the Task assignment module, draw its use-case diagram and describe the use-case using a table format	7
1.3.1	Janitor's workflow	7
1.3.2	Overview MCP	8
1.3.3	Collector workflow	9
1.3.4	Back officers assign vehicle to janitors and collectors	10
1.3.5	Assign janitors and collectors to MCPs	11
2	Task 2: System modeling	12
2.1	Task 2.1: Draw an activity diagram to capture the business process between systems and the stakeholders in Task Assignment module.	12
2.1.1	Back officers assign task to janitors and collectors, then janitors and collectors check in, do tasks and check out.	12
2.1.2	Janitors gather trash and update MCPs, back officer update MCPs every 15 minutes, a collector is assigned if an MCP is full.	13
2.2	Task 2.2: Think about a possible way for a back officer to assign vehicles to janitors and collectors. Draw a sequence diagram to visualize this process.	14
2.2.1	Back officer updates information of vehicles	14
2.2.2	User fails to login	15
2.2.3	Back officer sends vehicles to janitors and collectors	16
2.3	Task 2.3: Draw a class diagram of Task Assignment module as comprehensive as possible.	17
2.4	Task 2.4: Develop MVP 1 as user interfaces of a Mobile-view Task assignment for Janitors and Collectors.	18
2.4.1	Home screen	18
2.4.2	Screen 2	19
2.4.3	Screen 3	20
2.4.4	Screen 4	21
2.4.5	Screen 5	22
2.4.6	Screen 6	23
2.4.7	Screen 7	24



3	Task 3: Architecture design	25
3.1	Task 3.1: Use a layered architecture to design the UWC 2.0 system. Describe how will you present your User Interface. Describe how will you store your data. Describe how you will access to external services/ APIs	25
3.1.1	The architecture of UWC 2.0 system using layered architecture design: . .	25
3.1.2	UI presentation strategy:	25
3.1.3	Data storage approach:	26
3.1.4	API management:	27
3.2	Draw a component diagram for the Task Assignment module	27
4	Task 4: Implementation - Sprint 1	28
4.1	Setting up an online repository (github, bitbucket, etc) for version control.	28
4.2	Adding documents, materials and folders for Requirement, System modelling and Architectural design. Use the selected version control system to report the changes to these files	29
4.3	Conducted a usability test with the user interface you developed in MVP1. . . .	29

1 Task 1: Requirement elicitation

1.1 Task 1.1: Describe the domain context of Urban waste management in Vietnam. Who are relevant stakeholders? What are their current needs? In your opinion, what benefits UWC 2.0 will be for each stakeholder?

1.1.1 The domain context of Urban waste management in Vietnam:

Urban waste management is one of several significant problems faced by many countries in the world, including Vietnam. An excellent waste management is one of the important points in Sustainable Development Goal (SDG) 11: sustainable cities and communities, and SDG 6: clean water and sanitation, which are the keys to sustainable development, according to the United Nations General Assembly. Nevertheless, there are some problems with the current waste management in Vietnam, especially in Ho Chi Minh City. On the one hand, as the infrastructure in HCM City is constantly being synchronized and strongly developed every day, which follows Part 2 of Section III in the Politburo's Resolution No 31-NQ/TW on orientations and tasks for the development of Ho Chi Minh City by 2030, with a vision to 2045, the amount of garbage that households and companies release into the environment is increasing. However, HCM city's waste management is still ineffective for which the country is making continuous efforts to manage the solid waste in an innovative and creative way. According to a survey, in 2018, the annual HCM City waste production comprised more than 800 thousand tons consisting of 46 percent from municipal sources, the remainder delivered from agriculture and industry. This leads to the creation of UWC 1.0, which fulfils the practical demand of a more thorough garbage managing process and the appearance of some new jobs, such as janitors, back officers and collectors at MCPs. On the other hand, the existing software product, which is UWC 1.0, has some drawbacks that limit janitors from working efficiently. For example, it does not signal to the back officer when an MCP is full capacity, so it is very time-consuming for janitors to go to a full MCP ; also, the route created by the software is sometimes not the most convenient one at that time. Therefore, the UWC 2.0 software will overcome those limitations and create an excellent garbage management for the city.

1.1.2 The relevant stakeholders:

The relevant stakeholders of the UWC 2.0 project include back officers, janitors, collectors, the service provider Y, the organisation X, and the super back officers. The back officers need to control the janitors and collectors. Back officers also need to have enough information of vehicles and all MCPs regularly. The janitors and collectors need to keep in contact with other workers at all time and send messages to others when in need. They also need to be informed about their work calendar and their task on a regular basis. The service provider Y needs to manage garbage collection with high efficiency. The organiser X needs to have an efficient and convenient information management system. The super back officers need to have an overview of the Task Management module and all the MCPs.

1.1.3 The benefits of UWC 2.0 for each stakeholder:

By applying the UWC 2.0 project, the stakeholders will receive many benefits. The back officers can have a better overview of janitors and collectors, as well as their work calendar. Back officers can also update the information of all the MCPs more regularly and can send messages to janitors and collectors faster. The janitors and collectors can communicate with other workers quicker and

can check-in and check-out, as well as have a detailed view of their task on their work calendar on a daily and weekly basis, without scrolling down on the device. The janitors will be assigned the most convenient route and be notified about which MCP to deliver garbage. The collectors will be announced about the vehicle they use and the MCPs they are assigned regularly. The service provider Y will improve efficiency of garbage collection. The organisation X will receive a much more efficient and convenient information management system, so they can save much time and effort. The super back officers will have a better view of the Task Management module and all the MCPs.

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

1.2.1 Functional requirements:

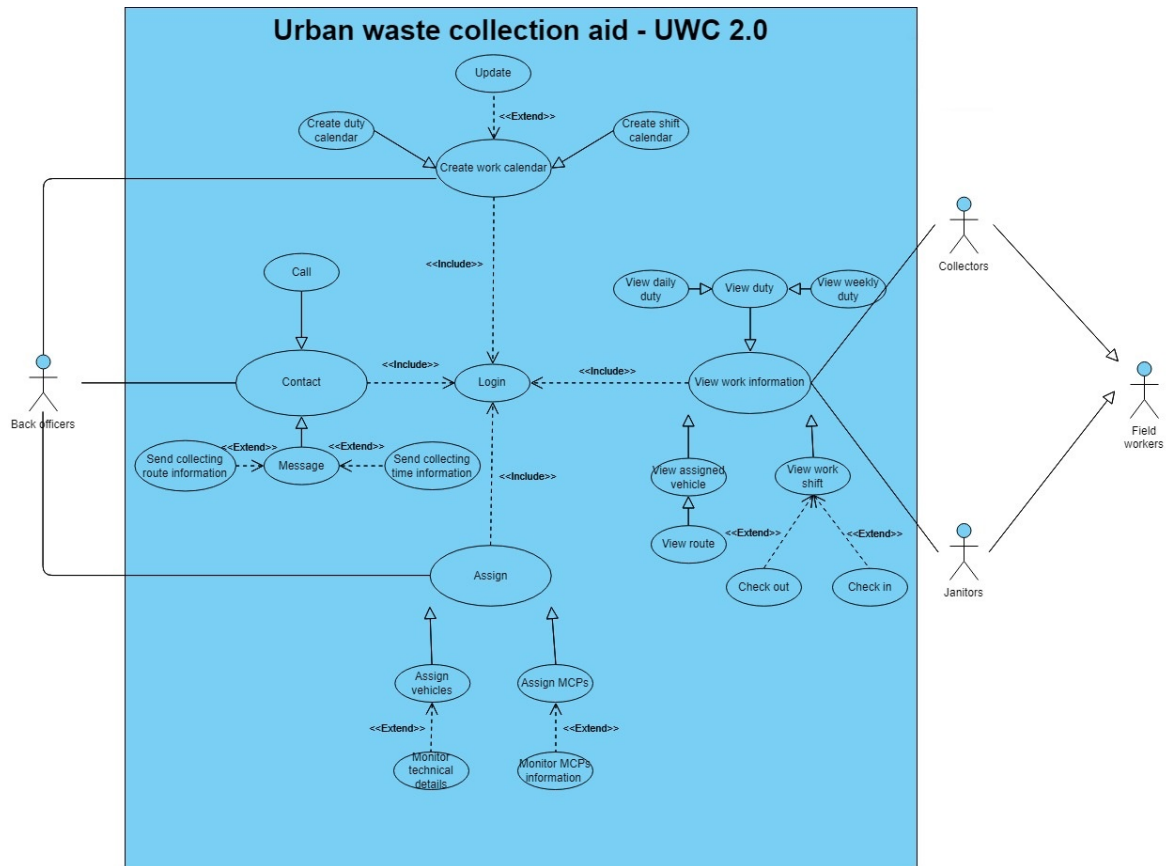
- As a back officer:
 - A back officer shall be able to create work calendar for collectors and janitors.
 - A back officer shall be able to send messages with information about collecting route and time to collectors and janitors.
 - A back officer shall be able to assign vehicles janitors and collectors.
 - A back officer shall be able to assign calendar and tasks for each team of janitors
 - A back officer shall be able to assign route for each vehicles to use.
 - A back officer shall be able to get information of vehicles technical details(weight, capacity, fuel consumption).
 - A back officer shall be able to know MCPs position, scope and information about their capacity.
- As a collector:
 - A collector shall be able to know his/her work calendar.
 - A collector shall be able to have a detail knowledge of his/her task on a daily and weekly basis.
 - A collector shall be able to send messages to or call other collectors, janitors and back officers.
 - A collector shall be able to check in, check out task every day.
 - A collectors shall be able to know which troller to use to collect garbage.
 - A collector shall be able to know which Major collecting points assigned to his/her.
- As a janitor:
 - A janitor shall be able to know his/her work calendar.
 - A janitor shall be able to have a detail knowledge of his/her task on a daily and weekly basis.
 - A janitor shall be able to send messages to or call other collectors, janitors and back officers.
 - A janitor shall be able to check in, check out task every day.

- A janitor shall be able to know which vehicle to use.
- A janitor shall be able to know which route assigned to him/her.
- A janitor shall be able to know which MCP to pick garbage from.

1.2.2 Non-functional requirements:

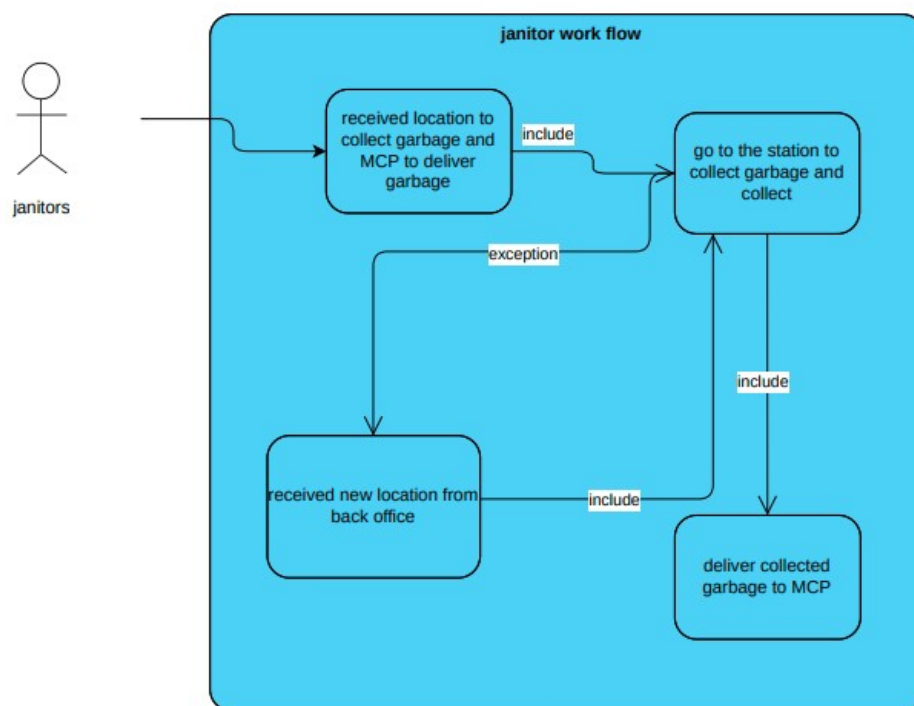
- Communication between janitors, collectors and back officers via message should be in real-time manner with delay less than 1 second.
- All important information about collector's and janitor's task should be displayed in one view (without scrolling down) and be accessed with only 1 (or the least) clicks.
- UWC 2.0 is expected to import and to use at least 95 percent of the existing data from UWC1.0.
- Task Management can call all (or up to 95 percent) of APIs made in UWC1.0.
- The system should be able to handle real-time data from at least 1000 MCPs at the moment and 10.000 MCPs in five years.
- UWC 2.0 system interfaces should be in Vietnamese, with an opportunity to switch to English in the future.
- The system can run for 8 consecutive hours with 5 consecutive concurrents without crashing.
- System can handle 10 concurrent connections, 100 messages sent by each, with error rate of missing message is 0.02 percent
- All the data should be correct and reliable at all time (at least 95 percent in a day).
- The response time of whole system is less than 2 seconds.
- UWC 2.0 can run on the following OS and devices:
 - OS: Android and later, iOS, Windows 10.0,...
 - Devices: iPhone 7.0 and later, iPad, computer,...

1.2.3 General use-case diagram for the whole system:



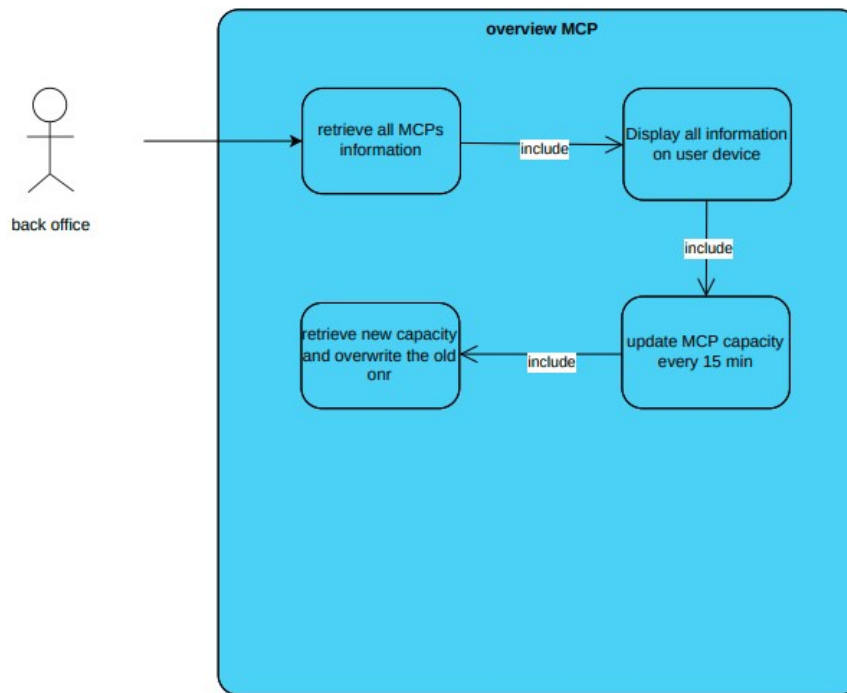
1.3 Task 1.3: For the Task assignment module, draw its use-case diagram and describe the use-case using a table format

1.3.1 Janitor's workflow



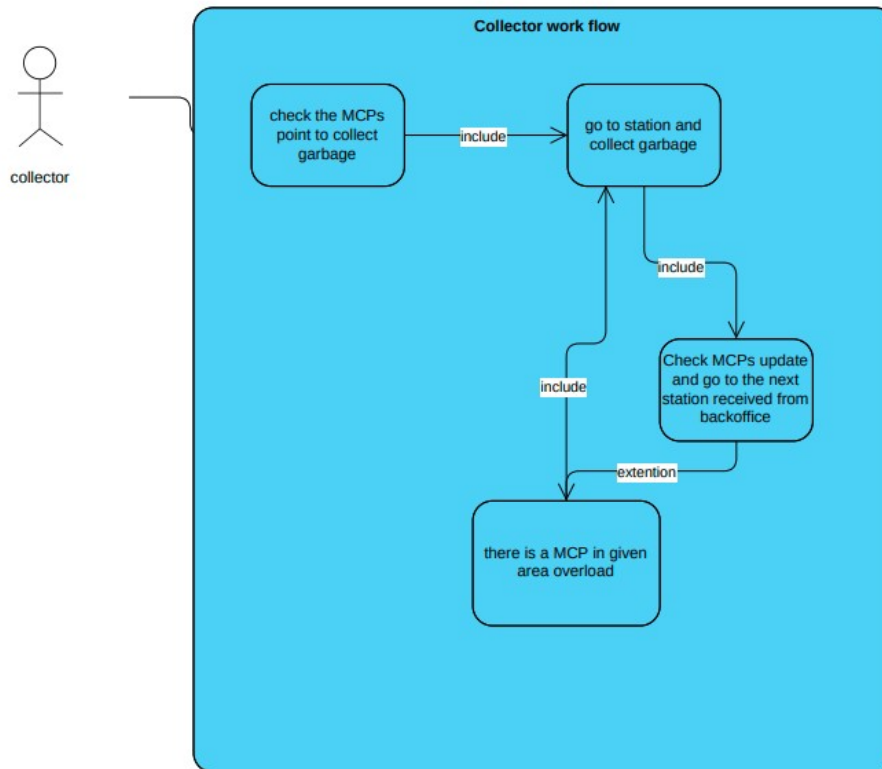
Use-case ID	U1
Use-case name	Janitor work.
Use-case overview	Janitor work flow in day.
Actors	Janitors
Preconditions	The role of the user must be janitor.
Trigger	The back officers sent work location and MCP to janitor.
Steps	1. Received location to collect garbage and MCP to deliver garbage. 2. Go to the station to collect garbage and collect. 3. Deliver garbage collected to MCP.
Post conditions	The garbage is collected from the assigned area to MCP by janitor.
Exception flow	E2: janitor don't have any trolley to work. E3: The MCP janitor was assigned is full.

1.3.2 Overview MCP



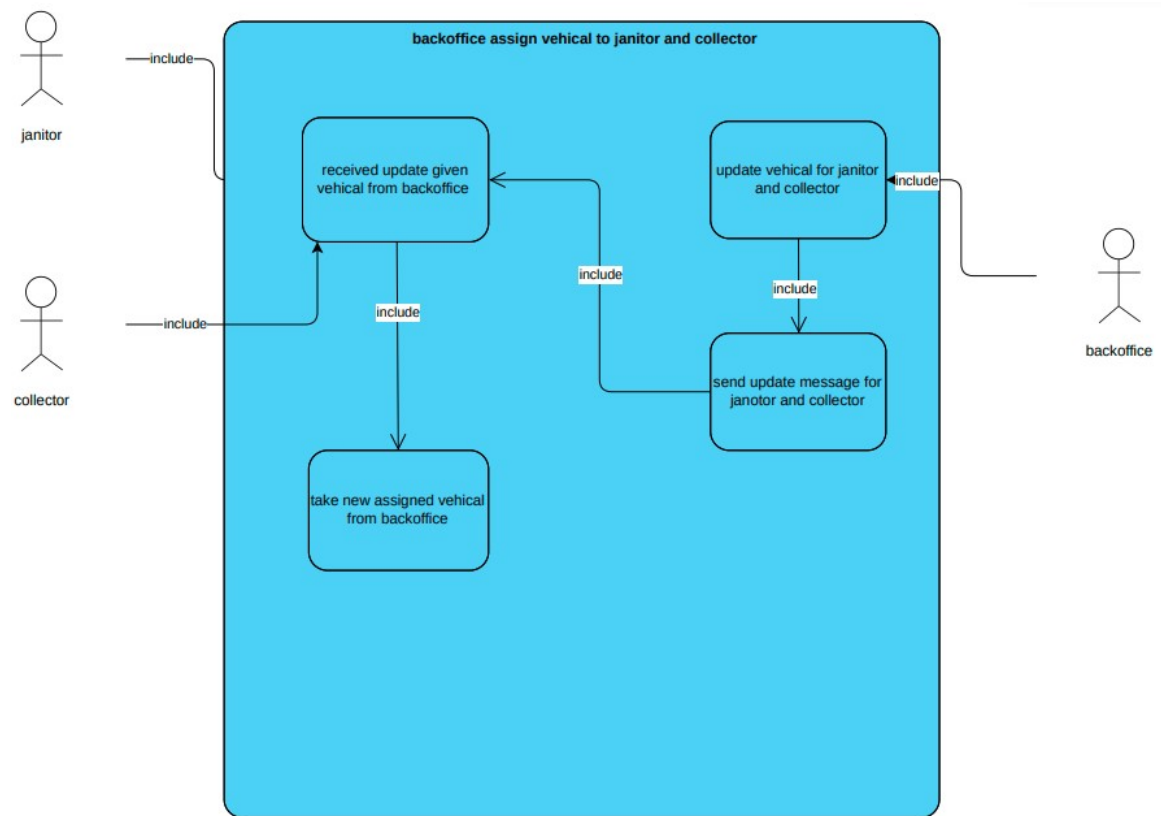
Use-case ID	U2
Use-case name	Overview MCP.
Use-case overview	To provide information about Major collecting points (MCPs) and their current capacity.
Actors	Back officers.
Preconditions	1.The role of the user must be back officer. 2. Back officer is logged in and access to the “Task assignment” on the sidebar.
Trigger	User click MCP overview button.
Steps	1. Retrieve all MCP information. 2. Display all information on user device. 3. Update MCP capacity every 15 min then retrieve capacity from database and overwrite it.
Post conditions	Information is displayed on the screen of user’s device and updated every 15 minutes.
Exception flow	E1:The system fails to retrieve MCPs information. E3:The system fails to update MCPs information.

1.3.3 Collector workflow



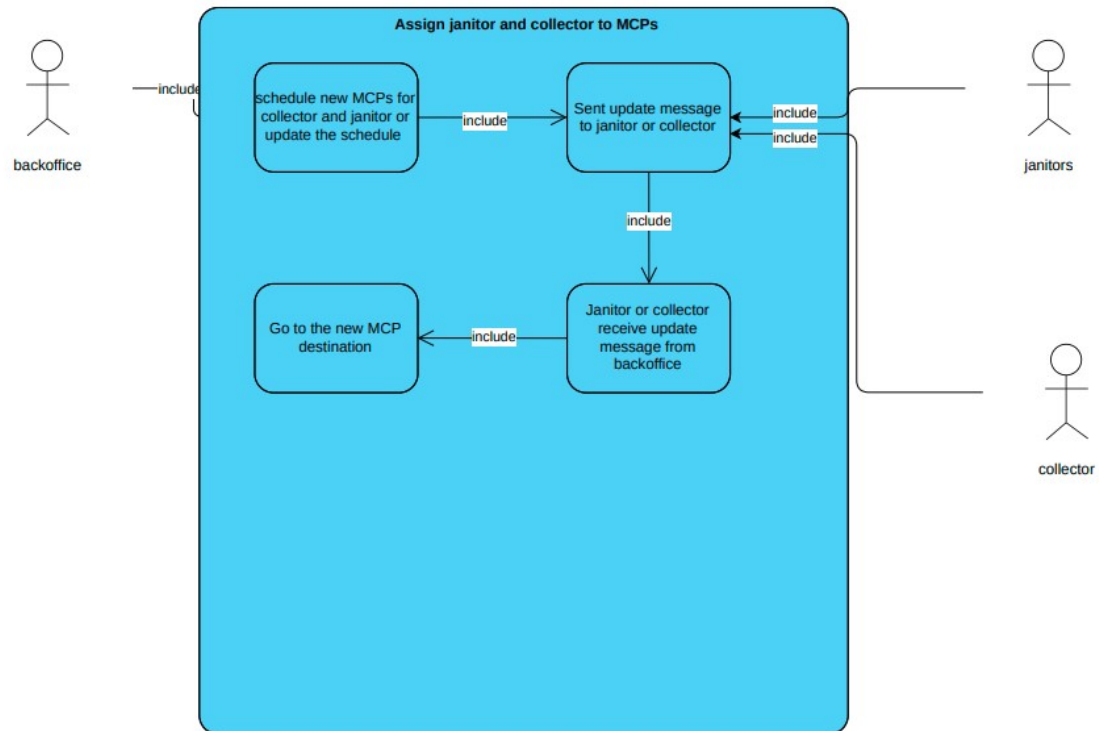
Use-case ID	U3
Use-case name	Collector work.
Use-case overview	Provide and overview about collector work flow.
Actors	Collectors.
Preconditions	1. There must exist free vehicle. 2. The role of the user must be collector.
Trigger	Back officer assigns MCP to collector.
Steps	1. Check the MCP point to collect garbage. 2. Go to the MCP point the collect all garbage from janitors. 3. Check updated MCPs point, given MCPs point and go to the next place.
Post conditions	The garbage at assigned MCP is collected by collector.
Exception flow	E1: There is an MCP in assigned area overloaded.

1.3.4 Back officers assign vehicle to janitors and collectors



Use-case ID	U4
Use-case name	Assign vehicle.
Use-case overview	Back officers assign new vehicles for janitors or collectors when their old vehicles got in trouble.
Actors	Back officers, collectors, janitors.
Preconditions	1. There must exist free vehicle. 2. The old vehicle got in trouble. 3. The assigner must be back officer.
Trigger	Some janitors or collectors don't have any vehicle to work.
Steps	1. Janitors update assigned vehicle for everyone. 2. Send update messages to janitors and collectors. 3. Janitors and collectors received message from back officers. 4. Collectors and janitors using new vehicle.
Post conditions	Every janitor and collector working must have a vehicle.
Exception flow	E1: No free vehicle.

1.3.5 Assign janitors and collectors to MCPs

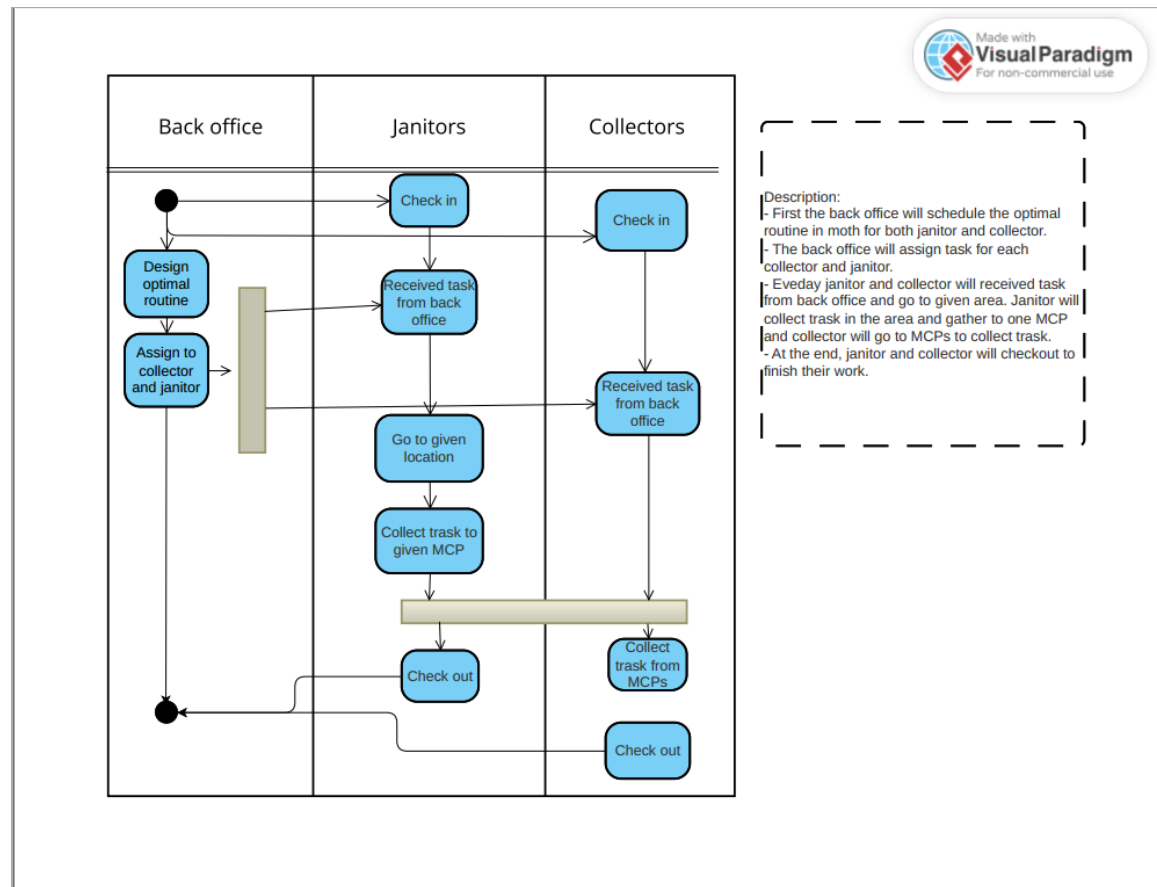


Use-case ID	U5
Use-case name	Assign MCPs.
Use-case overview	Assign new MCPs for collectors and janitors.
Actors	Back officers, collectors, janitors.
Preconditions	The mobile app and website are ready to use.
Trigger	Some areas are too dirty (have too many garbage) so the people there are not enough.
Steps	1. Schedule new MCPs for collectors and janitors or update the MCP. 2. Send updated messages to janitors and collectors. 3. Janitors and collectors received message from back officers. 4. Collectors and janitors go to new MCPs.
Post conditions	Janitor and collector work at new MCP.
Exception flow	E1: All the MCPs are fully-loaded.

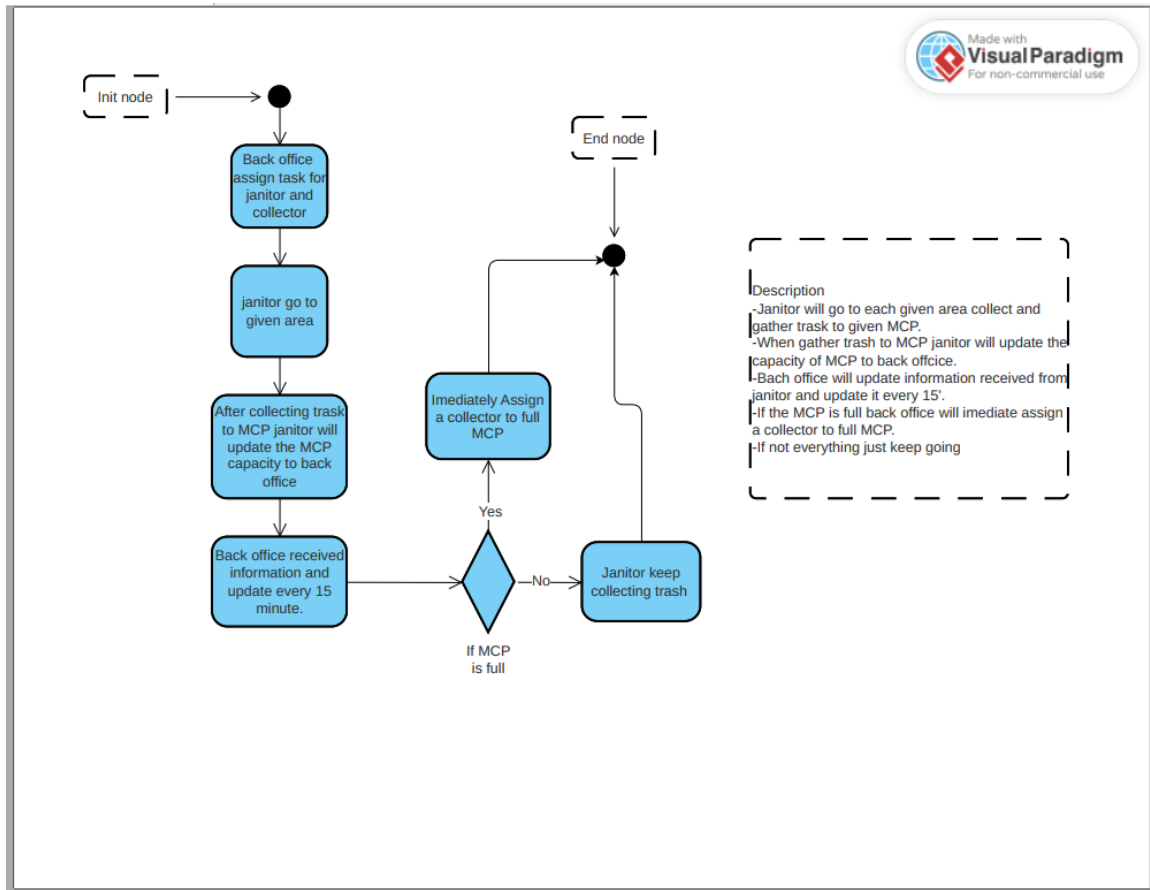
2 Task 2: System modeling

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

2.1.1 Back officers assign task to janitors and collectors, then janitors and collectors check in, do tasks and check out.

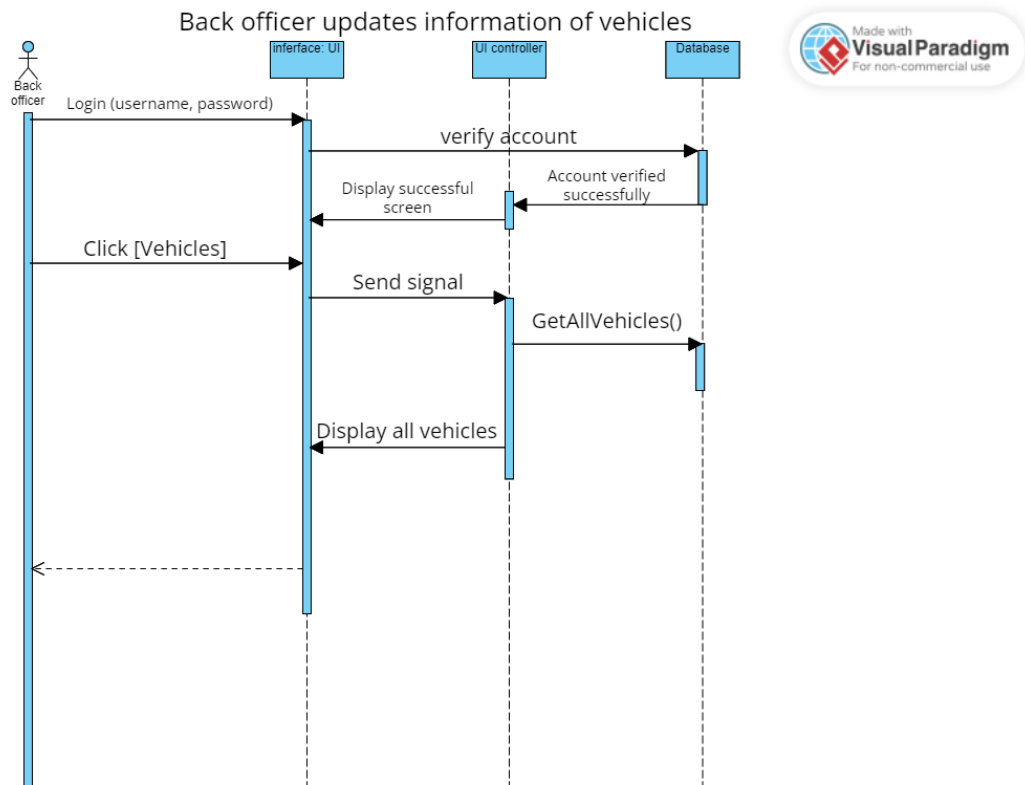


2.1.2 Janitors gather trash and update MCPs, back officer update MCPs every 15 minutes, a collector is assigned if an MCP is full.



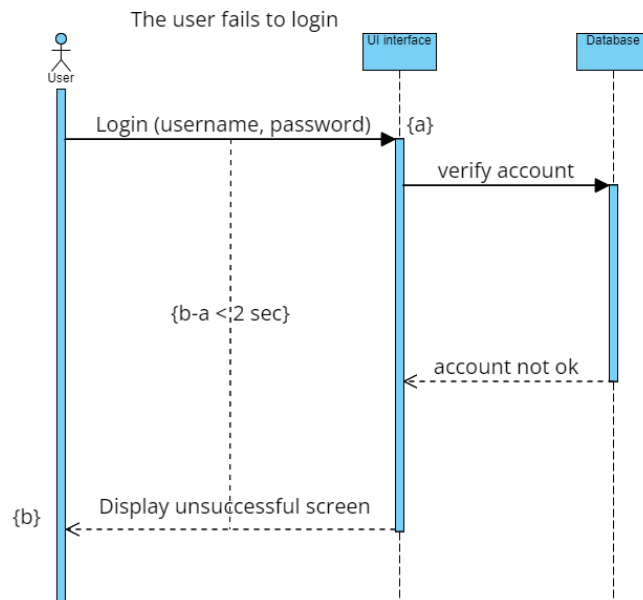
2.2 Task 2.2: Think about a possible way for a back officer to assign vehicles to janitors and collectors. Draw a sequence diagram to visualize this process.

2.2.1 Back officer updates information of vehicles



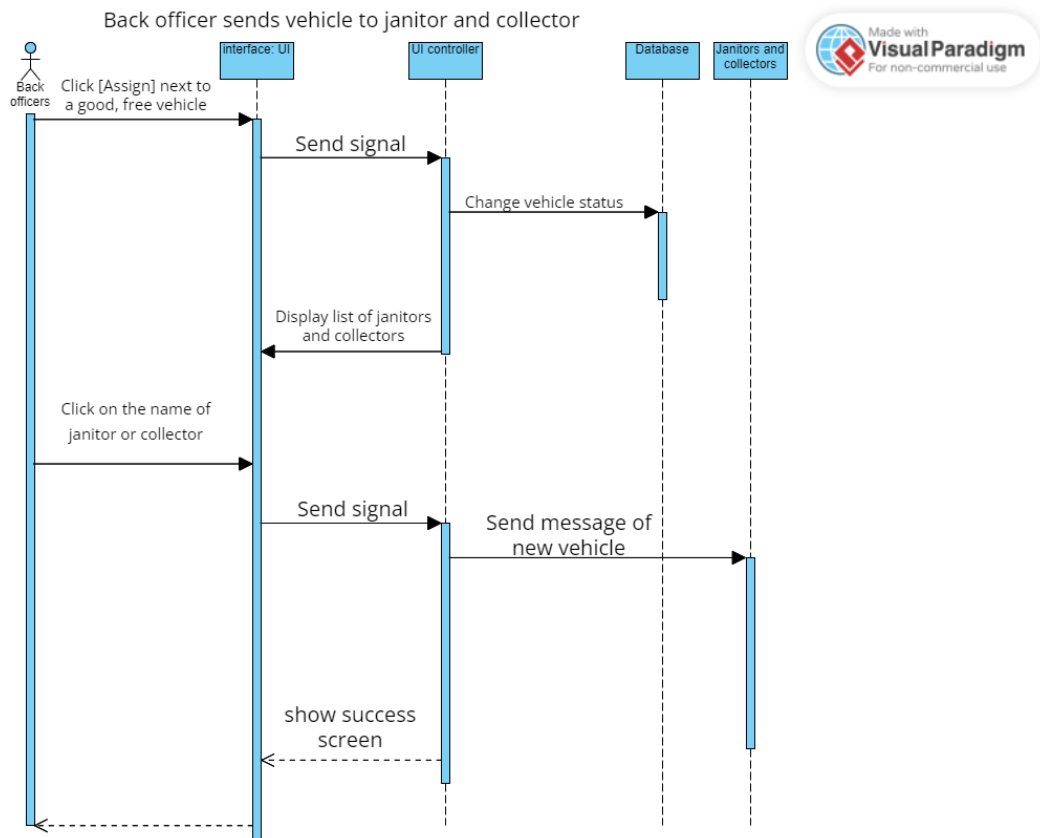
Description: At first, the back officer logs in with his/her username and password. Then, the system will verify the account using the database. If the information is correct, the system sends a signal to the interface, and the screen displays a successful login result. After that, the back officer clicks "Vehicles" on the UI. The UI will then get all vehicles data from the database and display all of them on the UI for the back officer to view.

2.2.2 User fails to login



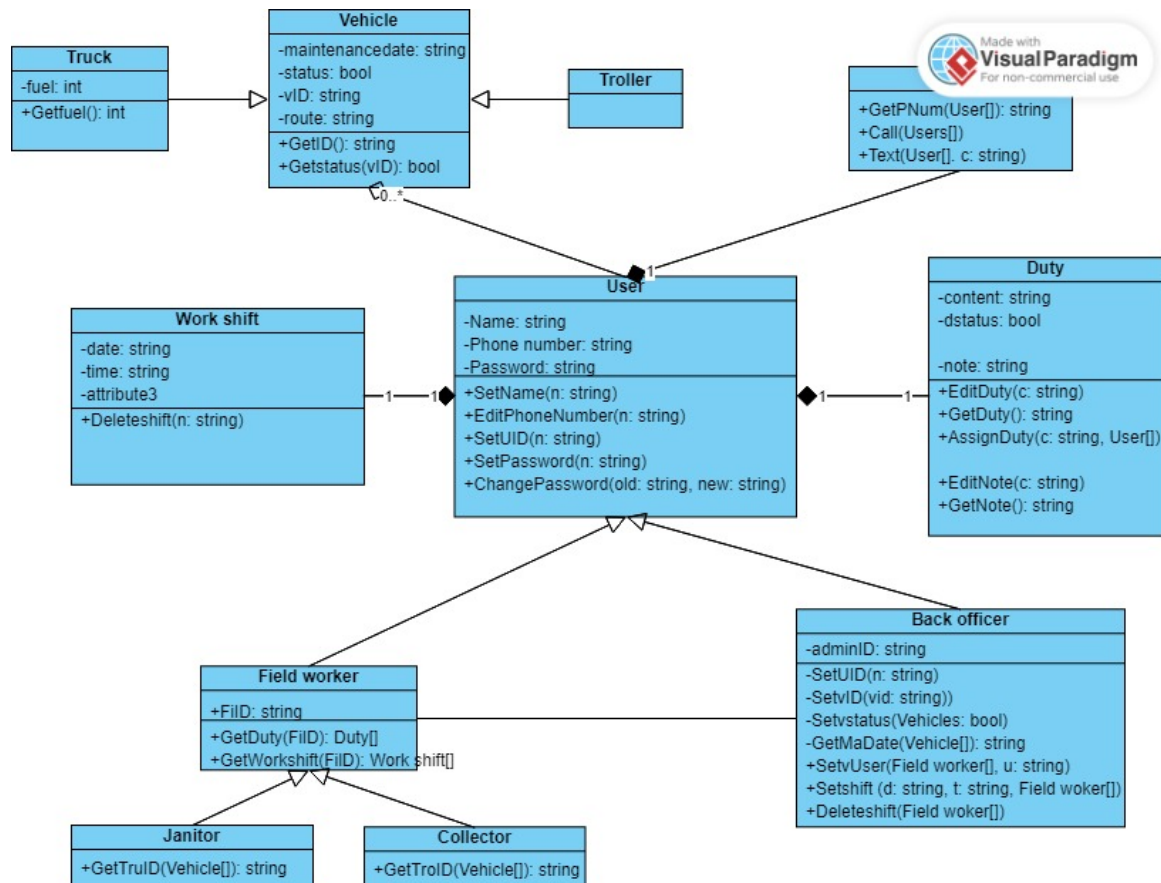
Description: If the information login is incorrect (because the username or password or both is typed wrong), the system sends a signal to the interface, and the screen displays an unsuccessful login result. The duration between the time user enters information and the time result appears must be smaller than 2 seconds.

2.2.3 Back officer sends vehicles to janitors and collectors



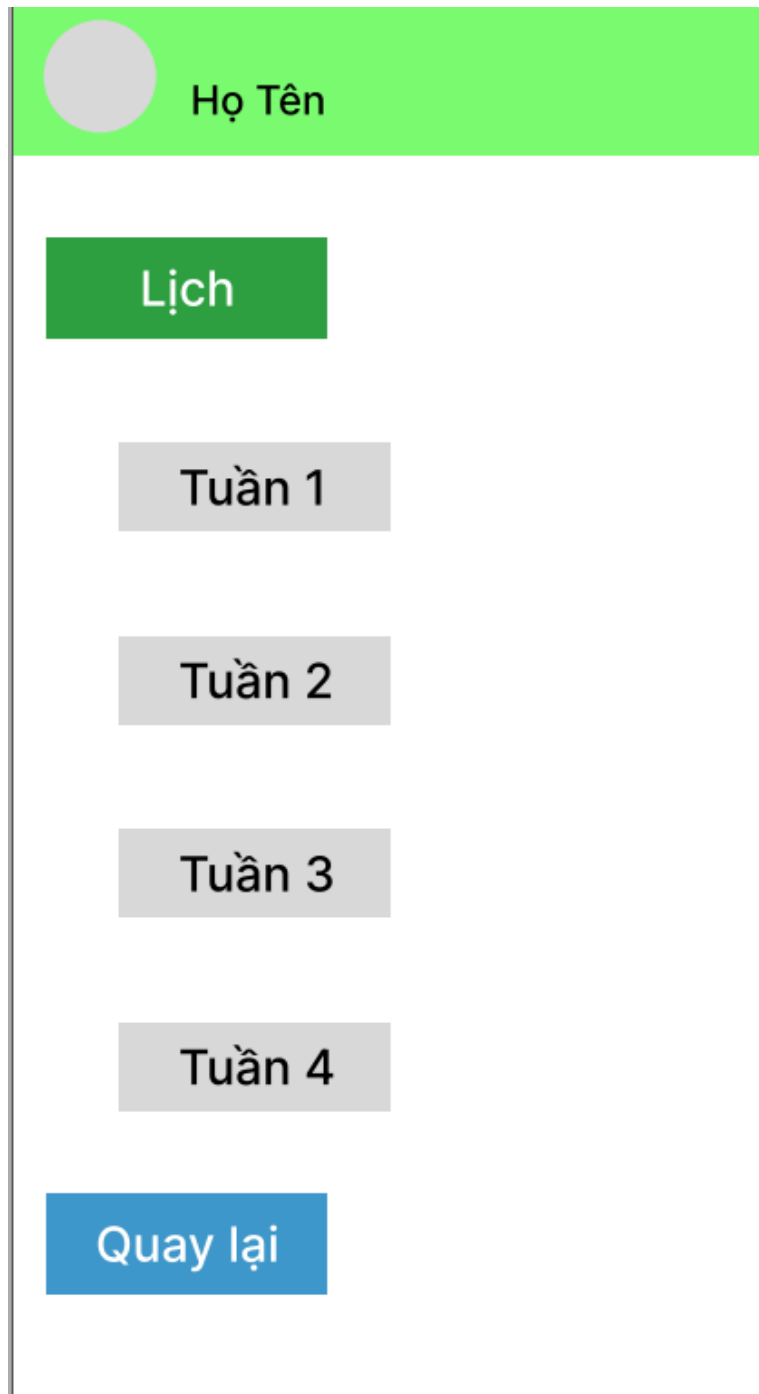
Description: When viewing all vehicles, the back officer chooses a good, free vehicle and clicks "Assign" next to it to choose the vehicle for the janitor or collector. The UI sends a signal to the UI controller, then changes the vehicle status in the database. The screen then displays the list of janitors and collectors. The back officer clicks on the name of the person he/she wants to assign the vehicle. The UI controller will send a message about the new vehicle to the janitor or collector, and the screen of the back officer displays a successful assigning result.

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



2.4 Task 2.4: Develop MVP 1 as user interfaces of a Mobile-view Task assignment for Janitors and Collectors.

2.4.1 Home screen

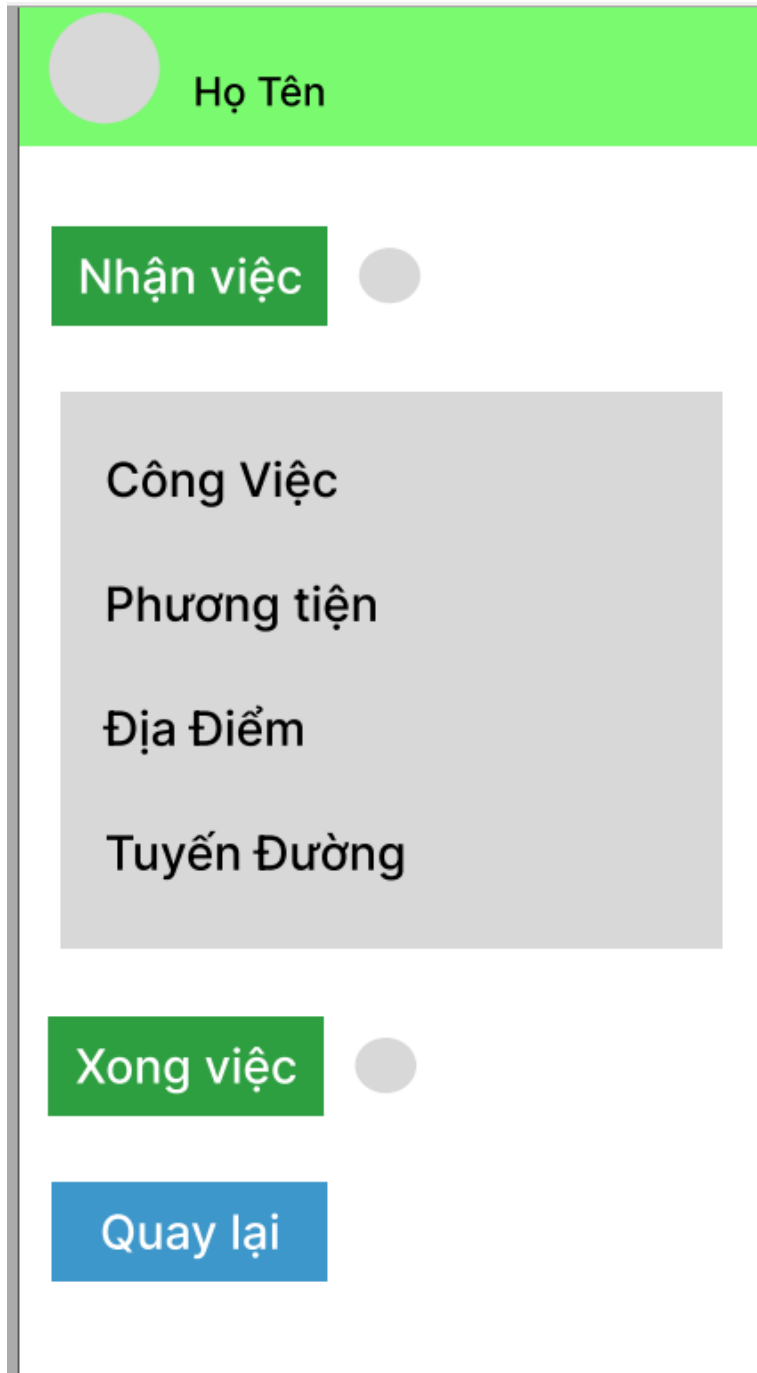


2.4.2 Screen 2



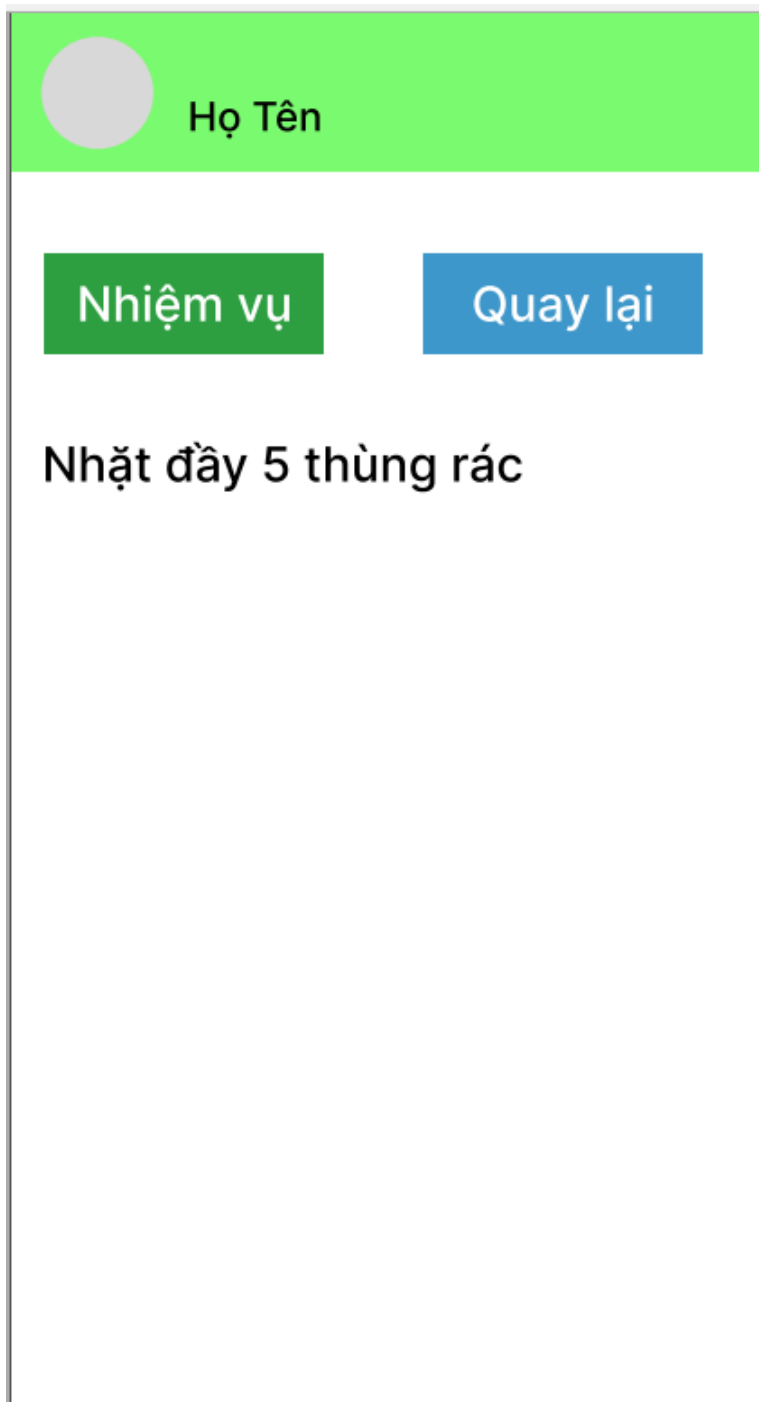
The image shows a mobile application screen with a light green header bar. On the left of the header is a grey circular placeholder for a profile picture, and on the right is the text 'Họ Tên'. Below the header, the screen has a white background. It contains six grey rectangular buttons arranged in two rows of three: 'Thứ hai', 'Thứ ba', 'Thứ tư' in the first row, and 'Thứ năm', 'Thứ sáu', 'Thứ bảy' in the second row. Below these is a single grey button labeled 'Chủ nhật'. At the bottom left of the screen is a blue rectangular button with the text 'Quay lại' in white.

2.4.3 Screen 3




The screenshot displays a mobile application interface for task management. At the top, a green header bar contains a grey circular placeholder for a profile picture and the text "Họ Tên". Below this, a green button labeled "Nhận việc" (Accept task) is accompanied by a grey circular progress indicator. A large grey rectangular area in the center lists four task details: "Công Việc", "Phương tiện", "Địa Điểm", and "Tuyến Đường". Below the grey area, another green button labeled "Xong việc" (Task completed) is shown with a grey circular progress indicator. At the bottom, a blue button labeled "Quay lại" (Go back) is visible.

2.4.4 Screen 4





2.4.5 Screen 5

 **Họ Tên**

Phương tiện

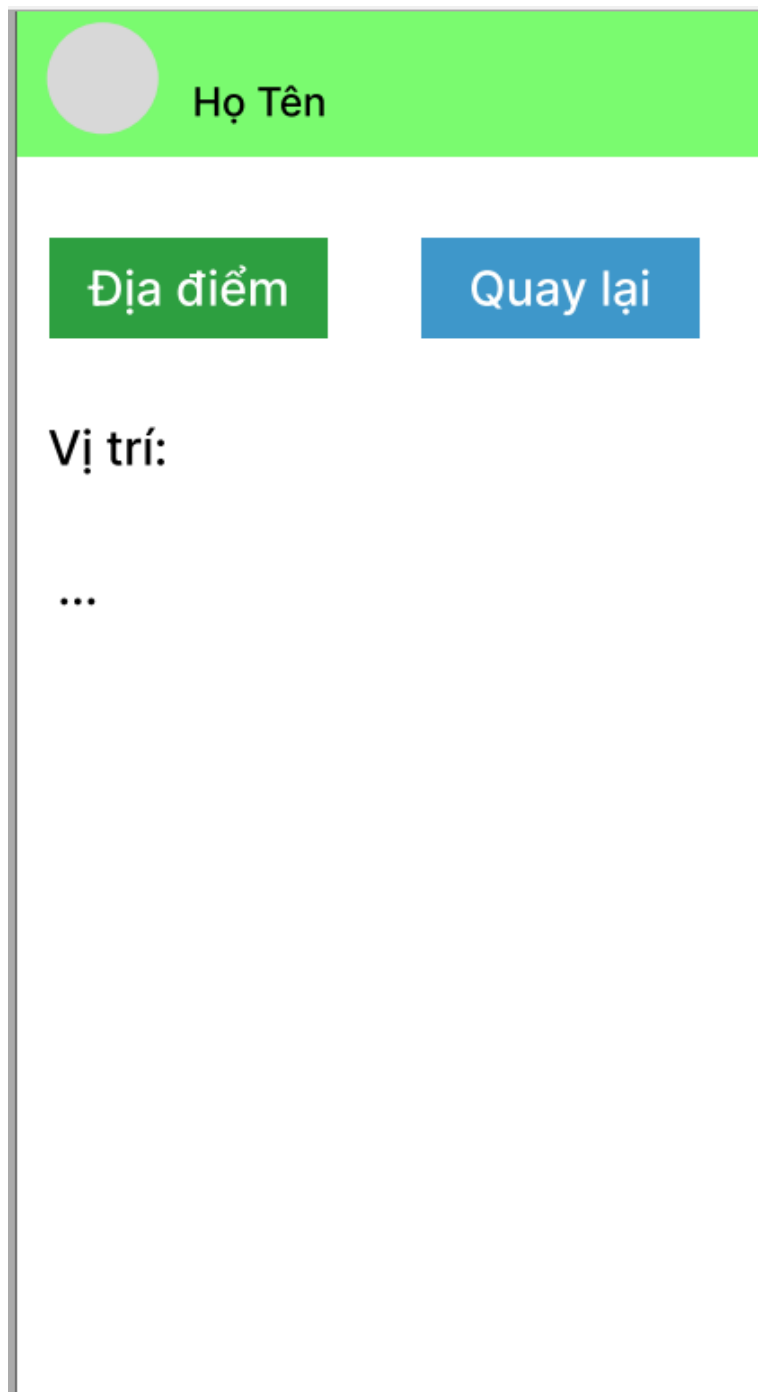
Quay lại

Biển số: 99A-333.33

Vị trí:

...

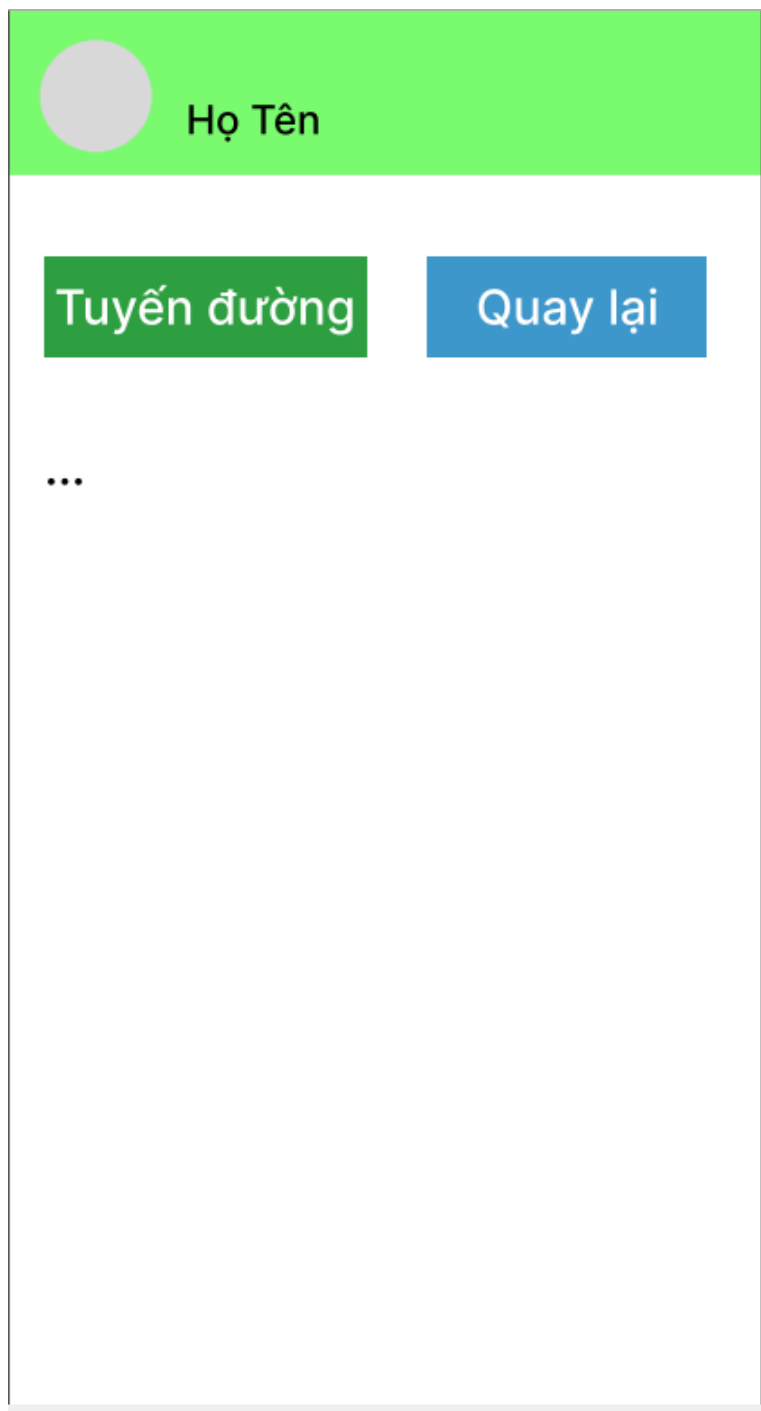
2.4.6 Screen 6



Screen 6 is a mobile application interface. It features a green header bar at the top containing a grey circular profile picture placeholder and the text "Họ Tên". Below the header, there are two buttons: a green button labeled "Địa điểm" and a blue button labeled "Quay lại". Underneath these buttons, the text "Vị trí:" is displayed, followed by three dots "...".



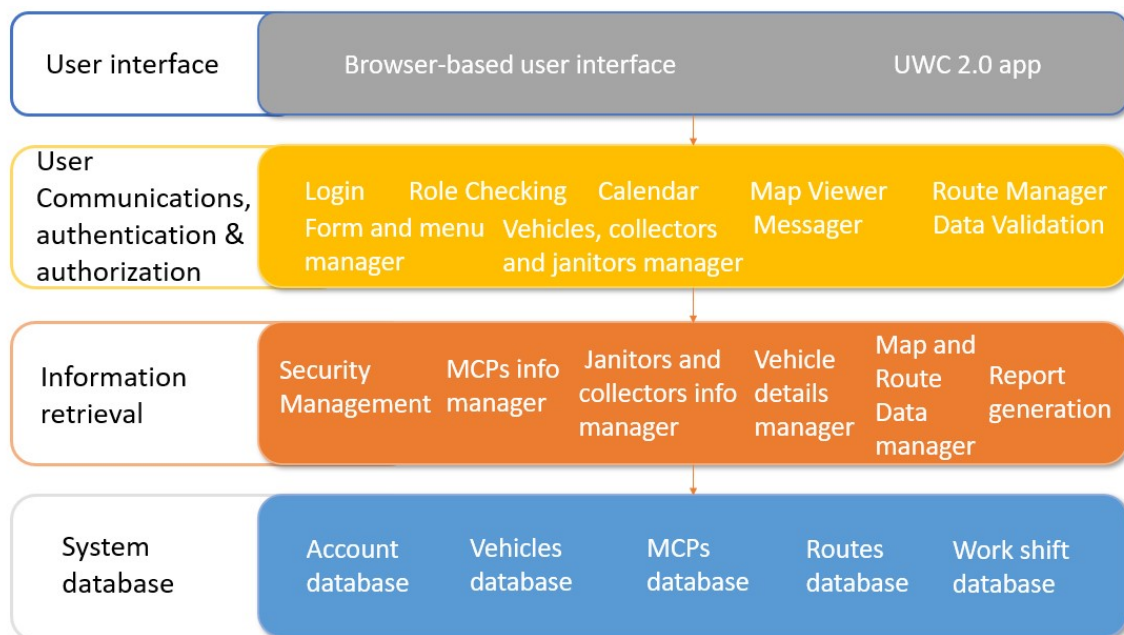
2.4.7 Screen 7



3 Task 3: Architecture design

3.1 Task 3.1: Use a layered architecture to design the UWC 2.0 system. Describe how will you present your User Interface. Describe how will you store your data. Describe how you will access to external services/ APIs

3.1.1 The architecture of UWC 2.0 system using layered architecture design:



3.1.2 UI presentation strategy:

The app should provide users with all the information they need to efficiently manage their tasks and communicate with other team members, while also providing valuable insights and data. We want to make the UI intuitive, user-friendly and visually appealing.

1. Login screen: The login screen will be simple with a clean design so the users can navigate and enter their login credentials with ease. There will be remember me option so that the users don't need to reenter their login credentials every time they access the app. Forget Password function will also be included in case the users forget their password or want to reset their password. There should be a toggle switch that allows the users to change the language (English and Vietnamese).
2. Home screen: Depending on the type of users, the home screen will be different:
 - For back officers: There should be a list of icons representing each function: overview of janitors, collectors, work calendars, vehicles, MCPs; task management; messaging; etc.

- For janitors and collectors: The home screen will start up with a work calendar, the current day should be highlighted. This would allow janitors and collectors to catch up with the events and tasks of the current day and check in much faster without the need of any extra click. The daily task and weekly task can be access by either clicking the day or the week on the calendar. For other functions like user profile, messaging, map viewer, etc. these will be accessible through the sidebar located at the left side of the app.
3. Task management: as this function would be used the most, everything should be clear and simple, the content of this page would be different for each type of users:
 - For back officers: Back officers would do the assignment of task for janitors and collectors, when a specific user is chosen, their info would show up, including their current tasks and their work calendars.
 - For janitors and collectors: while the janitors and the collectors can already view their tasks at the work calendars, this function would provide a more detail description of their tasks in the manner of a list, and the ability to note, comment, mark the task as completed or report to the back officers if there are any problems arisen with the task.
 4. Vehicles view: The vehicles will be displayed as a list. Vehicle status such as "available" or "in use" should be indicated directly on each vehicle. On clicking the vehicle, specific details of the vehicle will be displayed. The "assign" button for the back officers should be directly next to every vehicle.
 5. Calendar view: the design of this should be a simple calendar, users should be able to see their tasks directly on each day
 6. Map view and route management: This would be where the users view the map and for the back officers to assign routes for vehicles. The point of interest like MCPs should be displayed on the map.
 7. Messaging and communication: This would allow users to see any new messages or notifications from their colleagues or the back officers. The design should be like a simple message system, with the box to enter and send messages at the bottom of the screen and the box showing the messages right above.
 8. Reporting and analytic: The app should provide detailed reporting and analytic on the waste collection process, including data on collection times, route efficiency, and waste volumes.

3.1.3 Data storage approach:

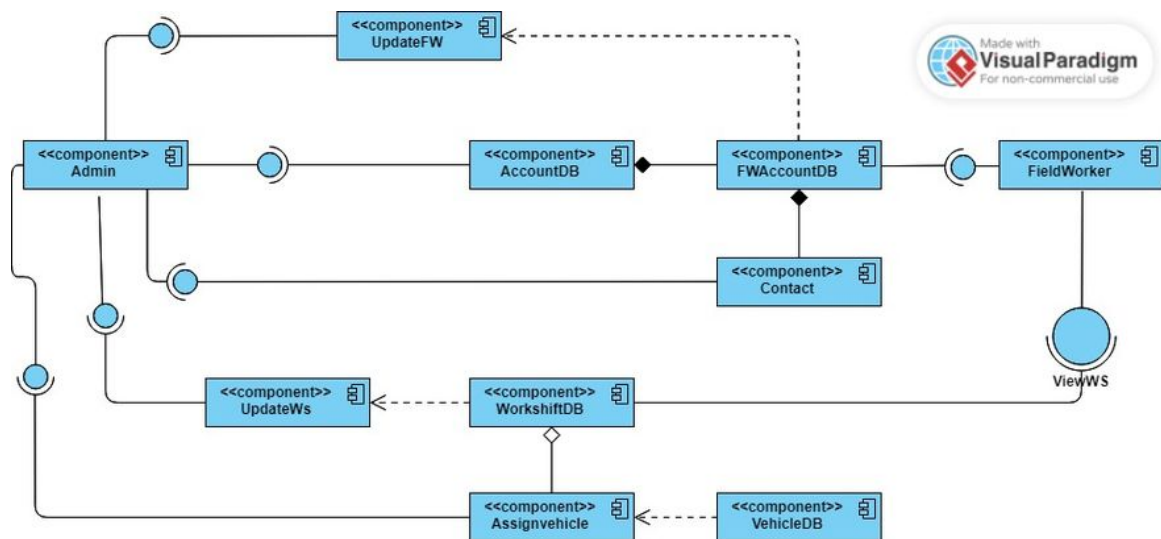
Waste collection involves a large amount of data, including account information, work shift, routes, vehicle information, and collection data, a relational database management system (RDBMS) would be a suitable data storage approach. RDBMS is a widely used and mature data storage approach that allows for efficient storage and retrieval of structured data. It provides a well-defined schema and enforces data integrity constraints, which is particularly important in applications where data accuracy is crucial.

To ensure the application can handle a high volume of transactions, it may be beneficial to use a distributed RDBMS, such as MySQL Cluster or PostgreSQL, which can improve performance and scalability by distributing data across multiple servers.

3.1.4 API management:

Amazon API Gateway will be used for API management. Amazon API Gateway is an AWS service for creating, publishing, maintaining, monitoring, and securing REST, HTTP, and Web-Socket APIs at any scale. API developers can create APIs that access AWS or other web services, as well as data stored in the AWS Cloud. In UWC 2.0, Amazon API Gateway can be used to manage the interfaces between different components of the system. For example the communication between the Task Management module and the existing UWC 1.0 database, as well as the integration with external services like Google Calendar, Google Maps, and Google Routes API for the calendar viewer, map viewer and route management.

3.2 Draw a component diagram for the Task Assignment module



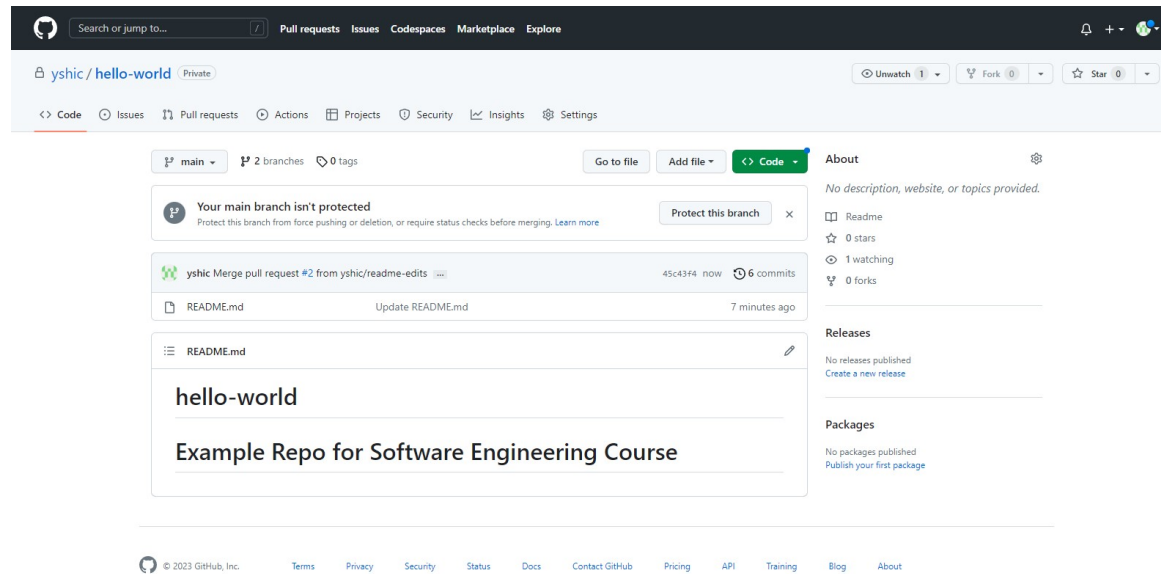


4 Task 4: Implementation - Sprint 1

4.1 Setting up an online repository (github, bitbucket, etc) for version control.

Github will be used for version control.

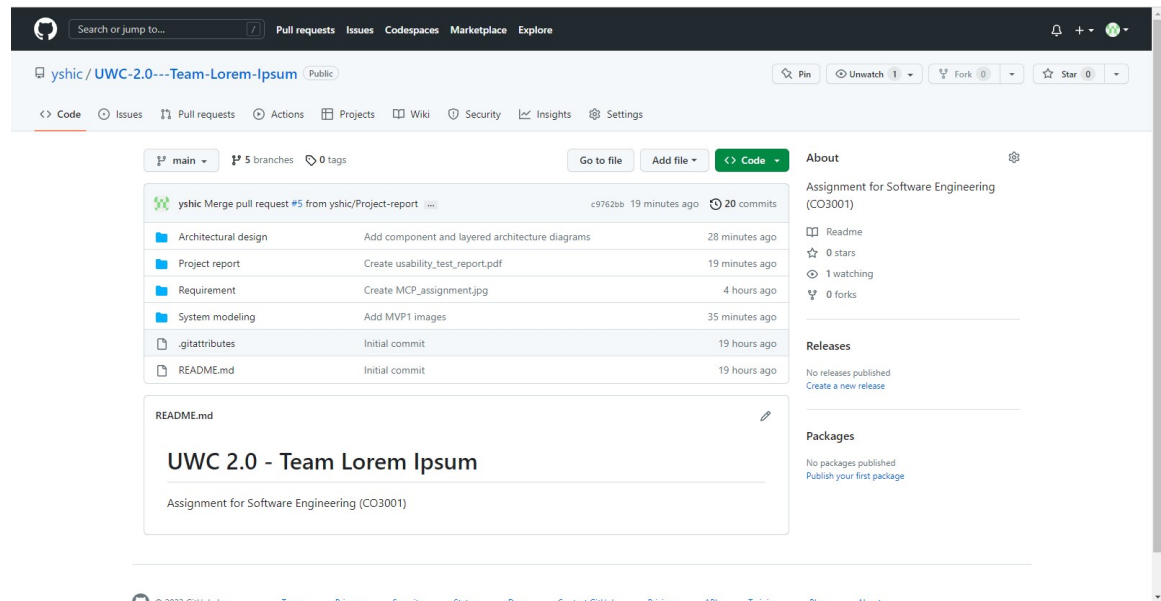
[Hello world example repository](#)





4.2 Adding documents, materials and folders for Requirement, System modelling and Architectural design. Use the selected version control system to report the changes to these files

Github repository for UWC 2.0



4.3 Conducted a usability test with the user interface you developed in MVP1.

The test report is included with the UWC 2.0 report as a separate file.