FPT UNIVERSITY

Assignment 1
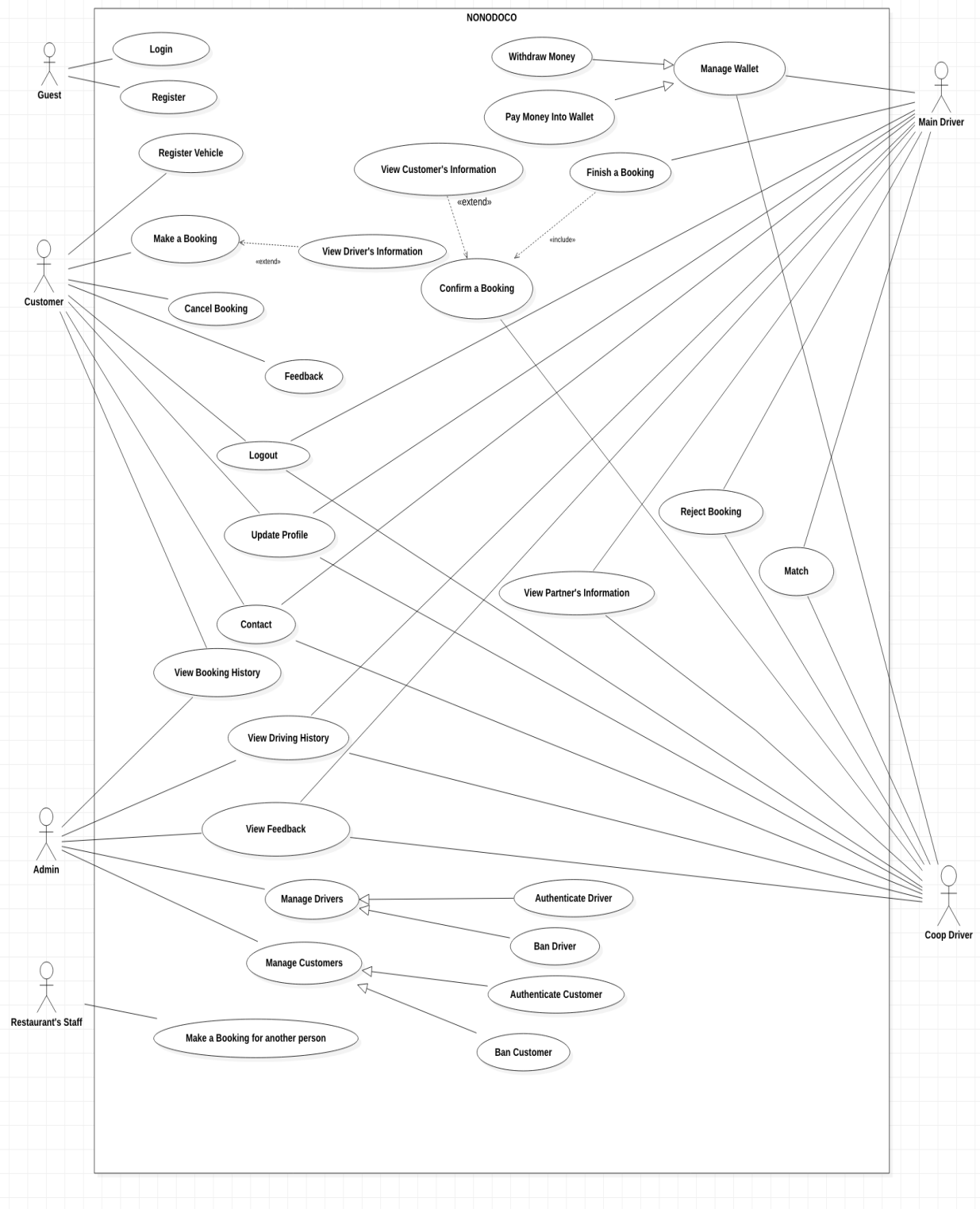
NoNODOCO Application

Group 2	
Group members	Nguyen Ngoc Quynh Nhu – SE140857
	Phan Huynh Dang Khoa – SE140609
	Bui Pham Minh Nhat – SE130488
	Pham Duc Hoang – SE130719
	Nguyen Duc Linh – SE130639
Supervisor	Mr. Lai Duc Hung

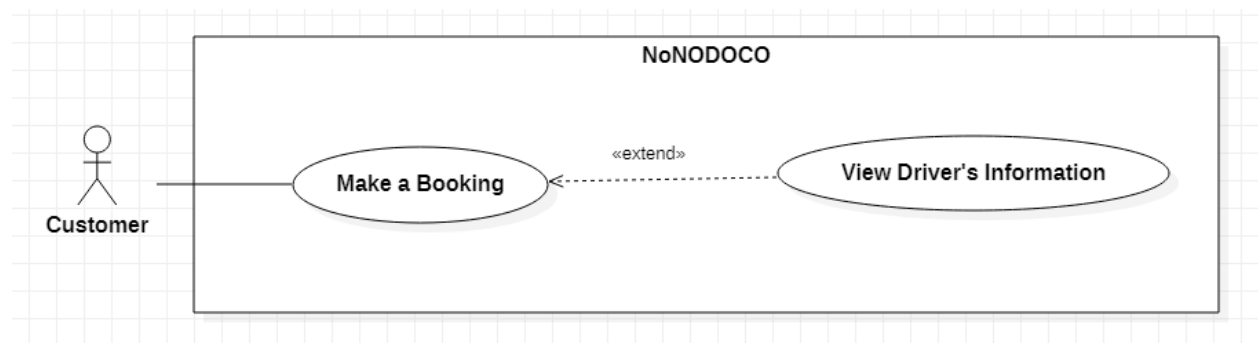-Ho Chi Minh City, 22/07/2020-

Table of Contents

I. Use Case Overview

II. Use Case Specification

1. <Customer> Make a Booking

Use Case Diagram

Use Case Specification

USE CASE - UC001			
Use Case No.	**UC001**	**Use Case Version**	1.0
Use Case Name	Make a Booking		
Author	Phan Huynh Dang Khoa		
Date	14/06/2020	**Priority**	High

Actor:
- Customer

Summary:
- This use case allows customers to make a booking.

Goal:
- A driver is dispatched to pick up the customer.

Triggers:
- On main screen, the customer clicks on the location search box.

Preconditions:
- User logged in as Customer.

Post Conditions:
- A driver is dispatched to pick up the customer.
- Booking's details are shown on the customer's screen.

Normal flow:

Step	Actor Action	System Response
1	Customer inputs the start location to the "From" box.	System marks the start location on the map. [Exception 1]
2	Customer inputs the end location to the "To" box.	System marks the end location on the map. [Exception 1]
3	Customer chooses a vehicle from the registration list.	System shows the distance and price for the trip. [Exception 2]
4	Customer confirms the booking, then clicks on the "Start booking" button.	- System searches for nearby drivers and waits until one accepts the request. [Exception 3] - When a driver accepts the request, the system dispatches that driver to the customer. - System displays the booking's details on the customer's screen.

Alternative flow:
    <N/A>

Exceptions:

No.	Exception Detail	System Response
1	Customer inputs a location that does not exist.	System shows message "The location cannot be found".
2	No vehicle is available in the registration list.	System shows message "You must register a vehicle to make a booking".
3	No driver near the customer is available.	System shows message "No driver is available".

Relationship:
- Extends to "View Driver's Information"

Business Rules:
- Price's calculation for a booking:
    - Motorbikes:
        - Under 1km: 20.000 VND
        - From 1km onwards: Distance (minus the first km) * 15.000 VND
    - Cars:
        - Under 1km: 30.000 VND
        - From 1km onwards: Distance (minus the first km) * 22.000 VND
- When searching for a driver, system searches within the radius of 2km.
- When a driver has accepted the booking, the booking is stored in the database with the status "New".

2. <Customer> Register Vehicle

Use Case Diagram

Use Case Specification

USE CASE - UC002			
Use Case No.	**UC002**	**Use Case Version**	1.0
Use Case Name	Register Vehicle		
Author	Nguyen Ngoc Quynh Nhu		
Date	14/06/2020	**Priority**	High

Actor:
- Customer

Summary:
- This use case allows customers to register a vehicle.

Goal:
- A vehicle is registered.

Triggers:
- Customer clicks on the "Register Vehicle" button.

Preconditions:
- User logged in as Customer.

Post Conditions:
- Registered vehicle's information is stored in the database.
- System shows a successful message with the registered vehicle's information.

Normal flow:

Step	Actor Action	System Response
1	Customer clicks on the "Register Vehicle" button.	System displays a form for the customer to fill in the following information: • Vehicle's brand (required) • Vehicle's color (required) • License plate (required) • Vehicle's insurance number (required) • Vehicle's registration certificate number (required)
2	Customer fills in the form and clicks on the "Submit" button.	- System verifies the form. [Exception 1] - System stores the vehicle's information in the database. - System shows the message "Vehicle was registered successfully" with the vehicle's information on the customer's screen.

Alternative flow:
　　<N/A>

Exceptions:

No.	Exception Detail	System Response
1	One of the required form's fields is not filled in.	System shows message on the corresponding field "This field is required".

Relationship:
　　<N/A>

Business Rules:
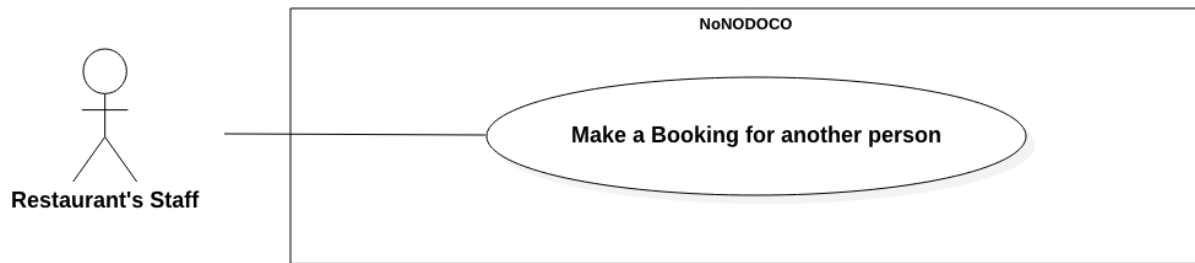- After a customer has successfully registered a vehicle, the status of the vehicle is "Verifying".
- When the admin has verified the vehicle and it is a new one, the status of the vehicle is set to "Active".
- When the admin has verified the vehicle and it already existed in the database but is

owned by a new customer, the status of the vehicle is changed from "Inactive" to "Active".

3. <Restaurant's staff> Make a Booking for another person

Use Case Diagram

Use Case Specification

USE CASE - UC003			
Use Case No.	**UC003**	**Use Case Version**	1.0
Use Case Name	Make a Booking for another person		
Author	Pham Duc Hoang		
Date	14/06/2020	**Priority**	Medium

Actor:
- Restaurant's staff

Summary:
- This use case allows restaurant's staff to make a booking for a customer.

Goal:
- A driver is dispatched to pick up the customer.

Triggers:
- On main screen, the restaurant's staff clicks on the location search box.

Preconditions:
- User logged in as Restaurant's staff.

Post Conditions:
- A driver is dispatched to pick up the customer.
- Booking's details are shown on the staff's screen.

Normal flow:

Step	Actor Action	System Response
1	Restaurant's staff inputs the start location to the "From" box.	System marks the start location on the map. [Exception 1]
2	Restaurant's staff inputs the end location to the "To" box.	System marks the end location on the map. [Exception 1] System displays a form for the staff to fill in the following information: • Customer's name (required) • Customer's phone number (required) • Vehicle's type (required) • Vehicle's brand (required) • Vehicle's color (required) • License plate (required)
3	Restaurant's staff fills in the form and tap "Submit" button.	System verifies the form. [Exception 2] System shows the distance and price for the trip.
4	Restaurant's staff confirms the booking, then clicks on the "Start booking" button.	- System searches for nearby drivers and waits until one accepts the request. [Exception 3] - When a driver accepts the request, the system dispatches that driver to the customer. - System displays the booking's details on the staff's screen.

Alternative flow:
    <N/A>

Exceptions:

No.	Exception Detail	System Response

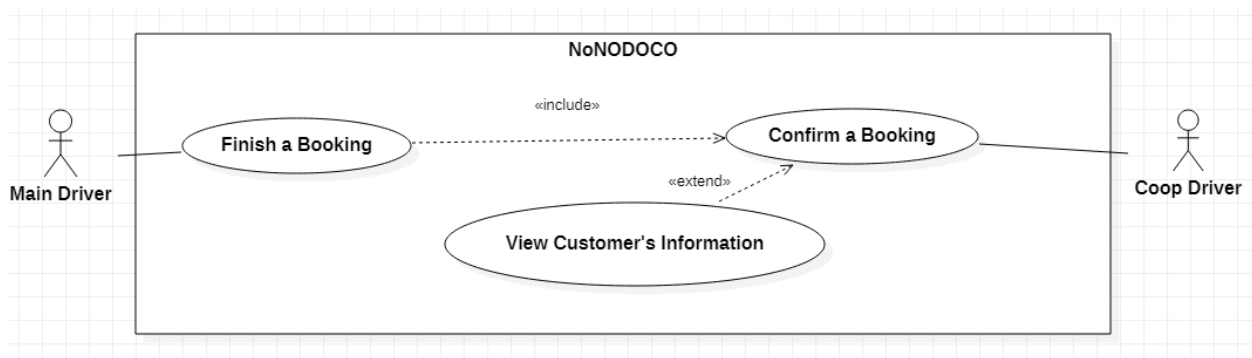|		
1	Restaurant's staff inputs a location that does not exist.	System shows message "The location cannot be found".
2	One of the required form's fields is not filled in.	System shows message on the corresponding field "This field is required".
3	No driver near the customer is available.	System shows message "No driver is available".

Relationship:
    <N/A>

Business Rules:
- Price's calculation for a booking:
  - Motorbikes:
    - Under 1km: 20.000 VND
    - From 1km onwards: Distance (minus the first km) * 15.000 VND
  - Cars:
    - Under 1km: 30.000 VND
    - From 1km onwards: Distance (minus the first km) * 22.000 VND
- When searching for a driver, system searches within the radius of 2km.
- When a driver has accepted the booking, the booking is stored in the database with the status "New". If the vehicle does not exist in the database, it is also stored with the status "Booked by Restaurant".

4. <Main Driver, Coop Driver> Confirm a Booking

Use Case Diagram

Use Case Specification

USE CASE - UC004			
Use Case No.	**UC004**	**Use Case Version**	1.0
Use Case Name	Confirm a Booking		
Author	Nguyen Duc Linh		
Date	14/06/2020	**Priority**	High

Actor:
- Main Driver, Coop Driver

Summary:
- This use case allows the main driver and the coop driver to confirm a booking.

Goal:
- The drivers accept the booking and pick up the customer.

Triggers:
- A booking is sent to the main driver and coop driver.

Preconditions:
- User logged in as Main Driver/Coop Driver.
- The two drivers have matched.
- The status of both drivers is Ready.

Post Conditions:
- Booking's details are shown on both drivers' screens.

Normal flow:

Step	Actor Action	System Response
1	When the booking's request is shown on the screen, Main driver and Coop driver click on the "Confirm" button.	System shows the booking's details and money that each driver receives on both drivers' screens. [Exception 1, 2] System shows the message "Driver is found" on the customer's screen.

Alternative flow:
    <N/A>

Exceptions:

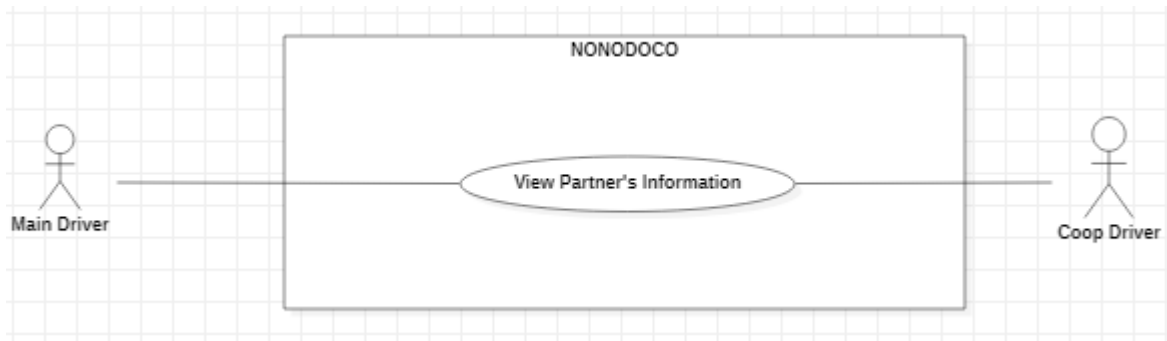| No.	Exception Detail	System Response
1	One of the drivers did not click on the "Confirm" button.	System show message "Your partner is not ready".
2	The booking is cancelled by the customer.	System show message "Booking is cancelled".

Relationship:
- Extends to "View Customer's Information"
- Included by "Finish a Booking"

Business Rules:
- The money that each driver will receive:
  - Main driver: 60%
  - Coop driver: 40%
- When a driver has accepted the booking, the booking is stored in the database with the status "New".

5. <Main Driver, Coop Driver> View Partner's Information

Use Case Diagram

Use Case Specification

USE CASE - UC005			
Use Case No.	**UC005**	**Use Case Version**	1.0
Use Case Name	View Partner's Information		
Author	Nguyen Duc Linh		
Date	14/06/2020	**Priority**	Low

Actor:
- Main Driver, Coop Driver

Summary:
- This use case allows the main driver and the coop driver to view partner's information.

Goal:
- Each driver can view the partner's information.

Triggers:
- Main driver/Coop driver clicks on the "View Partner's Information".

Preconditions:
- User logged in as Main Driver/Coop Driver.
- The two drivers have matched.

Post Conditions:
- The partner's information is shown on the driver's screen.

Normal flow:

Step	Actor Action	System Response
1	Driver click on the "View Partner's Information" button.	System shows the partner's information: ▪ Partner's name ▪ License plate of coop driver (main driver only) ▪ Vehicle's type of coop driver (main driver only) ▪ Average star rating ▪ Status [Exception 1]

Alternative flow:
<N/A>

Exceptions:

No.	Exception Detail	System Response
1	Partner cancelled the match before viewing.	System show message "Partner cancelled matching".

Relationship:
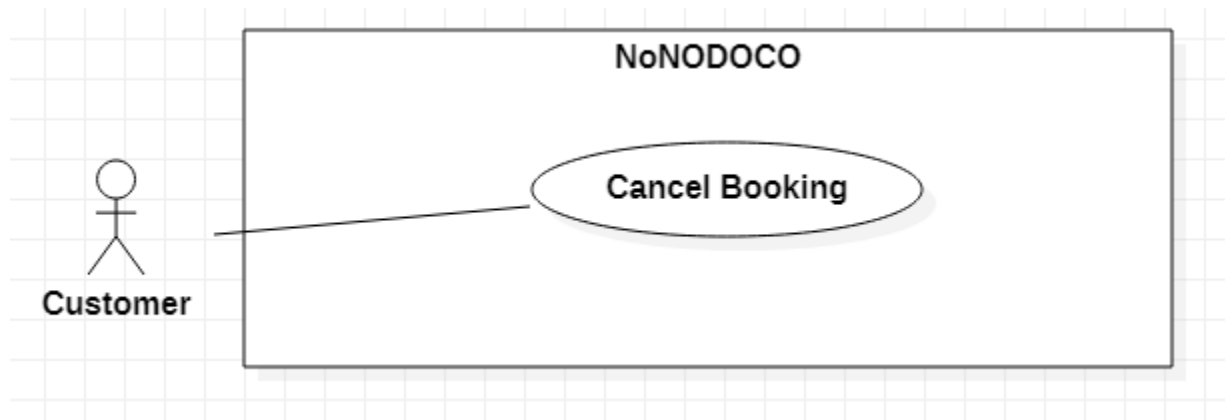<N/A>

Business Rules:
- The driver can only view the partner's information if the status of the matching is "Active".

6. <Customer> Cancel Booking

Use Case Diagram

Use Case Specification

USE CASE - UC006			
Use Case No.	**UC006**	**Use Case Version**	1.0
Use Case Name	Cancel Booking		
Author	Bui Pham Minh Nhat		
Date	14/06/2020	**Priority**	High

Actor:
- Customer

Summary:
- This use case allows customer to cancel the booking that is in process.

Goal:
- Booking is cancelled.

Triggers:
- Customer clicks on the "Cancel booking" button.

Preconditions:
- User logged in as Customer.
- User already made a booking.

Post Conditions:
- System shows message indicating that booking is cancelled.

Normal flow:

Step	Actor Action	System Response
1	Customer clicks on the "Cancel Booking" button.	- System shows confirmed message "Are you sure to cancel this booking?".
2	Customer chooses Yes or No.	System acts based on customer's choice: + "Yes": System will cancel the booking. + "No": System will continue the booking process.

Alternative flow:
      <N/A>

Exceptions:
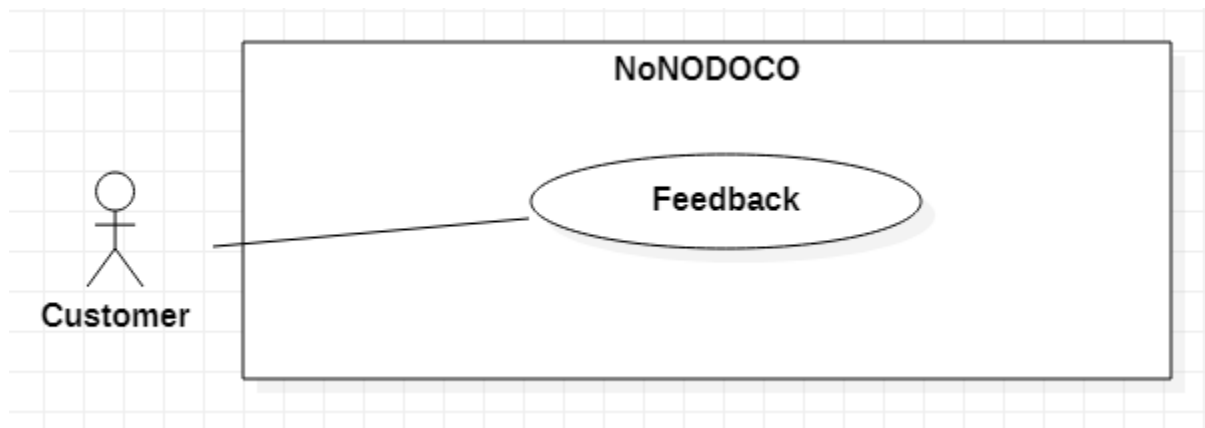   <N/A>

Relationship:
      <N/A>

Business Rules:
- The customer can only cancel the booking when the status of the booking is still "New".
- When the driver arrives at the location, the "Cancel Booking" button is disabled.
- If the customer cancels the booking, the status of the booking is changed from "New" to "Cancelled".

7. <Customer> Feedback

Use Case Diagram

NoNODOCO

Feedback

Customer

Use Case Specification

USE CASE - UC007			
Use Case No.	**UC007**	**Use Case Version**	1.0
Use Case Name	Feedback		
Author	Bui Pham Minh Nhat		
Date	14/06/2020	**Priority**	Medium

Actor:
- Customer

Summary:
- This use case allows customer to give feedback about their booking.

Goal:
- Customer can give feedback successfully.

Triggers:
- Customer clicks on the "Feedback" button.

Preconditions:
- User logged in as Customer.
- Booking has finished.

Post Conditions:
- Feedback is stored in the database.

Normal flow:

Step	Actor Action	System Response
1	Customer clicks on the "Feedback" button.	System show feedback form for customer to fill in: ■ Star rating (required) ■ Comment (no more than 250 characters)
2	Customer fills in the form and clicks on the "Submit" button.	System verifies the form. [Exception 1, 2] System stores the feedback to the database.

Alternative flow:
      <N/A>

Exceptions:

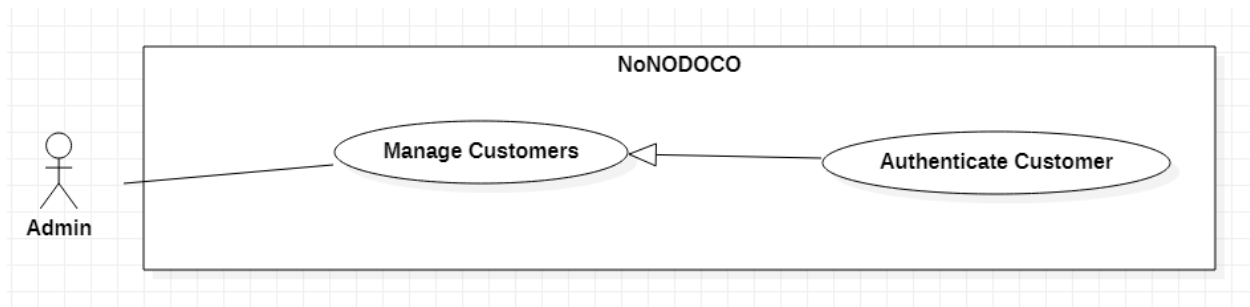| No.	Exception Detail	System Response
1	Customer did not rate based on stars.	System shows message "Star rating is required".
2	Customer inputs comment that exceeds 250 characters.	System shows message "Comment must not exceed 250 characters".

Relationship:
      <N/A>

Business Rules:
■ Feedback of customer does not need to provide accurate data in textbox.

8. <Admin> Authenticate Customer

Use Case Diagram

Use Case Specification

USE CASE - UC008			
Use Case No.	**UC008**	**Use Case Version**	1.0
Use Case Name	Authenticate Customer		
Author	Phan Huynh Dang Khoa		
Date	14/06/2020	**Priority**	Medium

Actor:
- Admin

Summary:
- This use case allows admin to authenticate customers.

Goal:
- The customer is authenticated.

Triggers:
- On the admin's screen, the admin clicks on the "Authenticate" button after viewing the customer's information.

Preconditions:
- Customer has made at least one successful booking.

Post Conditions:
- System shows message indicating that customer is authenticated.

Normal flow:

Step	Actor Action	System Response
1	Admin clicks on "Customers to be authenticated" button	System shows summarized information of all customers who need to be authenticated. Summarized information includes: o Customer ID o Full Name
2	Admin clicks on "Details" button to view a particular customer's details.	System shows details of a particular customer. Details include: o Customer ID o Full Name o Phone Number o Email o Address
3	Admin clicks on "Authenticate" button	System updates the status of this customer in database into "Active". System redirects the admin to the main screen and reload the list of customers to be authenticated.

Alternative flow:
      <N/A>

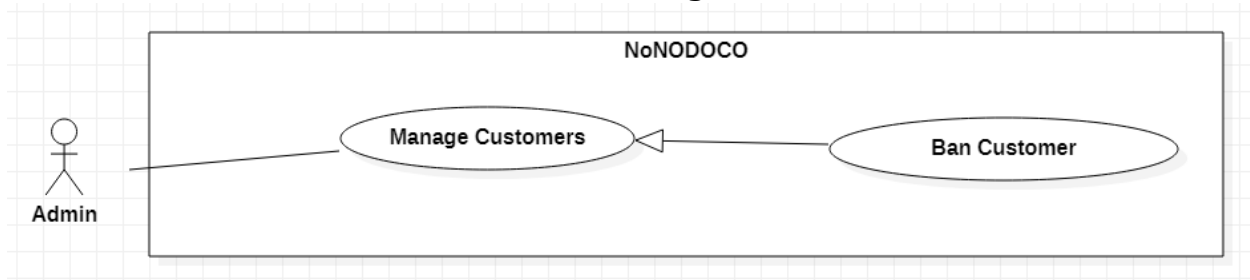**Exceptions:**
      <N/A>

Relationship:
- Specializes for "Manage Customers"

Business Rules:
- When a new customer registers, the information is stored in the database with the status "New".
- When the admin authenticates that customer, the status is changed from "New" to "Active".

9. <Admin> Ban Customer

Use Case Diagram

Use Case Specification

USE CASE - UC009			
Use Case No.	**UC009**	**Use Case Version**	1.0
Use Case Name	Ban Customer		
Author	Nguyen Ngoc Quynh Nhu		
Date	14/06/2020	**Priority**	Medium

Actor:
- Admin

Summary:
- This use case allows admin to ban customers.

Goal:
- The customer is banned.

Triggers:
- On the admin's screen, the admin clicks on the "Ban" button after viewing the customer's information.

Preconditions:
- The customer is not banned.

Post Conditions:
- System shows message indicating that customer is banned.

Normal flow:

Step	Actor Action	System Response
1	Admin clicks on "Show All Customers" button	System shows summarized information of all customers. Summarized information includes: ○ Customer ID ○ Full Name ○ Number of cancelled bookings
2	Admin clicks on "Details" button to view a particular customer's details.	System shows details of a particular customer. Details include: ○ Customer ID ○ Full Name ○ Phone Number ○ Email ○ Address ○ Number of cancelled bookings
3	Admin clicks on "Ban" button	System updates the status of this user in database into "Banned". System redirects the admin to the main screen and reload the list of customers.

Alternative flow:
   <N/A>

Exceptions:
   <N/A>

Relationship:
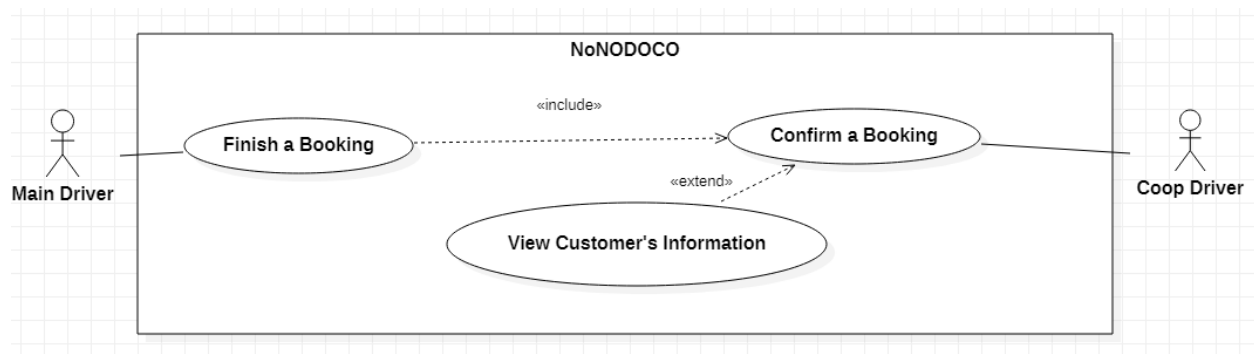- Specializes for "Manage Customers"

Business Rules:
- When the system gets the list of customers, the system only selects the customers that has not been banned.

10. <Main Driver> Finish a Booking

Use Case Diagram

NoNODOCO

Main Driver — Finish a Booking ···«include»···> Confirm a Booking — Coop Driver

«extend»

View Customer's Information

Use Case Specification

USE CASE - UC010			
Use Case No.	**UC010**	**Use Case Version**	1.0
Use Case Name	Finish a Booking		
Author	Pham Duc Hoang		
Date	14/06/2020	**Priority**	High

Actor:
- Main Driver

Summary:
- This use case allows the main driver to finish a booking.

Goal:
- The booking is finished and the main driver drops off the customer.

Triggers:
- Main driver clicks on the "Pick up" button.

Preconditions:
- User logged in as Main driver.
- The booking is not cancelled by the customer.
- The drivers have confirmed the booking.

Post Conditions:
- System shows message indicating that booking is finished.

Normal flow:

Step	Actor Action	System Response
1	Driver begins taking the customer to the destination.	System updates the status of booking from "New" to "Process". System continuously updates the driver location on the map.
2	Driver arrives at the destination and clicks on the "Drop off" button.	System updates the status of booking from "Process" to "Success". System shows the bill of the booking on both the customer's screen and the driver's screen.
3	Driver clicks on the "OK" button.	System redirects to the main screen.

Alternative flow:
        \<N/A>

**Exceptions:**
        \<N/A>

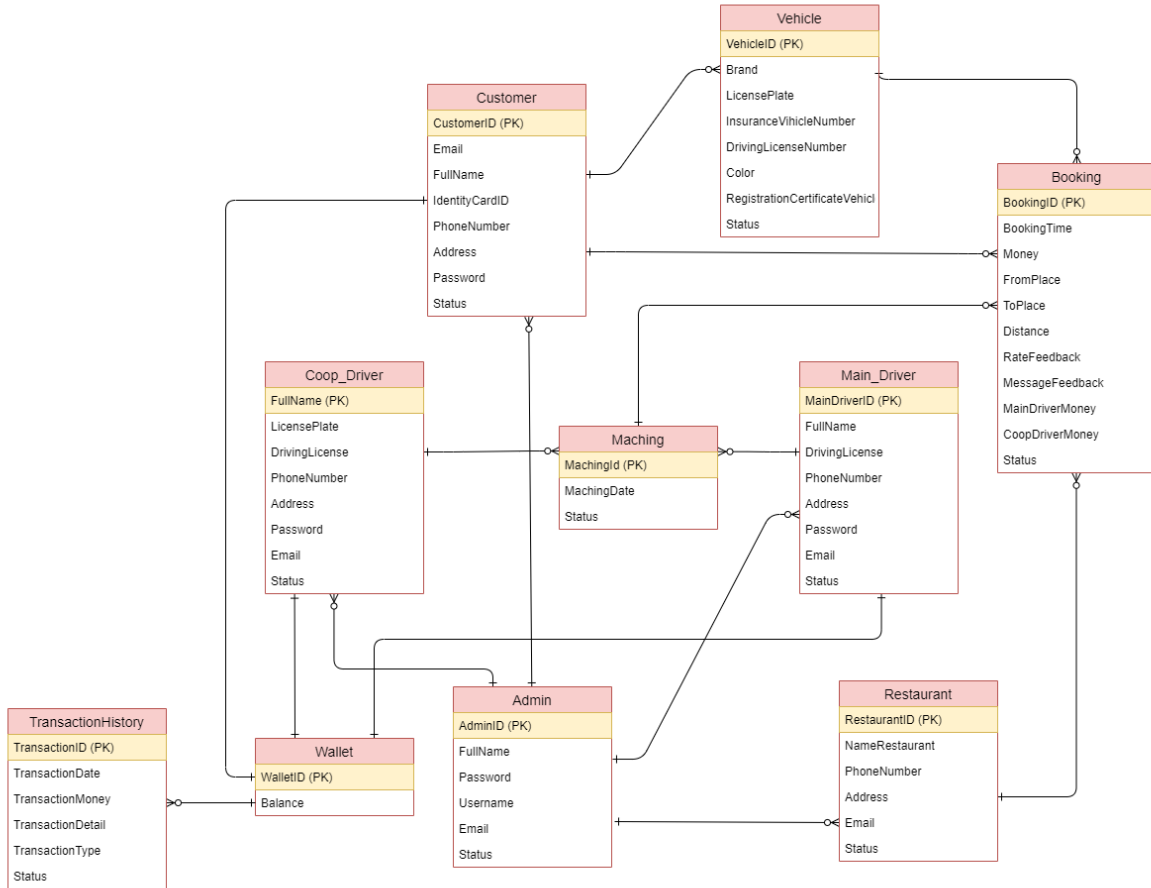**Relationship:**
- Includes "Confirm a Booking"

Business Rules:
- The money that each driver will receive:
  - Main driver: 60%
  - Coop driver: 40%
- When the driver goes back to the main screen, the money of his wallet is updated according to the received money.
- If the customer pays in cash, system will subtract 20% (of the money each driver receives) FROM the wallet of each driver.
- If the customer pays through wallet, the money is subtracted from the customer's wallet to the drivers' wallets (based on the ratio of 60/40). System will subtract 20% (of the money each driver will receive) from both drivers' wallets.

III. Conceptual Diagram
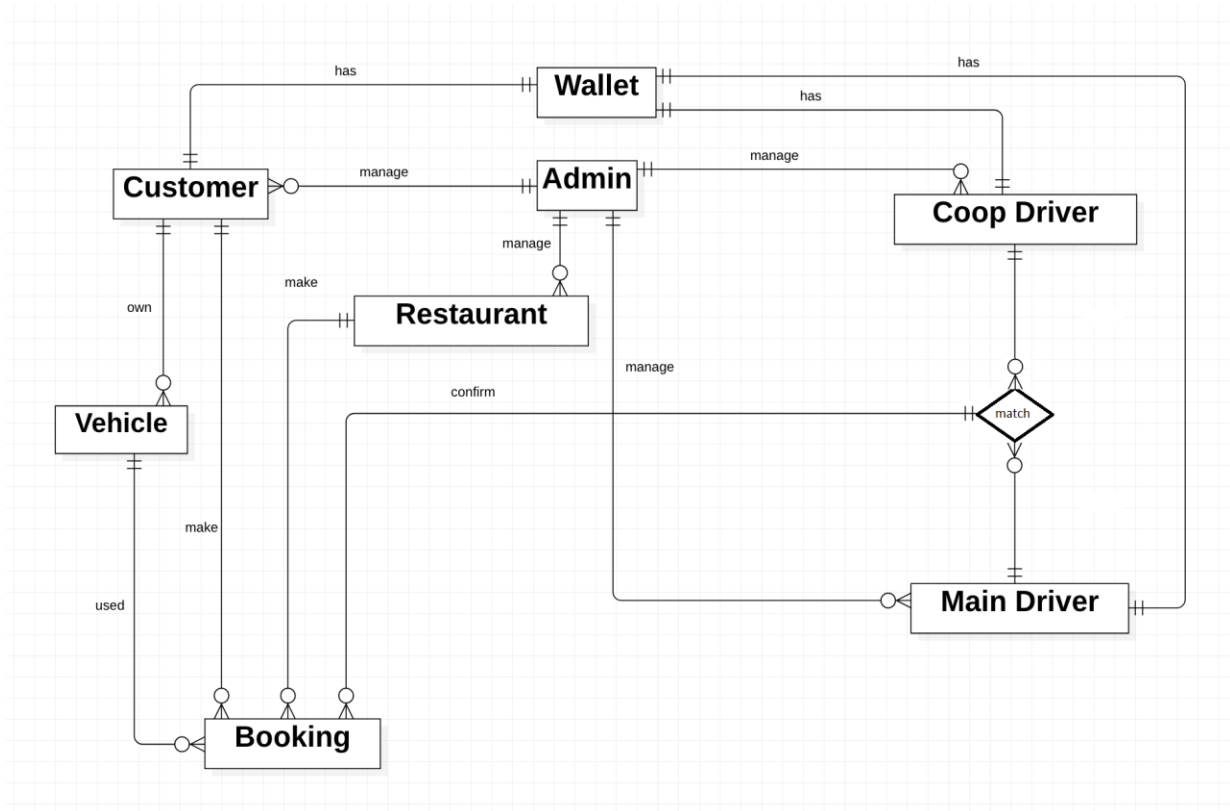
IV. Entity Relationship Diagram

V. Database Diagram