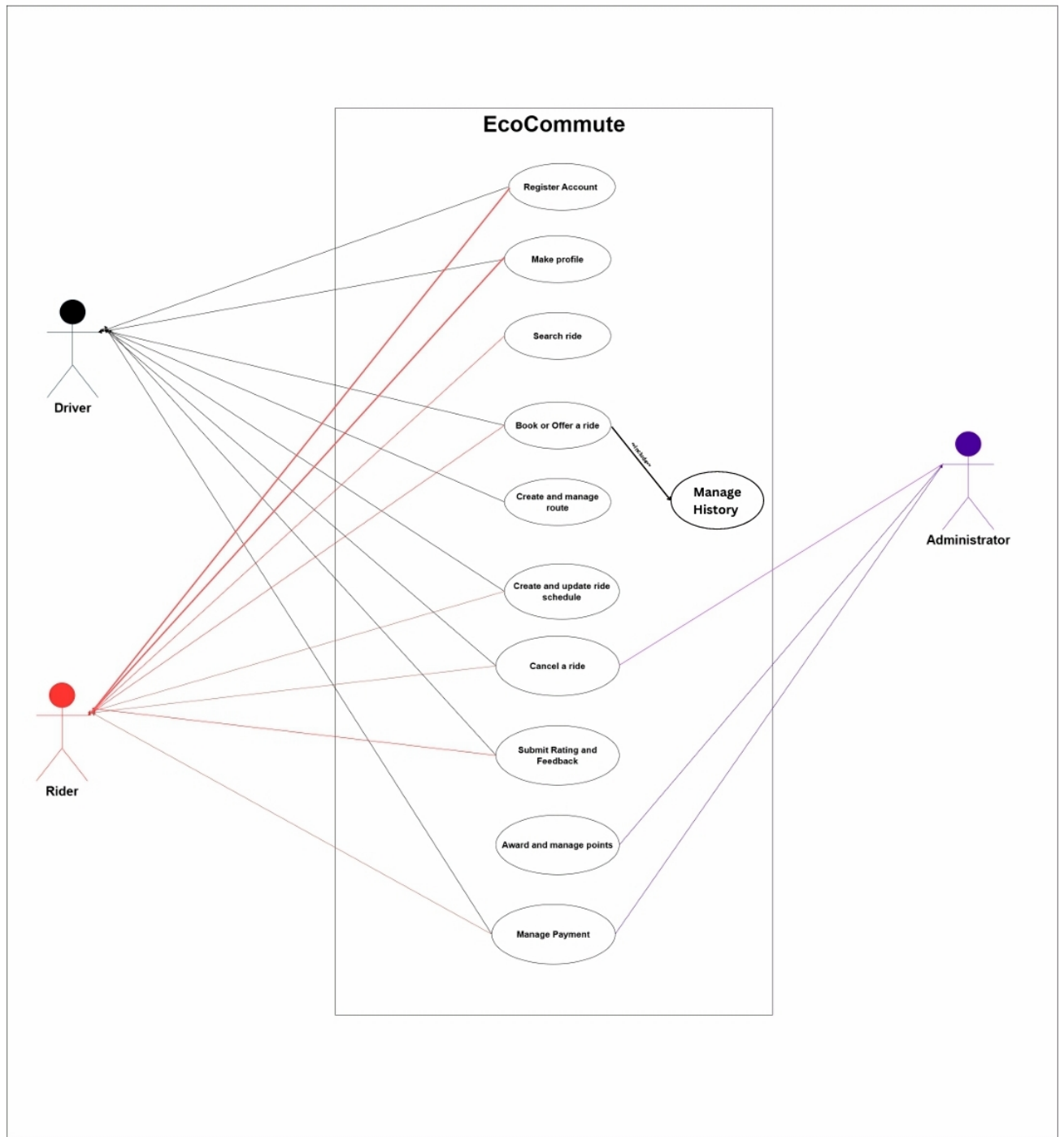
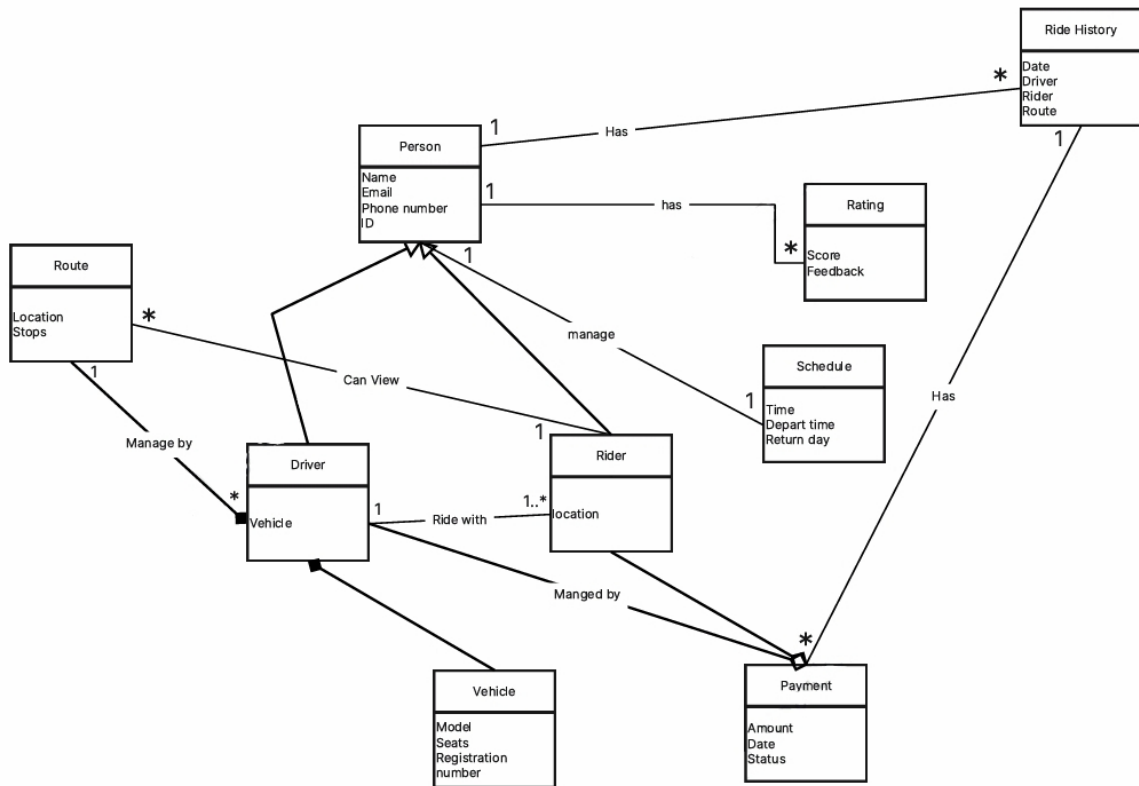


EcoCommute

22F-8813 Muhammad Fawwaz

22F-3356 faraz ul khaf





1. Create Profile (Driver)

Identifier	UC001		
Name	Create Driver Profile		
Purpose	To enable drivers to create a profile, add car details, routes, and schedules.		
Priority	High		
Actors	Driver		
Preconditions	Driver has opened the app, registered account, ready for profile setup.		
Postconditions	Driver profile is created and stored in the system. Car details, routes, and schedules are saved.		
Dependencies	Network connectivity, Valid phone number, System to store data		
Typical course of actions (Primary Role: Driver)			
Actor Action		System Response	
1. Driver selects "Create Profile".		1 System displays registration form.	
2. Driver enters personal information (Name, ID, Phone, Email).		2 System validates and saves details.	
3. Driver enters car details (Model, Registration, Capacity).		3 System verifies and saves car details.	
4. Driver selects "Add Route".		4 System prompts for start, end, and stop locations.	
5. Driver specifies route locations.		5 System verifies and saves route.	
Alternative course of actions			
Actor Action		System Response	
1. Driver skips car details.		1 System prompts to complete later.	
2. Driver skips route details.		2 System allows saving basic profile, prompts to complete route later.	
Exceptions			
1. Driver enters invalid car details. System prompts error for corrections.			
2. Driver leaves required fields blank. System prevents submission and highlights missing fields.			

2: Create profile (Rider)

Identifier	UC002	
Name	Create Rider Profile	
Purpose	To allow riders to create a profile and set their location and schedule preferences.	
Priority	High	
Actors	Rider	

Preconditions	Rider has opened the app, registered account, ready for profile setup.
Postconditions	Rider profile is created and saved in the system. Location and schedules are stored.
Dependencies	Network connectivity, Valid phone number, System to store data
Typical course of actions (Primary Role: Rider)	
Actor Action	System Response
1. Rider selects "Create Profile".	1 System displays registration form.
2. Rider enters personal information (Name, ID, Phone, Email).	2 System validates and saves details.
3. Rider selects "Add Location".	3 System prompts for location details.
4. Rider specifies home/university locations.	4 System verifies and saves location.
5. Rider selects "Add Schedule".	5 System prompts for days and times.
6. Rider specifies schedule.	6 System verifies and saves schedule.
7. Rider reviews and submits the profile.	7 System confirms profile creation and displays success message.
Alternative course of actions	
Actor Action	System Response
1. Rider skips location details.	1 System allows saving basic profile, prompts to add location later.
2. Rider skips schedule.	1 System allows rider to proceed without schedule and prompts to complete later.
Exceptions	
1. Rider enters invalid location. System prompts error for correction. 2. Rider leaves required fields blank. System prevents submission and highlights missing fields.	

3: Search for ride

Identifier	UC003
-------------------	-------

Name	Search for a Ride	
Purpose	To allow riders to search for available rides based on location and schedule.	
Priority	High	
Actors	Rider	
Preconditions	Rider has created a profile and is logged into the system.	
Postconditions	Rider receives a list of rides that match their search criteria.	
Dependencies	Active network connection, accurate driver routes and schedules in the system	
Typical course of actions (Primary Role: Rider)		
Actor Action	System Response	
1. Rider selects the "Search Ride" option.	1 System prompts rider to enter location.	
2. Rider enters current or preferred location.	2 System retrieves and displays available routes passing through that location.	
3. Rider selects route of interest.	3 System displays ride details, including schedules and driver information.	
4. Rider filters available schedules by preferred time range.	4 System updates the results to show only matching schedules.	
5. Rider selects a specific ride.	5 System displays final ride details and booking options.	
Alternative course of actions		
Actor Action	System Response	
1. Rider skips location input.	1.1 System retrieves default location based on GPS or previously saved location.	
2. Rider does not apply schedule filters.	1.2 System shows all available rides for the entered location.	
Exceptions		
No Available Rides: If no rides match the entered location or schedule, the system informs the rider and suggests trying another location or time.		
Invalid Location: The rider enters a location that is not recognized or does not have any routes. The system prompts the rider to enter a valid location.		
System Timeout: The system fails to retrieve ride data due to network issues. An error message is displayed, and the rider is asked to retry later.		

4: Book or Offer Ride

Identifier	UC004
-------------------	-------

Name	Book or Offer a Ride	
Purpose	To allow a rider to request a ride from a driver, confirm availability, and book a seat.	
Priority	High	
Actors	Rider, Driver	
Preconditions	Rider has searched for a ride and selected a route; Driver has an available seat.	
Postconditions	Ride is booked, rider and driver are registered for the trip, and seat availability is updated.	
Dependencies	Active network connection, available seats, system to track booking and seat status, messaging system for chat between rider and driver.	
Typical Course of Action (Primary Role: Rider and Driver)		
Actor Action	System Response	
1. Rider selects a ride from the search results.	1 System displays detailed ride info, including available seats, driver info, and car details.	
2. Rider clicks "Request Ride".	2 System notifies driver of the booking request.	
3. Driver receives request and reviews the details.	3 System prompts driver to accept or decline the request.	
4. Driver accepts the ride request.	4 System sends confirmation to the rider.	
5. Rider and driver open the chat window (if needed) to communicate about pickup details.	5 System enables chat between the rider and driver.	
6. Rider confirms the booking.	6 System registers the rider, updates seat count, and marks the ride as booked.	
7. Driver receives the final booking confirmation.	7 System updates the driver's schedule with the new rider and adjusts availability.	
Alternative Course of Action		
Actor Action	System Response	
1. Driver declines the ride request.	1 System notifies the rider and prompts them to search for another ride.	
2. Rider sends a message for clarification.	2 Driver responds, and system updates the chat log.	
Exceptions		
Driver Busy or Unavailable: The driver is not able to accept the request. The system informs the rider and suggests trying another driver.		
No Available Seats: The requested ride has no more available seats. The system notifies the rider that the ride is full.		
Booking Conflict: Another rider books the seat before the current rider's request is confirmed. The system informs the rider of the conflict and prompts them to search for other rides.		
Communication Error: If the messaging system fails, the rider and driver are notified.		

5: Create and Manage Route

Identifier	UC005
Name	Create and Manage Route
Purpose	To allow drivers to create, update, and manage their routes by adding or modifying stops as they pick up riders.

Priority	High
Actors	Driver
Preconditions	Driver has created a profile and added their car information.
Postconditions	Driver's route is created and updated as they add stops or modify existing ones, and the route is visible to potential riders.
Dependencies	location services, active network connection, ride requests from riders.
Typical Course of Action (Primary Role: Driver)	
Actor Action	System Response
1. Driver selects "Create Route" option.	1. System prompts the driver to add the starting location and destination.
2. Driver enters the starting location, destination, and route stops.	2. System saves the locations and displays a map of the route.
3. Driver reviews and confirms the route.	3. System confirms the route creation and makes it visible to riders.
4. Driver updates route while driving (e.g., adds a new stop).	4. System allows modifications to the route and updates it in real-time.
Alternative Course of Action	
Actor Action	System Response
1. Driver adds only starting point and destination (no stops).	1. System automatically generates the default route between these points.
2. Driver temporarily pauses the route.	2. System marks the route as temporarily unavailable to new riders.
Exceptions	
<p>Invalid Location: Driver enters a location that the system does not recognize. The system prompts the driver to correct the location.</p> <p>Network Failure: The system is unable to update or create the route due to connectivity issues. The driver is notified, and the system suggests retrying later.</p>	

6: Create and Update Ride Schedule

Identifier	UC006
Name	Create and Update Ride Schedule
Purpose	To allow both riders and drivers to create and update their ride schedules, which are stored and linked to their profiles.

Priority	High
Actors	Person (Rider or Driver)
Preconditions	Rider/Driver has created a profile.
Postconditions	Ride schedule is created or updated and saved in the profile of the rider/driver.
Dependencies	Profile class, system's scheduling functionality, calendar system, network connection.
Typical Course of Action (Primary Role: Rider and Driver)	
Actor Action	System Response
1. Rider/Driver selects "Create Schedule" option.	1. System prompts user to enter ride timing and location details.
2. Rider/Driver enters start time, end time, and preferred pickup locations (Rider) or route stops (Driver).	2. System saves the schedule and links it to the user's profile.
3. Rider/Driver reviews and confirms the schedule.	3. System confirms the schedule creation and displays it in their profile.
4. Rider/Driver updates an existing schedule (e.g., changing ride timing).	4. System modifies and updates the schedule in real-time.
5. Rider/Driver can view the schedule in their profile.	5. System retrieves and displays the updated schedule in their profile.
Alternative Course of Action	
Actor Action	System Response
1. Rider/Driver only enters partial schedule details.	1. System prompts the user to complete missing information before saving.
2. Rider/Driver decides to pause or deactivate a schedule.	2. System marks the schedule as inactive but keeps it stored for later use.
Exceptions	
<p>Invalid Time Format: The system detects an invalid time format and prompts the user to correct the time.</p> <p>Network Error: If the system cannot save or update the schedule due to network issues, an error is displayed, and the user is advised to retry later.</p>	

7: Cancel a ride

Identifier	UC007
Name	Cancel a Ride
Purpose	To allow both drivers and riders to cancel a ride, while the system records cancellations for review by the admin.
Priority	High

Actors	Rider, Driver, Admin	
Preconditions	The ride is booked and confirmed.	
Postconditions	Ride is canceled, system logs the cancellation, and admin can review the action.	
Dependencies	Profile class, ride history, admin monitoring system, cancellation log.	
Typical Course of Action (Primary Role: Rider or Driver)		
Actor Action	System Response	
1. Rider/Driver selects "Cancel Ride" option.	1. System prompts user to confirm the cancellation.	
2. Rider/Driver confirms the cancellation.	2. System cancels the ride and updates seat availability (if applicable).	
3. System records the cancellation under the rider/driver's profile.	3. System logs the cancellation in the ride history and flags it for admin review.	
4. Admin is notified to review the cancellation.	4. System generates a report for the admin to review cancellation activity.	
Alternative Course of Action		
Actor Action	System Response	
1. Rider/Driver changes their mind and does not confirm the cancellation.	1. System retains the booking without making changes.	
Exceptions		
Multiple Cancellations: The system detects that the rider or driver has made too many cancellations and prompts for an explanation or notifies them of a potential suspension.		
Admin Suspension: The system places the rider or driver under review after repeated cancellations and alerts them to contact the admin for resolution.		
System Error: If the system fails to cancel the ride, an error is displayed, and the user is advised to try again later.		

8: Submit Rating and Feedback

Identifier	UC008
Name	Submit Feedback and Rating
Purpose	To allow drivers to rate riders and riders to rate drivers, storing the

	ratings for future reference and admin monitoring.
Priority	Medium
Actors	Person (Driver/Rider)
Preconditions	The ride has been completed successfully.
Postconditions	Feedback and rating are stored, and the system may flag it for review by admins.
Dependencies	Profile class, ride history, rating system.
Typical Course of Action (Primary Role: Driver or Rider)	
Actor Action	System Response
1. Driver/Rider selects "Submit Feedback" option.	1. System prompts for a rating and optional feedback text.
2. Driver/Rider enters a rating and provides feedback.	2. System saves the rating and links it to the rider's/driver's profile.
3. Driver/Rider submits the feedback.	3. System confirms submission and updates the overall rating.
4. System logs the feedback for potential admin review.	4. System stores the feedback in ride history and calculates an updated rating average.
Alternative Course of Action	
Actor Action	System Response
1. Driver/Rider provides only a rating (no feedback).	1. System accepts and saves the rating without feedback.
Exceptions	
<p>Invalid Rating Format: If the rating is outside the accepted range, the system prompts the user to submit a valid rating.</p> <p>No Previous Ride: The system detects that the driver/rider has no completed rides with the other party and blocks the feedback submission.</p> <p>Duplicate Feedback: The system detects a duplicate rating or feedback for the same ride and informs the user that the feedback has already been submitted.</p>	

9: Award and manage points

Identifier	UC009
Name	Award and Manage Points
Purpose	To allow the admin to review riders and drivers with good ratings and reward them with points or discounts.
Priority	Medium
Actors	Admin

Preconditions	Riders and drivers have accumulated feedback and ratings.	
Postconditions	Points or rewards are awarded and linked to the profiles of riders/drivers.	
Dependencies	Rating system, reward management system, profile class.	
Typical Course of Action (Primary Role: Admin)		
Actor Action	System Response	
1. Admin logs into the system and selects the "Review Ratings" option.	1. System displays a list of riders and drivers with their ratings.	
2. Admin selects a rider/driver with good ratings.	2. System displays detailed feedback and rating history for that rider/driver.	
3. Admin assigns points or rewards (e.g., discounts).	3. System updates the rider's/driver's profile with the points or rewards.	
4. System notifies the rider/driver of the awarded points or discount.	4. System logs the reward in the history for future tracking.	
Alternative Course of Action		
Actor Action	System Response	
1. Admin sets a threshold for automatic rewards based on ratings.	1. System automatically awards points to riders/drivers who meet the threshold and sends notifications.	
2. Admin revokes previously awarded points (e.g., in case of misuse).	2. System updates the rider's/driver's profile by deducting points and logs the change.	
Exceptions		
.		
System Error During Point Assignment: An error occurs while awarding points, and the admin is notified to retry later.		
.		

10: Manage Payment

Identifier	UC010
Name	Manage Payment
Purpose	To handle the payment process between the driver and rider, allow negotiation if needed, and ensure admin oversight of payments to maintain standard pricing.
Priority	High
Actors	Driver, Rider, Admin

Preconditions	The ride is completed, and a payment is due.	
Postconditions	Payment is processed, receipts are generated, and records are stored in the payment history.	
Dependencies	Payment system, profile class, payment history, admin monitoring system.	
Typical Course of Action (Primary Role: Driver, Rider)		
Actor Action	System Response	
1. Driver initiates the payment offer to the rider.	1. System generates a payment request with the offered amount and sends it to the rider.	
2. Rider accepts the payment or makes a counteroffer.	2. System processes the rider’s response and updates the payment amount if a counteroffer is made.	
3. Driver accepts or adjusts based on rider's counteroffer.	3. System confirms the payment amount after both parties agree.	
4. System generates a receipt for the payment.	4. Payment is recorded in the payment history and linked to both the driver’s and rider’s profiles.	
5. Admin reviews the payment to ensure it is within the standard range.	5. System flags any payments outside the standard range for further admin action.	
Alternative Course of Action		
Actor Action	System Response	
1. Rider requests additional discounts before accepting payment.	1. System alerts the driver for negotiation or informs the rider if discounts are not applicable.	
2. Payment is declined by either party	2. System resets the payment process or alerts admin if necessary.	
Exceptions		
<p>Failed Payment Processing: The payment transaction fails due to technical reasons, and the system prompts both the driver and rider to retry later.</p> <p>Overpayment: If a rider or driver attempts to pay or charge an excessive amount, the system blocks the transaction and alerts the admin.</p>		

