

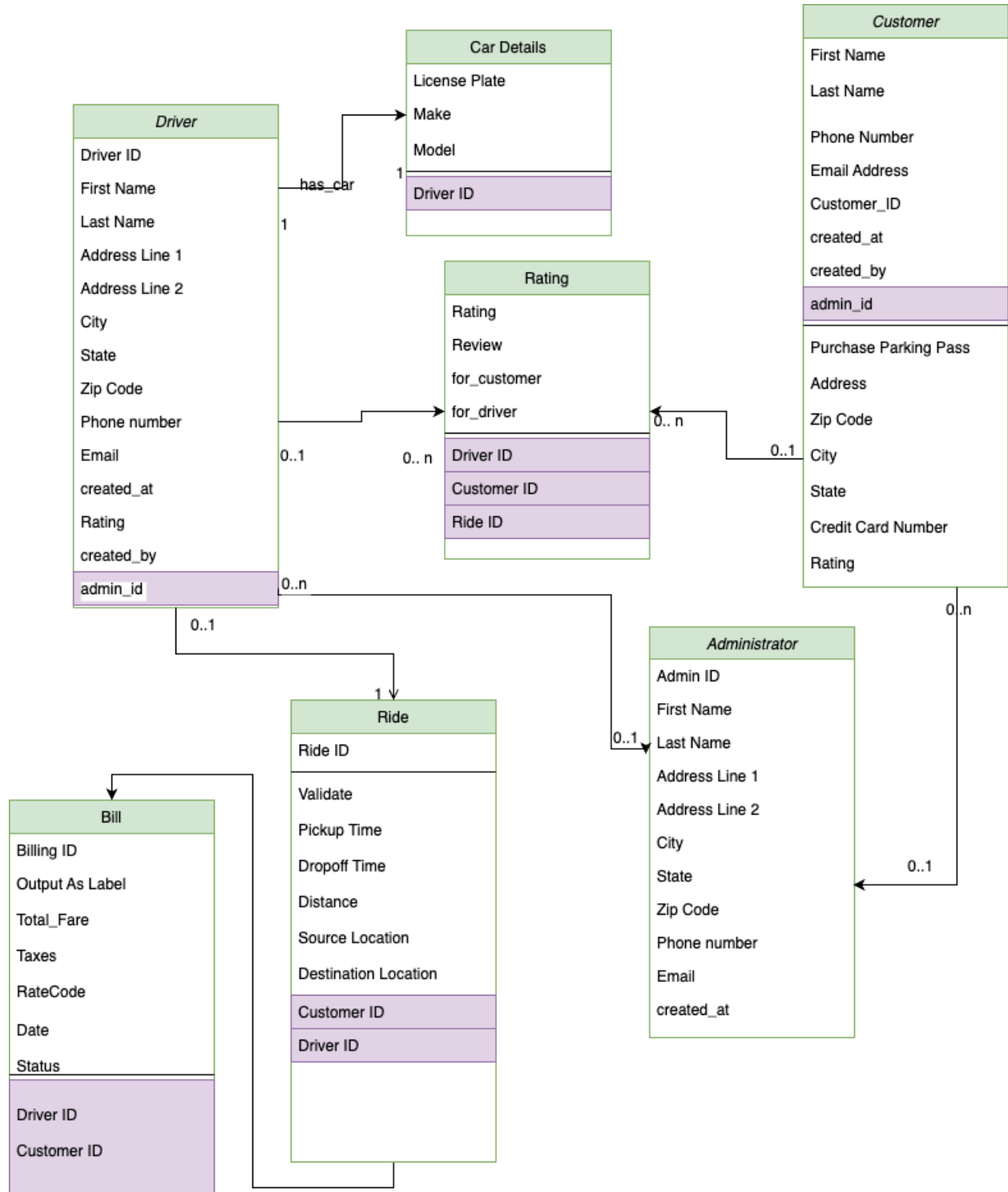
UBER Simulation

DATA 236 Group 3

Himani Shah, Yashasvi Kanchugantla , Kanchan Naik, Chelsea Jaculina

Data Model	2
Auth APIS	2
Driver Module/Service	5
Customer Module/Service	9
Billing Module/Service	17
Admin Module/Service	20
Rides Module/Service	25

Data Model



Auth APIS

Operation	URL	Request Body	Response
Customer signup	POST /api/customers/	Same as create customer	Same as create customer
Driver Sign up	POST /api/drivers	Same as create driver	Same as create driver
Login (single api for driver and customer login)	POST /api/auth/login/	{ "username": "john_doe", "password": "securepassword" }	401 - Bad Request 404 - Not found 200 OK { "message": "Login successful", "token": "JWT_TOKEN_HERE" } 200 OK Unsuccessful login { "error": "Invalid username or password" } 500 Internal Server Error { "error": "Invalid username or password" }
LogOut	/api/auth/logout/ CSRF and JWT		200 OK {

	token in header		<pre>"message": "Logged out successfully" } 200 Ok { "message": "Error in logging out" }</pre>
--	-----------------	--	---

Driver Module/Service

This module provides functionality related to driver management in the system.

- **Create a new Driver**
 - Adds a new driver to the system.
- **Delete an existing Driver**
 - Removes a driver record from the system.
- **List all Drivers known by the system**
 - Displays a list of all drivers.
- **Change Driver Information**
 - Allows updates to all driver attributes, such as name, address, etc.
- **Search for Drivers based on Attributes**
 - Enables searching drivers by specific attributes (e.g., name, location).
- **Display Information about a Driver**
 - Shows all available details about a selected driver.
- **Introduction Video of the Driver**
 - Allows drivers to upload an introduction video.

Operation	API Endpoint	Request Body	Response
<ul style="list-style-type: none">● Create a new Driver Adds a new driver to the system.	POST /api/drivers	{ "driver_id": "123-45-6789", "name": "John Doe", "phone": "123-456-7890", "email": "john.doe@exampl e.com", "address": "123 Main St, San Francisco, CA", "license_number" : "CA1234567", "vehicle_details" : {	201 Created: Driver successfully created. { "message": "Driver created successfully", "driver_id": "123-45-6789" } 400 Bad Request: Missing or invalid fields.

		<pre> "vehicle_id": "567-89-1234", "make": "Toyota", "model": "Camry", "year": "2020", "license_plate": "8XYZ123" } </pre>	
<ul style="list-style-type: none"> Delete an existing Driver <p>Removes a driver record from the system.</p>	DELETE /api/drivers/{driver_id}		<p>200 OK: Driver successfully deleted.</p> <pre> { "message": "Driver deleted successfully" } </pre> <p>404 Not Found: Driver not found.</p>
<ul style="list-style-type: none"> List all Drivers known by the system <p>Displays a list of all drivers.</p>	GET /api/drivers		<p>200 OK: List of all drivers</p> <pre> [{ "driver_id": "123-45-6789", "name": "John Doe", "phone": "123-456-7890" }, { </pre>

			<pre>"driver_id": "234-56-7891", "name": "Jane Smith", "phone": "987-654-3210" }]</pre>
<ul style="list-style-type: none"> • Change Driver Information <p>Allows updates to all driver attributes, such as name, address, etc.</p>	<p>PUT /api/drivers/{driver_id}</p>	<pre>{ "name": "John D. Doe", "phone": "321-654-0987", "email": "john.d.doe@example.com", "address": "456 Elm St, San Francisco, CA" }</pre>	<p>200 OK: Driver information updated successfully.</p> <pre>{ "message": "Driver information updated successfully" }</pre> <p>404 Not Found: Driver not found.</p>
<ul style="list-style-type: none"> • Search for Drivers based on Attributes <p>Enables searching drivers by specific attributes (e.g., name, location).</p>	<p>GET /api/drivers/search</p> <p>Query parameters: name=John & location=San Francisco</p>		<p>200 OK: List of drivers matching the search criteria.</p> <pre>[{ "driver_id": "123-45-6789", "name": "John Doe", "phone": "123-456-7890", "location": "San</pre>

			<pre> Francisco, CA" }] </pre>
<ul style="list-style-type: none"> • Display Information about a Driver <p>Shows all available details about a selected driver.</p>	<pre> GET /api/drivers/{driver_id} </pre>		<p>200 OK: Driver details.</p> <pre> { "driver_id": "123-45-6789", "name": "John Doe", "phone": "123-456-7890", "email": "john.doe@example.com", "address": "123 Main St, San Francisco, CA", "license_number" : "CA1234567", "vehicle_details ": { "vehicle_id": "567-89-1234", "make": "Toyota", "model": "Camry", "year": "2020", "license_plate": "8XYZ123" } } </pre>

			404 Not Found: Driver not found.
<ul style="list-style-type: none"> Introduction Video of the Driver <p>Allows drivers to upload an introduction video.</p>	POST /api/drivers/{driver_id}/upload-video	Video file	200 OK: Video uploaded successfully. <pre>{ "message": "Video uploaded successfully" }</pre> 400 Bad Request: Video format not supported.

Customer Module/Service

This module manages customer information and interactions with drivers.

- **Create a new Customer**
 - Adds a new customer to the system.

Operation	API End Point	Request Body	Response
<ul style="list-style-type: none"> Create a new Customer <p>Adds a new customer to the system.</p>	POST /api/customers/	<pre>{ "username": "john_doe1", "password": "securepassword", "profile_image": null, "address": "123"</pre>	201 Created 401 - Bad Request 404 - Not found <pre>{ "customer_id":</pre>

		<pre> Main St", "city": "New York", "state": "NY", "zip_code": "10001", "phone_number": "1234567890", "email": "john@example.com", "credit_card": "4111111111111111", "name": "Kanchan Naik" } </pre>	<pre> "bd3e7eda-9f", "Message": "Custom er Created Successfully" } 500 - Internal Server Error { "Message": "Error in creating customer" } </pre>
<ul style="list-style-type: none"> • List all Customers known by the system <p>Displays a list of all customers.</p>	GET /api/customers		<p>401 - Bad Request 403 - Forbidden 404 - Not found</p> <p>200 OK Success (With Data)</p> <pre> [{ "username": "john_doe1", "password": "securepassword", "profile_image": null, "address": "123 Main St", "city": "New York", </pre>

			<pre> "state": "NY", "zip_code": "10001", "phone_number": "1234567890", "email": "john@example.com", "credit_card": "4111111111111111", "name": "Kanchan Naik" }] ----- -- 200 OK (Without Data) [] ----- -- 500 - Internal Server Error { "Message": "Error in fetching customers" } </pre>
GET customer by id	GET /api/customers/ {customer_id}/		401 - Bad Request 403 - Forbidden 404 - Not found 200 OK Success

			<pre>{ "customer_id": "8c4d3410-d7", "name": "Kanchan Naik", "address": "2 Pierce", "city": "San Jose", "state": "CA", "zip_code": "95123", "phone_number": "565-1734", "email": "kanchan.naik@exa mple.com", "credit_card": "4131-1131-1111-1 151", "rating": null }</pre> <p>----- --</p>
--	--	--	--

			500 - Internal Server Error <pre>{ "Message": "Error in fetching customer" }</pre>
<ul style="list-style-type: none"> Generate Bill for a Customer <p>Generates a bill after each ride for the customer.</p>	POST /api/customers/{customer_id}/generate-bill/	<pre>{ "rideID": "7890", "distance": 15.0, "rate_code": 2 }</pre>	401 - Bad Request 403 - Forbidden 404 - Not found 201 Created <pre>{ "message": "Bill generated successfully.", "billID": "4567", "customer_id": "8c4d3410-d7", "distance": 15.0, "fare_amount": 20.50, "tax_amount": 2.05, "total_amount": 22.55, "pickup_location": "123 Main St, San Francisco, CA", "dropoff_location": "456 Elm St, San Francisco, CA", }</pre>

			<pre> "ride_date": "2024-11-07T14:30:00Z" } ----- 500 - Internal Server Error { "Message": "Error in creating bill" } </pre>
<ul style="list-style-type: none"> • Select Driver within 10 Miles from Customer Location <p>Searches for drivers located within a 10-mile radius of the customer's location.</p>	<pre> GET /api/customers/ {customerID}/ne arby-drivers </pre>		<pre> 401 - Bad Request 403 - Forbidden 404 - Not found 200 OK Success [{ "driverID": "567-89-1234", "name": "Jane Smith" "location": { "latitude": 37.7749, "longitude": -122.4194 }, "rating": 4.8 }] ----- 500 - Internal Server Error { "Message": "Error in fetching </pre>

			nearby drivers” }
<ul style="list-style-type: none"> • Delete an existing Customer 	DELETE /api/customers/ {customerID}		401 - Bad Request 403 - Forbidden 404 - Not found 204 - OK <pre>{ "message": "Customer deleted successfully.", "customerID": "123-45-6789" }</pre> ----- 500 - Internal Server Error <pre>{ "Message": "Error in deleting customer" }</pre>
Update Customer Details Partial Update supported	PUT /api/customers/ {customerID}	<pre>{ "address": "5678 Oak Street", "city": "Santa Clara", "zipCode": "95050", "phoneNumber": "555-5678" }</pre>	401 - Bad Request 403 - Forbidden 404 - Not found 200 OK <pre>{ "message": "Customer information updated successfully.", "customerID": "123-45-6789" }</pre> ----- 500 - Internal Server Error

			<pre>{ "Message": "Error in updating customer" }</pre>
<ul style="list-style-type: none"> • Upload Ride Images <p>Allows customers to upload images related to their ride experience.</p>	<pre>POST /api/customers/ {customer_id}/{ ride_id}/upload -ride-images/</pre>	<pre>{ "images": ["fileurl1", "fileurl2",] }</pre>	<p>401 - Bad Request 403 - Forbidden 404 - Not found</p> <p>201 Created</p> <pre>{ "message": "Images uploaded successfully for the ride." }</pre> <p>500 - Internal Server Error</p> <pre>{ "Message": "Error in uploading ride images" }</pre>

- **Delete an existing Customer**
 - Removes a customer record from the system.
- **List all Customers known by the system**
 - Displays a list of all customers.
- **Generate Bill for a Customer**
 - Generates a bill after each ride for the customer.
- **Select Driver within 10 Miles from Customer Location**
 - Searches for drivers located within a 10-mile radius of the customer's location.
- **Upload Ride Images**
 - Allows customers to upload images related to their ride experience.

Billing Module/Service

This module handles billing-related operations for rides.

- **Create a New Bill for Each Ride**
 - Generates a bill with details for each ride.
- **Delete an Existing Bill**
 - Allows deletion of a specific bill record.
- **Search for a Bill**
 - Provides search functionality to find bills by various criteria.

Operation	API Endpoint	Request Body	Response
<ul style="list-style-type: none">● Create a New Bill for Each Ride<ul style="list-style-type: none">○ Generates a bill with details for each ride.	<p>POST /api/billing</p> <p>Creates a new billing record associated with a specific ride.</p>	<pre>{ "billing_id": "100-20-3000", "date": "2024-11-09", "pickup_time": "14:30", "dropoff_time": "15:00", "distance_covered": "12.5", "total_amount": "25.00", "source_location": "37.7749, -122.4194", "destination_location":</pre>	<ul style="list-style-type: none">● 201 Created● 400 Bad Request (if required fields are missing or invalid) <pre>{ "message": "Billing record created successfully", "billing_id": "100-20-3000" }</pre>

		<pre>"37.8044,-122.2711", "driver_id": "123-45-6789", "customer_id": "987-65-4321" }</pre>	
<ul style="list-style-type: none"> • Delete an Existing Bill <ul style="list-style-type: none"> ○ Allows deletion of a specific bill record. 	<p>DELETE /api/billing/{billing_id}</p> <p>Deletes an existing billing record by its unique billing ID.</p>		<ul style="list-style-type: none"> • 204 No Content (if successfully deleted) • 404 Not Found (if the billing record does not exist)
<ul style="list-style-type: none"> • Search for a Bill <ul style="list-style-type: none"> ○ Provides search functionality to find bills by various criteria. 	<p>GET /api/billing/search</p> <p>Searches for billing records based on query parameters</p> <p>Query Parameters (Optional):</p> <ul style="list-style-type: none"> • billing_id: unique identifier for the billing record • date: date of the billing record • driver_id: ID of the driver associated 		<ul style="list-style-type: none"> • 200 OK • 404 Not Found (if no matching billing record is found) <pre>[{ "billing_id": "100-20-3000", "ride_id": "555-11-4444", "date": "2024-11-09", "pickup_time": "14:30",</pre>

	with the ride <ul style="list-style-type: none"> • customer_id: ID of the customer associated with the ride • ride_id: ID of the associated ride • 		<pre> "dropoff_time": "15:00", "distance_covered": "12.5", "total_amount": 25.00, "source_location": {"latitude": "37.7749", "longitude": "-122.4194"}, "destination_location": {"latitude": "37.8044", "longitude": "-122.2711"}, "driver_id": "123-45-6789", "customer_id": "987-65-4321" }] </pre>
--	--	--	--

Admin Module/Service

The Admin module provides administrative functionalities.

- **Add Drivers to the System**
 - Enables adding drivers into the system.
- **Add Customers to the System**
 - Allows administrators to add customers into the system.
- **Review Driver/Customer Account**

- Facilitates reviewing driver and customer accounts for verification.
- **Show Statistics**
 - Displays revenue statistics per day and the total rides by area.
- **Show Graphs/Charts**
 - Visualizes data for rides per area, per driver, and per customer.
- **Search for a Bill Based on Attributes**
 - Allows administrators to locate bills by specific attributes.
- **Display Information about a Bill**
 - Shows details for a specific bill.

Operation	API Endpoint	Request Body	Response
<ul style="list-style-type: none"> ● Enable adding a Driver to the system <ul style="list-style-type: none"> ○ Enables Adding a new driver to the system. 	POST /api/drivers	<pre>{ "driverId": "123-45-6789", "firstName": "John", "lastName": "Doe", "address": "123 Main St", "city": "San Francisco", "state": "CA", "zipCode": "94105", "phoneNumber": "1234567890", "email": "john.doe@example.com", "carDetails": { "make": "Toyota", "model": "Camry", "year": 2020, "licensePlate": "ABC123" } }</pre>	<p>201 Created: Driver successfully created.</p> <pre>{ "status": "success", "message": "Driver added successfully", "driverId": "123-45-6789" }</pre> <ul style="list-style-type: none"> ● 400 Bad Request: Invalid input data ● 409 Conflict: Driver already exists ● 500 Internal Server Error: Server error

		}	
POST /api/admin/customers		<pre>{ "customerId": "987-65-4321", "firstName": "Jane", "lastName": "Smith", "address": "456 Oak St", "city": "San Jose", "state": "CA", "zipCode": "95113", "phoneNumber": "9876543210", "email": "jane.smith@example.com", "creditCardDetails": { "cardNumber": "*****1234", "expirationDate" : "12/25", "cvv": "****" } }</pre>	<pre>{ "status": "success", "message": "Customer added successfully", "customerId": "987-65-4321" }</pre> <ul style="list-style-type: none"> • 400 Bad Request: Invalid input data • 409 Conflict: Customer already exists • 500 Internal Server Error: Server error
Review Driver/Customer Account	GET /api/admin/accounts/{accountType}/{id}		<pre>{ "id": "123-45-6789", "accountType": "driver", "firstName": "John", "lastName": "Doe",</pre>

			<pre> "email": "john.doe@example.com", "phoneNumber": "1234567890", "rating": 4.8, "totalRides": 500, "accountStatus": "active" } </pre> <ul style="list-style-type: none"> • 400 Bad Request: Invalid account type or ID format • 404 Not Found: Account not found • 500 Internal Server Error: Server error
Show Statistics	GET /api/admin/statistics	Query Parameters <ul style="list-style-type: none"> • <code>startDate</code>: Start date for the statistics (YYYY-MM-DD) • <code>endDate</code>: End date for the statistics (YYYY-MM-DD) 	<pre> { "revenuePerDay": [{ "date": "2024-11-08", "revenue": 5000.50 }, { "date": "2024-11-09", "revenue": 5500.75 }], </pre>

			<pre>"totalRidesByArea": [{ "area": "Downtown", "totalRides": 1000 }, { "area": "Suburb", "totalRides": 750 }]</pre> <ul style="list-style-type: none"> • 400 Bad Request: Invalid date format • 404 Not Found: No data available for the specified date range • 500 Internal Server Error: Server error
Show Graphs/Charts	GET /api/admin/charts	Query Parameters <ul style="list-style-type: none"> • <code>chartType</code>: Type of chart ("area", "driver", "customer") • <code>startDate</code>: Start date for 	<pre>{ "chartType": "area", "data": [{ "label": "Downtown", "value": 1000 }] }</pre>

		<p>the data (YYYY-MM-DD)</p> <ul style="list-style-type: none"> • <code>endDate</code>: End date for the data (YYYY-MM-DD) 	<pre> }, { "label": "Suburb", "value": 750 }] } </pre> <ul style="list-style-type: none"> • 400 Bad Request: Invalid chart type or date format • 404 Not Found: No data available for the specified parameters • 500 Internal Server Error: Server error
Search for a Bill Based on Attributes	GET /api/admin/bills/search	<p>Query Parameters</p> <ul style="list-style-type: none"> • <code>billingId</code>: Billing ID • <code>driverId</code>: Driver ID • <code>customerId</code>: Customer ID • <code>startDate</code>: Start date for the search (YYYY-MM-DD) • <code>endDate</code>: End date for the search (YYYY-MM-DD) 	<pre> { "bills": [{ "billingId": "BILL123456", "driverId": "123-45-6789", "customerId": "987-65-4321", "date": "2024-11-09", "amount": 25.50, "status": </pre>

		D) <ul style="list-style-type: none"> • <code>minAmount</code>: Minimum bill amount • <code>maxAmount</code>: Maximum bill amount •) 	<pre>"paid" }], "totalResults": 1 }</pre> <ul style="list-style-type: none"> • 400 Bad Request: Invalid search parameters • 404 Not Found: No bills found matching the criteria • 500 Internal Server Error: Server error
Display Information about a Bill	GET /api/admin/bills/{billingId}		<pre>{ "billingId": "BILL123456", "driverId": "123-45-6789", "customerId": "987-65-4321", "date": "2024-11-09", "pickupTime": "14:30:00", "dropoffTime": "15:00:00", "distance": 10.5, "amount": 25.50, "status": "paid", "sourceLocation"</pre>

			<pre> : { "latitude": 37.7749, "longitude": -122.4194 }, "destinationLoca tion": { "latitude": 37.3382, "longitude": -121.8863 } } </pre> <ul style="list-style-type: none"> • 400 Bad Request: Invalid billing ID format • 404 Not Found: Bill not found • 500 Internal Server Error: Server error
--	--	--	---

Rides Module/Service

This module manages ride-related actions and records.

- **Create a New Ride**
 - Allows the creation of a new ride entry.
- **Edit an Existing Ride**
 - Facilitates modification of ride details.
- **Delete an Existing Ride**
 - Removes a specific ride record.
- **List all Rides of a Customer**
 - Displays the ride history of a specific customer.

- **List all Rides of a Driver**
 - Shows all rides associated with a particular driver.
- **Show Ride Statistics by Location**
 - Provides statistical data on rides based on location.

Operation	API Endpoint	Request Body	Response
<ul style="list-style-type: none"> ● Create a New Ride <ul style="list-style-type: none"> ○ Allows the creation of a new ride entry. 	<p>POST /api/rides</p> <p>Creates a new ride with the specified details</p>	<pre>{ "ride_id": "555-11-44 44", "pickup_lo cation": {"latitude ": "37.7749", "longitude ": "-122.4194 "}, "dropoff_l ocation": {"latitude ": "37.8044", "longitude ": "-122.2711 "}, "date_time ": "2024-11-0 9T14:30:00 "</pre>	<ul style="list-style-type: none"> ● 201 Created ● 400 Bad Request

		<pre> Z", "customer_ id": "987-65-43 21", "driver_id ": "123-45-67 89", "rate_code ":2 } </pre>	
<ul style="list-style-type: none"> • Edit an Existing Ride <ul style="list-style-type: none"> ○ Facilitate s modification of ride details. 	<p>PUT api/rides/{ride_id}</p> <p>Updates details of an existing ride</p>	<pre> { "pickup_lo cation": {"latitude ": "37.7749", "longitude ": "-122.4194 "}, "dropoff_l ocation": {"latitude ": "37.8044", "longitude ": "-122.2711 "}, "date_time </pre>	<ul style="list-style-type: none"> • 200 OK • 400 Bad Request (if invalid data) • 404 Not Found (if ride does not exist) <pre> { "message": "Ride created successfully", "ride_id": "555-11-4444" } </pre>

		<pre> ": "2024-11-0 9T14:30:00 Z", "customer_ id": "987-65-43 21", "driver_id ": "123-45-67 89" } </pre>	
<ul style="list-style-type: none"> • Delete an Existing Ride <ul style="list-style-type: none"> ○ Removes a specific ride record. 	<p>DELETE api/rides/{ride_id}</p> <p>Deletes an existing ride by ride ID</p>		<ul style="list-style-type: none"> • 204 No Content (if successfully deleted) • 404 Not Found (if ride does not exist)
<ul style="list-style-type: none"> • List all Rides of a Customer <ul style="list-style-type: none"> ○ Displays the ride history of a specific customer. 	<p>GET api/rides/customer/{customer_id}</p>		<ul style="list-style-type: none"> • 200 OK • 404 Not Found (if customer does not exist or has no rides) <pre> [{ "ride_id": "555-11-4444", "pickup_location": {"latitude": "37.7749", </pre>

			<pre> "longitude": "-122.4194"}, "dropoff_location": {"latitude": "37.8044", "longitude": "-122.2711"}, ... }] </pre>
<ul style="list-style-type: none"> • List all Rides of a Driver <ul style="list-style-type: none"> ○ Shows all rides associated with a particular driver. 	<p>GET api/rides/driver/{driver_id}</p> <p>Retrieves a list of all riders for a specified driver</p>		<ul style="list-style-type: none"> • 200 OK • 404 Not Found (if driver does not exist or has no rides) <pre> [{ "ride_id": "555-11-4444", "pickup_location": {"latitude": "37.7749", "longitude": "-122.4194"}, "dropoff_location": {"latitude": "37.8044", "longitude": "-122.2711"}, ... }] </pre>

<ul style="list-style-type: none"> • Show Ride Statistics by Location <ul style="list-style-type: none"> ○ Provides statistical data on rides based on location. 	<p>GET /api/rides/statistics/location</p> <p>Provides ride statistics based on pickup or drop-off locations</p> <p>Query Parameters (Optional):</p> <ul style="list-style-type: none"> • location_type: "pickup" or "dropoff" (default: "pickup") • latitude: latitude of the location to filter by • longitude: longitude of the location to filter by • radius: radius in miles to include in the statistics (default: 10) 		<pre>{ "location_type": "pickup", "location": {"latitude": "37.7749", "longitude": "-122.4194"}, "total_rides": 250, "average_fare": 23.5, "most_frequent_destination": {"latitude": "37.8044", "longitude": "-122.2711"}, "average_distance": 12.3 }</pre>
--	---	--	---