Car rental system

Requirements Specification

Version 1.0

May 24, 2024

Worked by: Alesio Rajta
Arlind Alliu
Denaldo Selala
Hedije Jazaj
Inva Lybeshari
Joel Bitri

Table of Contents

Executive Summary

1.1 Project Overview

The goal of this project is to develop a web-based software application designed to manage a car rental franchise efficiently. The software will enable the franchise to track every vehicle in their fleet, capturing details such as make, model, year of manufacture, color, transmission type, cost per day, fuel type, number of doors and seats, and more. Additionally, it will facilitate the management of reservations, including dates, times, and locations, and support the creation of client invoices and contracts.

This application will streamline operations for franchise employees, reducing reliance on paper-based documentation. Agents will be able to store and share transaction data within the application, both with other agents and administrators. The software's direct integration with the invoicing system will ensure all transactions are recorded and allow bills to be issued according to customer preferences. A key feature will be the ability for agents to update rental statuses in real-time, preventing any discrepancies between customers.

The software will also enhance the customer experience with features such as a secure and user-friendly payment system, and a catalog providing detailed car information, including prices and specifications. The application will strike a balance between user-friendliness for customers and professionalism for agents and administrators, ensuring menus are simple and accessible for customers while providing detailed, professional views for internal use, supporting the compilation of financial and legal documents.

Data security will be a top priority in the development of this application. Measures will include the creation of a secure database and encryption of sensitive information, such as employees' ID or driver's license. Ensuring users feel confident in the integrity and security of their data is essential before they can fully enjoy the software's features.

Development Team

- Project Manager
- Software Architects
- UI/UX Designers
- Frontend Developers
- Backend Developers
- Database Developers
- Quality Assurance Engineers
- Technical Writer

Collaboration with external providers may be necessary for GPS tracking, marketing/advertising integration, and payment processing.

2. Product/Service Description

2.1 Product Context

This software application is designed to enhance and streamline the car rental process and improve the interaction between the car rental franchise and its customers. While there are many similar apps on the market, our team has developed this software to incorporate all the valuable and desirable features from existing solutions while introducing new functionalities that set our app apart and revolutionize the use of technology in the car rental industry.

The software boasts a user-friendly GUI that is easy to use and compatible with all devices. Its straightforward design ensures that workers can quickly adapt to it, and it attracts more clients by being intuitive and easy to navigate.

For customers, the application provides interactive menus that allow them to make reservations quickly and securely. For administrators and agents, the emphasis is on facilitating seamless communication within the organization. The system includes a grid of messages and notifications to keep everyone updated and offers real-time commands for effective internal communication.

2.2 User Characteristics

Administrator:

- Manages all actions performed by agents and drivers.
- Oversees employee information management.
- Accesses data on total incomes and expenses.
- Requires access to information on all cars, reservations, and customers.
- Adjusts car prices and specifications.
- Adds and removes cars from the catalog.

Car Rental Agent:

- Accesses data on available cars and reservations.
- Issues invoices and rental contracts.
- Requires access to GPS data for tracking cars.
- Receives notifications about daily reservations and rented cars.

- Checks for fines and ensures customers pay them.
- Notifies other agents when a car becomes unavailable.

Customer:

- Logs in or signs up to make a reservation.
- Views available cars and makes reservations through the website.
- Provides personal information, including name, surname, birthday, ID, email, and phone number.
- Pays via credit card.
- Returns the car in good condition and pays any fines incurred.
- Notifies the agency of any car problems.
- Inquires about prices and car specifications.
- May have different subscription levels (regular, premium, or platinum) with varying rental options and extra features.

Car maintenance operator:

- Accesses reservation information.
- Checks off whether a car has been delivered.
- Inspects the car before and after delivery, ensuring it is in good condition.
- Reports any damage or issues to the administrator.
- Reports any necessary investments for the cars.

2.3 Assumptions

- Staff and customers are assumed to be familiar with English or Albanian.
- A stable and reliable internet connection is assumed to be available for both staff and customers.
- The application will be available on popular mobile devices running the latest iOS and Android operating systems.
- Staff and customers are assumed to have basic knowledge of operating the application and troubleshooting minor issues.
- The car rental company is assumed to have a valid and up-to-date insurance policy for all vehicles and customers, with customers being informed about the coverage and limitations.
- The car rental company is assumed to have enough vehicles to meet customer demand during peak times.
- A customer support system will be in place to address any concerns or issues customers may have while using the application or renting a vehicle.

2.4 Constraints

- Every manager, agent, driver, and customer must log in or register with a username and password.
- The software must integrate with other systems, such as financial platforms, to enable online payments for reservations.
- Customers must create an account or log into an existing one to make a reservation.
- The software must efficiently handle a large volume of customers, reservations, and other operations, particularly during peak seasons.
- The software must ensure the security and protection of customer data, reservation details, and payment history.
- It must provide a backup solution to recover data in case of crashes or errors.

2.5 Dependencies

1. To print the bill contract, the car rental agent must enter the customer's information and details of the available car.
2. If a car is not available, the car rental agent cannot make a reservation for it. Customers must log in, select an available car, and pay for the service to make a rental reservation.
3. If the chosen car is unavailable, the customer may select another car for reservation.
4. To finalize the return of the vehicle, the customer must pay any fines associated with the vehicle.
5. Customers can only select certain car models based on their subscription level.
6. Customers without a subscription cannot select any car they want.
7. To receive a discount on their next rental, customers must have made at least five reservations.
8. A car cannot be available for reservations if the driver has confirmed it is being used by another customer.
9. The availability of a car depends on its condition after rental, as checked by the driver, to ensure it can be rented again.

3. Requirements

3.1 Functional Requirements

Req#	Requirement	Comments	Priority	Date Reviwes	SME Reviewed / Approved
R_01	The system should support a multi-user environment.	This is done to facilitate the permission based features. It is much easier to have users rather than check permissions on action each time the server receives a request.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_02	The system should be supported by all browsers.	The system should not have limitations about the browser since clients are free to use any browser they want.	1	22/5/2024	Joel Bitri & Denaldo Selala

R_03	The system should be usable in tablets and mobile phones.	Obviously the goal of the project is facilization of the car rental business and the opportunity to use such an application from the comfort of your phone is a must	1	22/5/2024	Joel Bitri & Denaldo Selala
R_04	The system should differentiate users' permission and features based on their level	For the application to function logically as well as technically, we need to implement a system that is able to tell if the person who is using the app is allowed to do certain things. For example an agent is not allowed to fire employees so the system will not give this feature to an agent user	1	22/5/2024	Joel Bitri & Denaldo Selala
R_05	The system should provide quick log-in information validation based on the most recent security regulations.	This is an obligation for any app that is trying to enter the market in 2024	2	22/5/2024	Joel Bitri & Denaldo Selala

R_06	The system should be able to carry out encrypted transactions.	This is mostly related to safety and to the banking process within the boundaries of legislation.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_07	The system should keep track of financial statistics.	Since the system will be used to help the manager file for taxes but also run the business it is obligatory to gather some financial statistics so as to enable other functionalities required.	3	22/5/2024	Joel Bitri & Denaldo Selala

R_08	The system should be able to transfer data in almost real time.	This is a must because the application only achieves its goal if all the employees are connected properly.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_09	The system should keep track of a user's job status.	This is so that accounts of employees that are fired or suspended will not have the right to modify or enter new information.	3	22/5/2024	Joel Bitri & Denaldo Selala
R_10	The system should assign unique user id-s based on UUID technology.	This is to differentiate users in the database.	3	22/5/2024	Joel Bitri & Denaldo Selala
R_11	The system should calculate employees' progress and estimate their productivity during a period.	This is related to the business end of the application.	3	22/5/2024	Joel Bitri & Denaldo Selala

R_12	The system should provide data about the cars that are being rented the most or the least and should update such data dynamically.	This is related to the business end of the application.	2	22/5/2024	Joel Bitri & Denaldo Selala
R_13	The system should handle data incoming from GPS tracking devices.	It is important to allow the agents to track the vehicles	2	22/5/2024	Joel Bitri & Denaldo Selala
R_14	The system should maintain only sufficient active information.	This is related to the data structures that the system should employ to guarantee efficient time of execution and sufficient information to the users.	1	22/5/2024	Joel Bitri & Denaldo Selala

R_15	The system should allow users to download and print information such as bills or contracts.	One of the functionalities of the application is to manage hard copies of important information such as invoices or transaction documents.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_16	The system should be able to generate and save invoices.	This is important to keep track of the financial activity of the business.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_17	The system should facilitate communication between coworkers.	Crucial because the system is supposed to save time.	3	22/5/2024	Joel Bitri & Denaldo Selala
R_18	The system should calculate incomes and return taxes automatically.	Related to the business end of the application.	1	22/5/2024	Joel Bitri & Denaldo Selala

R_19	The system should allow the registration of new cars or car models.	The list must be updatable.	3	22/5/2024	Joel Bitri & Denaldo Selala
R_20	The system should allow users to modify cars' information.	It is important that objects within the system are dynamic rather than static.	3	22/5/2024	Joel Bitri & Denaldo Selala
R_21	The system should be able to differentiate between users and display the same view as read-only or as editable based on the user level.	This, for example, allows the admin to view and edit the car's list but the agent can only pick one of the cars.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_22	The system should be able to notify for abnormal situations.	Abnormal situations such as invalid inputs, failure, problems with the servers, etc.	3	22/5/2024	Joel Bitri & Denaldo Selala

R_23	The system should allow users to check in and out of their shifts.	This is so that the hours of work are kept track of automatically.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_24	The system should notify users who are concerned about any important update, such as the change of the rental time for example.	Notifications should be enabled.	2	22/5/2024	Joel Bitri & Denaldo Selala
R_25	The system should provide safe transactions for online payments.	Since customers can use the website to make reservations online, they should also be able to pay online for their reservations. This is why the system should make sure that the payment process is safe.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_26	The system should support embedded systems.	The website will have a GPS system and a Fine system to support the needs of the agency. This is why the system should be able to handle the embedded system.	1	22/5/2024	Joel Bitri & Denaldo Selala

R_27	The system should generate reports for incomes, number of reservations, and cars.	The admin may want to check how the agency is doing and the incomes to make a better business plan or get general information.	2	22/5/2024	Joel Bitri & Denaldo Selala
R_28	The admin can create, update or delete services or cars from the system.	This is done so the clients and agents are up to date with what the agency is offering.	2	22/5/2024	Joel Bitri & Denaldo Selala
R_29	The user has to log in to make a reservation.	If they do not log in, they can only view the services that are provided by the agency.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_30	The admin can create, update or delete an employee's profile.	The admin can add agents or drivers in the system or modify existing ones.	2	22/5/2024	Joel Bitri & Denaldo Selala

R_31	The system should support online payment.	The system should be able to offer the customer the opportunity to pay via credit card through different banks.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_32	The agent can add reservations or modify them.	They can change the reservation dates or other information or add new ones when getting the requirements from the clients.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_33	The agency should provide a catalog with all their cars and other services.	This way, the people visiting the website can get a better understanding of the agency and know what services they are providing.	2	22/5/2024	Joel Bitri & Denaldo Selala
R_34	The customers can leave comments and reviews about the services offered by the agency.	In order to get feedback from the customers, the website provides a section for them to leave comments and ratings.	4	22/5/2024	Joel Bitri & Denaldo Selala

R_35	The website should provide the customers the price of every service.	When looking through the catalog of services, the customer should also have information about the price.	3	22/5/2024	Joel Bitri & Denaldo Selala

R_36	The driver should be able to change the delivery status and returning status of the cars.	The driver will have a section to update if the car has been delivered to the customer or taken from them through a check box.	1	22/5/2024	Joel Bitri & Denaldo Selala
R_37	The system should provide customer support through phone, email, or chat.	In case the customer has problems during the rental period and requires assistance, the website should provide them with ways to communicate with the agents.	2	22/5/2024	Joel Bitri & Denaldo Selala

3.2 Non-Functional Requirements

3.2.1 Product Requirements

3.2.1.1 User Interface Requirements

The car rental franchise application UI should provide a seamless and user-friendly experience for all user roles: managers, car rental agents, customers, and drivers.

Login Screen

- A login screen with clearly marked username and password fields should be included to allow users to access their accounts securely.

Dashboards

- Upon login, each user role should be greeted with a visually appealing and intuitive dashboard tailored to their specific needs.
    - **Administrator's Dashboard**: Displays revenue and expense totals.
    - **Car Rental Agent's Dashboard**: Shows notifications about bookings and available cars for the day.

Fleet Management Interface

- This interface should allow easy addition, editing, and deletion of vehicle details such as model, year, color, transmission type, fuel type, doors, seats, and price per day.
- Filters and search options should enable users to find specific cars based on their criteria.
- A visual indicator should clearly show vehicle availability, and a notification should alert the company if a vehicle is already rented.

Booking Management Interface

- A clear calendar view or booking system should be provided for users to create and manage bookings.
- The interface should include notifications and reminders for upcoming reservations and overdue returns.

Customer Interface

- Customers should be able to enter their first name, last name, date of birth, card ID, email address, and phone number to log in or register.
- Customers should be able to view available cars, make bookings, and select the time, date, and location for vehicle pickup.
- An online credit card payment area should be provided.
- Customers should be informed of any fines incurred during the rental period and be able to pay them upon returning the vehicle.

Driver Interface

- Drivers should be able to view reservation details without modifying them.
- The interface should include a checkbox to indicate whether the car has been delivered and a checklist to

ensure the car is in good condition before and after delivery.
- The checklist should cover advanced functions and the overall condition of the vehicle.

Overall, the user interface should prioritize usability, clear navigation, and intuitive interactions to enhance the user experience for each role within the car rental franchise application.

3.2.1.2 Usability

- The software should enable real-time notifications for drivers to communicate with other employees while in the field.
- The software should be easy to use, even for individuals with minimal experience using smartphones or computers.
- The design should be compatible with and adaptable to all smart devices.
- The software should display clear messages when user action is required.
- The software should maintain the same functionalities regardless of user preferences, such as language, screen mode, and font size.
- The software must support both keyboard shortcuts and mouse/touchscreen interactions.
- The software should support common file types, allowing users to upload files freely, regardless of their device (e.g., .zip files on Linux and .rar files on Windows).
- Navigation should be clear and consistent, ensuring users are aware when switching menus and maintaining the same screen size during transitions.

3.2.1.3 Efficiency

3.2.1.3.1 Performance Requirements

Response Time:

- The software must ensure immediate execution of user interactions, with target response times under a few seconds to provide users with a seamless and responsive experience.

Processing Speed:

- The program should swiftly analyze input and perform calculations, completing complex operations within a few milliseconds to prevent delays or slowdowns.

Scalability:

- The software must be scalable to accommodate growing data and user volumes, supporting numerous concurrent transactions, including 1000+ concurrent users, while maintaining high performance levels.

Dependability:

- System faults or breakdowns should be minimized, ensuring high dependability. The software should aim for high availability, with an uptime goal of at least 99% or greater.

Network Performance:

- The software should exhibit efficient network performance for data transfer and remote communication, aiming for low latency with network response times around 100 milliseconds.

3.2.1.3.2 Space Requirements

System Resource Usage:

- The software must utilize system resources wisely to prevent excessive consumption, maintaining low resource usage levels. For example, CPU utilization should be kept below 70%, and RAM usage below 80%, to avoid slowdowns or performance issues.

Load Handling:

- The program should effectively handle anticipated loads and peak periods without significantly degrading performance. It should enable concurrent user requests to achieve a maximum response time of 2 seconds even under heavy loads.

Data Storage and Retrieval:

- The software must efficiently retrieve and store data, particularly for large datasets or databases. Database queries should be optimized for quick retrieval, aiming for a target response time of 500 milliseconds or less.

Database Capacity:

- The software's database should have substantial capacity, supporting a minimum of 500 GB to accommodate millions of records and transactions.

File Attachment Limit:

- Users should be able to attach files and documents to their records, with the software allowing uploads of files up to 20 megabytes (MB) in size to ensure users can upload and store larger files if needed.

Memory Utilization:

- The software should optimize memory utilization for effective performance. It should aim to keep memory consumption below 500 megabytes (MB) or 10% of the available system memory to avoid excessive utilization and associated system slowdowns.

3.2.1.4 Dependability

Availability:

- The software should be reliable and available 24/7 with minimal downtime, accessible from any internet-connected device.
- It must have sufficient server capacity to support numerous concurrent users.
- Compatibility with widely used operating systems and browsers should be ensured.
- Security measures must be in place to prevent unauthorized access and safeguard data privacy.

Reliability:

- Data integrity must be maintained to accurately store and retrieve information about cars, reservations, users, and financial transactions.
- The system should be fault-tolerant, resilient to hardware or software failures, ensuring critical functionality remains available.
- Automated backup and recovery mechanisms should be implemented to ensure data integrity and continuity.
- Error handling should be graceful, with clear and user-friendly messages to guide users in troubleshooting or contacting support if needed.
- Strong security measures, including encryption of sensitive data and secure authentication, should be implemented to protect user data and financial transactions.

Extension Capability:

- The software should be designed to handle increasing user loads and data volumes without significant performance degradation, adjusting resources as needed to meet demand.

Performance Monitoring:

- Monitoring mechanisms should be in place to track system performance, identify bottlenecks, and ensure optimal response times. Proactive monitoring enables administrators to maintain high performance levels.

User Feedback and Support:

- Users should be able to provide feedback or report problems, with effective support channels available to address concerns and resolve issues promptly.

Compliance:

- The software must comply with industry standards, regulations, and best practices to ensure confidentiality and security of user data, as well as compliance with legal requirements for financial transactions and security.

Monitoring:

- Regular monitoring of performance, security, and backups should be conducted to maintain system integrity and prevent data loss in case of system failure.

Maintenance:

- Regular updates with the latest features and security patches should be applied.
- 24/7 user support should be provided via a help desk for issue reporting.
- Regular testing, including unit testing, integration testing, and system testing, should be performed to ensure software reliability.
- Up-to-date documentation should be maintained to facilitate ease of use.

3.2.1.5 Security

- Encryption and Authentication:

  - The software must employ robust encryption and authentication mechanisms to safeguard sensitive data, including credit card information and employee records.

- Role-Based Access Control:

  - Role-based access control should be implemented to limit user privileges and prevent unauthorized actions.

- Firewall and Intrusion Detection:

  - A firewall and intrusion detection system should be integrated into the software to thwart cyber-attacks and unauthorized access attempts.

- Compliance with Data Protection Regulations:

  - The software must comply with pertinent data protection regulations, such as GDPR or CCPA, to ensure the privacy and security of user data.

3.2.2 Organizational Requirements

3.2.2.1 Environmental Requirements

- All critical documents must be stored within the system's database due to its involvement in essential activities such as payments, tax filing, and financial management. This includes information related to bills, contracts, employee deals, and bonuses. All data will be retained with the consent of the company’s directors.
- Additionally, the system may require information about office locations and the whereabouts of cars to ensure the accuracy of data presented in the web application and to enhance public awareness.
- As the system does not involve hardware components, there are no specific environmental requirements for its usage. However, it is assumed that the computers and smart devices running the software will be equipped with the latest technologies.

3.2.2.2 Operational Requirements

• Users must log in before accessing the system.

• Users can create their own accounts or have them created by the administrator.

• The system must avoid overloading, ensuring that users do not perform multiple transactions from the same account simultaneously.

• The system must be available 24/7.

• Essential information changes require approval from the administrator.

• Users must ensure stable connections when conducting transactions of high importance.

• Printing invoices and contracts requires users to have access to resources such as a printer and paper.

• To utilize the tracking system, users must provide GPS devices and connect them to the application.

• The administrator has the authority to create, modify, and delete user accounts.

• The administrator has the authority to create, modify, and delete car listings.

• Users should verify schedules before confirming them.

• Users must exercise caution when providing sensitive information such as Social Security Numbers or legal names, as errors may occur and impact tax filings.

3.2.2.3 Development Requirements

The application will be developed using the JavaScript programming language, specifically leveraging the React Native framework. React Native facilitates cross-platform development, allowing the app to be deployed on both iOS

and Android devices. It provides a robust development environment for implementing app functionality and designing user interfaces.

Firebase will be utilized to construct the application's database. Firebase, as a backend-as-a-service platform, offers real-time data synchronization, authentication, and other essential features, making it ideal for building dynamic and scalable mobile applications. Its cloud-based architecture and seamless integration capabilities can be leveraged to enhance the app's performance.

Incremental testing will be conducted throughout the development process to ensure that the app meets its objectives and specifications at each stage. This includes performance testing to evaluate speed, responsiveness, and resource utilization, as well as functional testing to verify that features operate as intended. Comprehensive testing across various usage scenarios will help identify and address any issues or faults early in development.

Regular maintenance checks will be performed to uphold the application's ongoing performance and stability. This involves monitoring for potential performance bottlenecks, security vulnerabilities, and software defects, and resolving them promptly. Maintenance checks may include software updates, security patches, database optimization, and overall system health evaluations to ensure smooth and efficient operation of the app.

3.2.3 External Requirements

The Car Rental software application and its services must adhere fully to the regulatory, ethical, and legal requirements of The Republic of Albania.

The Car Rental application system must ensure that all vehicles offered for rental meet the technical standards for vehicle safety and emissions as mandated by the Albanian government. Additionally, the software must be designed in accordance with personal data protection regulations to safeguard the security and confidentiality of customer information.

Furthermore, the system should be capable of generating accurate and comprehensive financial reports. Lastly, the application software should comply with consumer protection regulations, ensuring that rental agreement terms and conditions are transparent and easily understandable for customers.

3.2.3.1 Regulatory Requirements:

- Mandatory Insurance in the Transport Sector (no. 10076, dated 29.05.2009)

- Legislation for Merchants and Trade Companies (no. 9901, dated 17.04.2008)

- Personal Data Protection Law (no. 9887, dated 10.03.2008)

- Electronic Communications Law (no. 9918, dated 19.05.2008)

- Legislation for Licenses, Authorizations, and Permits in The Republic of Albania (no. 10 081, dated 23.02.2009)

- Consumer Protection Law (no. 9902, dated 17.04.2008)

3.2.3.2 Ethical Requirements:

- Albanian Constitution, Article 18

- Personal Data Protection Law (no. 9887, dated 10.03.2008)

3.2.3.3 Legislative Requirements:

- Road Code of The Republic of Albania (no. 8378, dated 22.07.1998)

- Value Added Tax Law (no. 92/2014, dated 24.07.2014)

- Income Tax Law (no. 8438, dated 28.12.1998)

3.2.3.3.1 Accounting requirements

Currently, there are no accounting requirements that we are obligated to follow. This situation is not expected to change, so we will not be implementing any accounting requirements at this time.

3.2.3.3.2 Security requirements

The personal and sensitive information stored by the application will be handled in full compliance with the legislation of the Republic of Albania regarding digital security. The Commission for Personal Data Protection will ensure that the application is secure and ready for deployment to any company interested in using it.

3.3 Domain Requirements

Our application operates within the car rental industry, and thus, it is expected to encompass various users and

features typical to this domain. While some of these requirements may overlap with those previously outlined, this section serves to highlight the specific industry-driven needs and facilitate comparison with standard industry requirements.

Domain requirements regarding Users:

The car rental application should be tailored to meet the demands of both drivers and rental companies.

- Drivers: These users could include leisure or business travelers, as well as individuals in need of personal transportation. They require an intuitive interface for browsing and reserving rental vehicles, transparent pricing details, and access to essential features like GPS tracking and roadside assistance.
- Rental companies: Whether small-scale operations or large enterprises, rental companies necessitate a management portal to manage vehicle inventory, pricing structures, and bookings efficiently. Additional functionalities such as data analytics and reporting can aid in optimizing operations and enhancing customer satisfaction.

Common features:

The application should offer an array of features to address the requirements of both drivers and rental companies.

- Vehicle search and booking: Drivers should be able to locate and reserve vehicles based on their preferences and requirements, facilitated by a straightforward and secure booking process.
- Payment processing: A secure payment gateway supporting various payment methods and currencies is essential, ensuring seamless transactions and invoicing.
- Customer reviews and ratings: Enabling drivers to provide feedback through reviews and ratings contributes to service improvement for rental companies and enhances user experience.
- Car availability and pricing information: Timely updates on vehicle availability, pricing details, and any promotional offers are crucial for informed decision-making.
- GPS tracking for rental vehicles: Integration with GPS tracking systems enables real-time tracking of rental vehicles, benefiting both drivers and rental companies by enhancing security and facilitating vehicle recovery if needed.

4 User Scenarios/ Use Case

4.1 User Scenarios

UC Name	*001 Sign In*
Summary	*Users have the ability to sign in to the system using their credentials*

Dependency	*The device should be connected to the internet.*
Actors	*All users*
Preconditions	*Users must have a valid account existing in the system database.*
Description of the Main Sequence	● *Step 1: Go to the sign in page* ● *Step 2: Put in your username and password* ● *Step 3: Authentication and sign in process*
Description of the Alternative Sequence	● *Step 1: Go to the sign in page* ● *Step 2: Put in your License ID and password* ● *Step 3: Authentication and sign in process*
Non functional requirements	*The sign in process must be quick and secure, with strong encrypted passwords and safe data storing. The device should be connected to the internet.*
Postconditions	*If the sign In process runs smoothly, users should be brought to the main page of the system.*

UC Name	***002 Sign Up using DL/ID***
Summary	*First time customers can sign up using their ID number or DL number as an identification tool.*
Dependency	*None*
Actors	*Primary actor: Customer*
Preconditions	*These users must be signing up for the first time.*
Description of the Main Sequence	● *Step 1: Customers will go to the main page and click the sign up button* ● *Step 2: The system will then ask users for necessary data to login* ● *Step 3: System will validate given data to guarantee their accuracy* ● *Step 4: User will be logged in the system and will receive credentials via email.*
Non functional	*Identification tool must be real and valid.*

requirements	
Postconditions	*If the sign up process runs smoothly, users should be brought to the main page of the system.*

UC Name	***003 View all Reservations***
Summary	*Car Maintenance Operator is able to view all car reservations.*
Dependency	*Sign in*
Actors	*Primary Actors: Car Maintenance Operator* *Secondary Actors: Rental Agent, Administrator, Business Owner.*
Preconditions	*Car Maintenance Operator should be a valid user signed in the system.*
Description of the Main Sequence	● *Step 1: CMO signs in the system* ● *Step 2: CMO is brought to the main page on the CMO side* ● *Step 3: CMO clicks view reservation to find a list of reserved vehicles*
Description of the Alternative Sequence	● *Step 1: User signs into the system* ● *Step 2: Go to the menu of the website* ● *Step 3: click on view all reservations* ● *Step 4: view and navigate through all actual reservations.*
Non functional requirements	*CMO are not able to view sensitive information about customer reservations, such as DL number or ID number,* *etc.*
Postconditions	*All reservations are retrieved form the system and displayed to the user.*

UC Name	***004 Report when a car is returned***
Summary	*CMO is responsible for reporting when vehicles arrive at dropping point and picking them up.*
Dependency	*Sign in*
Actors	*Primary Actors: Car Maintenance Operator*

Preconditions	*Car Maintenance Operator should be a valid user signed in the system, and the car should be dropped off at specified location.*
Description of the Main Sequence	• *Step 1: CMO receives notifications when a car is ready for pick up* • *Step 2: CMO arrives at destination and picks up assigned car* • *Step 3: CMO drives car to another location if needed* • *Step 4: CMO marks car as returned in the system*
Description of the Alternative Sequence	• *Step 1: CMO receives notifications when a car is ready for pick up* • *Step 2: Car is already at specified location* • *Step 3: CMO marks car as returned in the system.*
Non functional requirements	*CMO should have a valid driver's license.*
Postconditions	*Vehicle will be marked as returned in the system and will be shown as available for reservation on the client's side.*

UC Name	***005 Report damages or fines***
Summary	*CMO will check the car after every drop off for any damages or fines and add them in the system.*
Dependency	*Sign in*
Actors	*Primary Actors: Car Maintenance Operator*
Preconditions	*Car Maintenance Operator should be a valid user signed in the system, and the car should be received at a given location.*
Description of the Main Sequence	• *Step 1: CMO picks up car* • *Step 2: CMO checks car for any damages or fines* • *Step 3: CMO adds them to the system so that the customers pay extra for the damage* • *Step 4: CMO adds a note to customer's account to make the company aware if customer is a next time client.*
Description of the Alternative	• *Step 1: CMO picks up car* • *Step 2: CMO checks car for any damages or fines*

Sequence	• *Step 3: CMO adds no damages or fines to the system when the car is in the same condition as received.*
Non functional requirements	*CMO should take video and photo proof of before and after the car was dropped off and picked up to verify any potential damages.*
Postconditions	*Customers that have damaged the cars or received fines will pay a penalty fee to cover expenses and will be marked in the system.*

UC Name	***006 Browse and filter available cars***
Summary	*Users are able to view a catalog of available vehicles for reservation based on rates, model, mileage, etc.dropping point and picking them up.*
Dependency	*None*
Actors	*Primary Actor: Customers* *Secondary Actor: Rental Agent, Administrator, Business Owner*
Preconditions	*There should be at least 1 vehicle available for reservation, in order to create a list.*
Description of the Main Sequence	• *Step 1: Customer is in the main page of the customer side and they click ”Available Vehicles”* • *Step 2: Customer filters vehicles based on preferences* • *Step 3: Filtered cars are shown* • *Step 4: Click on cars to see more details.*
Description of the Alternative Sequence	• *Step 1: Rental Agent or Admins are in the main page and they click ”Available Vehicles”* • *Step 2: Rental Agent or Admins filter vehicles based on desired filter* • *Step 3: Filtered cars are shown* • *Step 4: Click on cars to see more details.*
Non functional requirements	*Available cars should contain real data about year of production, engine type, brand, mileage, prior damage and rates that should be visible to customers and updated frequently. Reserved cars will not be shown.*
Postconditions	*Vehicle will be marked as returned in the system and will be shown as available for reservation on the client's side.*

UC Name	*007 Select a car and make reservation*
Summary	*Customers cancel their existing reservations.*
Dependency	*Depends on the availability of cars in the system and reservation management features.*
Actors	*Primary Actor: Customers, Rental Agent*
Preconditions	*Users are logged into the system and the cars are listed in the system.*
Description of the Main Sequence	● *Customer selects a car from the list of available options.* ● *Customer provides reservation details (e.g., date, time).* ● *System confirms the reservation and updates availability.*
Description of the Alternative Sequence	● *If the selected car is not available:* ● *System notifies the customer about the unavailability.*
Non functional requirements	*User Interface, Reservation Management, Data Integrity.*
Postconditions	*Reservation is successfully made for the selected car.*

UC Name	*008 Cancel Reservation*
Summary	*Customers select a car from available options and make a reservation.*
Dependency	*Depends on the reservation management system and customer's reservation status.*
Actors	*Primary Actor: Customers, Rental Agent*
Preconditions	*Customer has an existing reservation.*

Description of the Main Sequence	• *Customer accesses the reservation management section.* • *Customer selects the reservation to cancel.* • *System confirms the cancellation and updates availability.*
Description of the Alternative Sequence	• *If the reservation cannot be canceled (e.g., already past cancellation deadline):* • *System notifies the customer about the inability to cancel.*
Non functional requirements	*User Interface, Reservation Management, Data Integrity.*
Postconditions	*Reservation is successfully canceled*

UC Name	***009 View car specifications***
Summary	*Customers and rental agents view detailed specifications of available cars*
Dependency	*Depends on the availability of car details in the system.*
Actors	*Primary Actor: Customers, Rental Agent*
Preconditions	*Customer or Rental Agent is logged into the system. Available cars have detailed specifications listed.*
Description of the Main Sequence	• *Customer or Rental Agent navigates to the car details section.* • *System displays specifications for each available car.*
Non functional requirements	*User Interface, Reservation Management, Data Integrity.*
Postconditions	*Car specifications are successfully viewed.*

UC Name	***010 View rental history***
Summary	*Customers and rental agents view rental history information.*
Dependency	*Depends on the availability of rental history data in the system.*
Actors	*Primary Actor: Customers, Rental Agent*

Preconditions	*Customer or Rental Agent is logged into the system. Rental history data is available and accessible.*
Description of the Main Sequence	● *Customer or Rental Agent navigates to the rental history section.* ● *System displays rental history information (e.g., past rentals, dates).*
Non functional requirements	*User Interface, Reservation Management, Data Integrity.*
Postconditions	*Rental history information is successfully viewed.*

UC Name	***011 Provide Feedback and Ratings***
Summary	*Customers provide feedback and ratings for their rental experiences.*
Dependency	*Depends on the feedback and rating submission features in the system.*
Actors	*Primary Actor: Customers*
Preconditions	*Customer has completed a rental.*
Description of the Main Sequence	● *Customer accesses the feedback and rating submission section.* ● *Customer provides feedback (comments, ratings).* ● *System records the feedback and updates ratings.*
Non functional requirements	*User Interface, Reservation Management, Data Integrity.*
Postconditions	*Feedback and ratings are successfully submitted and recorded.*

UC Name	***012 Pay via Credit Card or Cash***
Summary	*Customers make payments for their rentals using credit card or cash.*
Dependency	*Depends on the payment processing features and options in the system.*

Actors	*Primary Actor: Customers, Rental Agents*
Preconditions	*Customer has a completed reservation.*
Description of the Main Sequence	● *Customer navigates to the payment section.* ● *Customer selects payment method (credit card or cash).* ● *System processes the payment and updates payment status.*
Non functional requirements	*Payment Processing, Data Integrity.*
Postconditions	*Payment is successfully processed using the chosen method.*

UC Name	***013* Add Receipts for Reservations**
Summary	This use case involves generating and attaching receipts to reservations made by rental agents for the customers.
Dependency	This use case depends on successful reservation creation and payment processing.
Actors	*Primary Actor: Rental Agent* *Secondary Actor: Admin*
Preconditions	Reservation must be created, and payment must go through.
Description of the Main Sequence	● Step 1: The rental agent or administrator accesses the reservation details. ● Step 2: The agent generates a receipt containing transaction information. ● Step 3: The agent attaches the generated receipt to the reservation record.
Description of the Alternative Sequence	● Step 1: The rental agent or administrator accesses the reservation details. ● Step 2: The agent generates a receipt containing transaction information.

	• Step 3: The agent attaches the generated receipt to the reservation record. • Step 4: The rental agent or administrator prints the receipt.
Non functional requirements	The system should generate receipts without any errors.
Postconditions	A receipt is added to the reservation, providing a record of the transaction.

UC Name	***014*** **Send receipts to customer via email**
Summary	This use case involves emailing generated receipts to customers after they successfully reserve.
Dependency	Depends on the successful generation of receipts.
Actors	*Primary Actor: System, Rental Agent*
Preconditions	Receipts must be generated.
Description of the Main Sequence	• Step 1: The system initiates the process of sending a receipt to the customer via email. • Step 2: The system attaches the generated receipt to the email. • Step 3: The system sends the email containing the receipt to the customer's provided email address.
Description of the Alternative Sequence	• Step 1: The agent generates a receipt containing transaction information. • Step 2: The agent adds the email of the customer. • Step 3: The system sends the email containing the receipt to the customer's provided email address.
Non functional requirements	The system should send emails as quickly as possible, and ensure receipts are delivered without errors.
Postconditions	Customer receives the receipt via email, providing a digital record of the transaction.

UC Name	***015*** **Process Payment**
Summary	This use case involves handling payments being made for reservations by customers.
Dependency	Depends on successful reservation creation and available payment methods.
Actors	*Primary Actor: System, Customers*
Preconditions	Customers must choose a payment method and they should continue with correct payment credentials.
Description of the Main Sequence	● Step 1: The customer selects a payment method. ● Step 2: The customer enters payment details. ● Step 3: The customer confirms the transaction. ● Step 4: The system securely processes the payment.
Non functional requirements	The payment processing system should be secure and capable of handling various payment methods.
Postconditions	Payment is successfully processed, and the reservation is automatically confirmed.

UC Name	***016*** **Access customer information**
Summary	This use case involves retrieving and viewing customer information stored in the system.
Dependency	Depends on user authentication and existing customer records
Actors	*Primary Actor: Rental Agent, Admin*
Preconditions	Users must be signed in as a rental agent or admin.
Description of the Main Sequence	● Step 1: The rental agent or administrator navigates to the customer information section. ● Step 2: The user searches for the desired customer within the system. ● Step 3: The user selects the customer from the search results. ● Step 4: The system retrieves and displays the customer's information.

Non functional requirements	The system should retrieve customer information swiftly and present it in a user-friendly interface.
Postconditions	Rental agent or administrator successfully views customer information.

UC Name	***017* Check car location through GPS**
Summary	This use case involves retrieving the real-time location of rental vehicles using GPS technology.
Dependency	Depends on GPS functionality and availability of GPS data for rental vehicles.
Actors	*Primary Actor: Rental Agent, Admin*
Preconditions	Rental vehicles must be equipped with GPS tracking devices.
Description of the Main Sequence	• Step 1: The rental agent or administrator accesses the GPS tracking feature within the system. • Step 2: The user enters the vehicle identifier. • Step 3: The system retrieves the real-time location of the vehicle using GPS technology.
Non functional requirements	The GPS tracking system should provide accurate and up-to-date location information with minimal delay. The system will get alerts if the speed of the car exceeds a limit.
Postconditions	Rental agent or administrator successfully retrieves the location of the rental vehicle.

UC Name	***018 Notify customers of updates***
Summary	This use case involves sending automated notifications to customers regarding updates related to their reservations or the rental process, and also discounts.
Dependency	Depends on reservation status changes and customer contact information.
Actors	*Primary Actor: System*

Preconditions	Reservation status must change and updates must be available.
Description of the Main Sequence	● Step 1: The system identifies updates or changes relevant to the customer's reservation. ● Step 2: The system prepares automated notifications containing the relevant information. ● Step 3: The system sends the notifications to the customers via email or SMS.
Non functional requirements	The notification system should be reliable, capable of delivering notifications promptly and accurately.
Postconditions	Customers receive notifications regarding updates to their reservations or rental process

UC Name	***019 Add user***
Summary	*Administrators add new users to the system.*
Dependency	*Depends on the user management features and permissions in the system.*
Actors	*Primary Actor: Admin*
Preconditions	*Administrator is logged into the system. Access rights and permissions for user addition are granted.*
Description of the Main Sequence	● Administrator navigates to the user management section. ● Administrator selects the option to add a new user ● Administrator provides user details (e.g., name, role, credentials).

	• System validates and adds the new user.
Non functional requirements	*User Management, Data Integrity.*
Postconditions	*New user is successfully added to the system.*

UC Name	***020 Delete user***
Summary	*Administrators remove users from the system.*
Dependency	*Depends on the user management features and permissions in the system.*
Actors	*Primary Actor: Admin*
Preconditions	*Administrator is logged into the system. Access rights and permissions for user addition are granted.*
Description of the Main Sequence	• Administrator navigates to the user management section. • Administrator selects the user to be deleted • System confirms the deletion action • The user is removed from the system
Non functional requirements	*User Management, Data Integrity.*
Postconditions	*User is successfully removed from the system.*

UC Name	***021 Modify user***
Summary	*Administrators modify user information in the system.*
Dependency	*Depends on the user management features and permissions in the system.*
Actors	*Primary Actor: Admin*
Preconditions	*Administrator is logged into the system. Access rights and permissions for user addition are granted.*

Description of the Main Sequence	● Administrator navigates to the user management section. ● Administrator selects the user to be modified. ● Administrator updates user details (e.g., name, role, credentials). ● System validates and updates the user information.
Non functional requirements	*User Management, Data Integrity.*
Postconditions	*User information is successfully modified in the system.*

UC Name	***022 Manage Employees***
Summary	*Administrators manage employee information in the system.*
Dependency	*Depends on the employee management features and permissions in the system.*
Actors	*Primary Actor: Admin*
Preconditions	*Administrator is logged into the system. Access rights and permissions for user addition are granted.*
Description of the Main Sequence	● Administrator navigates to the employee management section. ● Administrator selects the employee to manage (e.g., add, delete, modify). ● Administrator performs the desired action (e.g., add employee). ● System validates and updates employee information.
Non functional requirements	Employee Management, Data Integrity.
Postconditions	*Employee information is successfully managed in the system.*

UC Name	***023 Add Car***
Summary	*Admin adds new cars to the system.*
Dependency	*Depends on the car management features and permissions in the system.*
Actors	*Primary Actor: Admin*

Preconditions	*Admin is logged into the system. Access rights and permissions for car addition are granted.*
Description of the Main Sequence	● Admin navigates to the car management section. ● Admin selects the option to add a new car. ● Admin provides car details (e.g., make, model, year). ● System validates and adds the new car.
Non functional requirements	Car Management, Data Integrity.
Postconditions	*New car is successfully added to the system.*

UC Name	***024 Remove Car***
Summary	*Admin removes cars from the system.*
Dependency	*Depends on the car management features and permissions in the system.*
Actors	*Primary Actor: Admin*
Preconditions	*Admin is logged into the system. Access rights and permissions for car addition are granted.*
Description of the Main Sequence	● Admin navigates to the car management section. ● Admin selects the car to be removed ● System confirms the removal action ● System removes the car from database.
Non functional requirements	Car Management, Data Integrity.
Postconditions	*The car is successfully removed from the system.*

UC Name	***025 Modify car information***
Summary	*Modify car specifications like color, price or pick up/drop off locations*
Dependency	*The car should be already in the database.*

Actors	*Primary actor: Admin*
Preconditions	*User must be signed in as administrator*
Description of the Main Sequence	● *Step 1: select the car you want to modify* ● *Step 2: click on the edit button* ● *Step 3: modify the wanted fields* ● *Step 4: click save.*
Non functional requirements	*System should handle format errors for the car fields.*
Postconditions	*A confirmation window pops up and the car info is changed or not according to the option clicked.*

UC Name	***026 Generate Reports***
Summary	The system empowers administrators to generate reports summarizing key metrics, such as rental revenue, popular car choices, and maintenance schedules and employer performance.
Dependency	*Depends on the reservations information.*
Actors	*Primary actor: Admin, System*
Preconditions	*Be signed in as admin. The system should be actively retrieving and sending data.*
Description of the Main Sequence	● *Step 1: Go to the reports menu.* ● *Step 2: Select what kind of report you want to view.* ● *Step 3: The system generates the report* ● *Step 4: View and optionally download the report.*
Non functional requirements	*Performance, security and reliability. The system should generate reports accurately to what is needed.*
Postconditions	*Reports will be ready to view or download.*

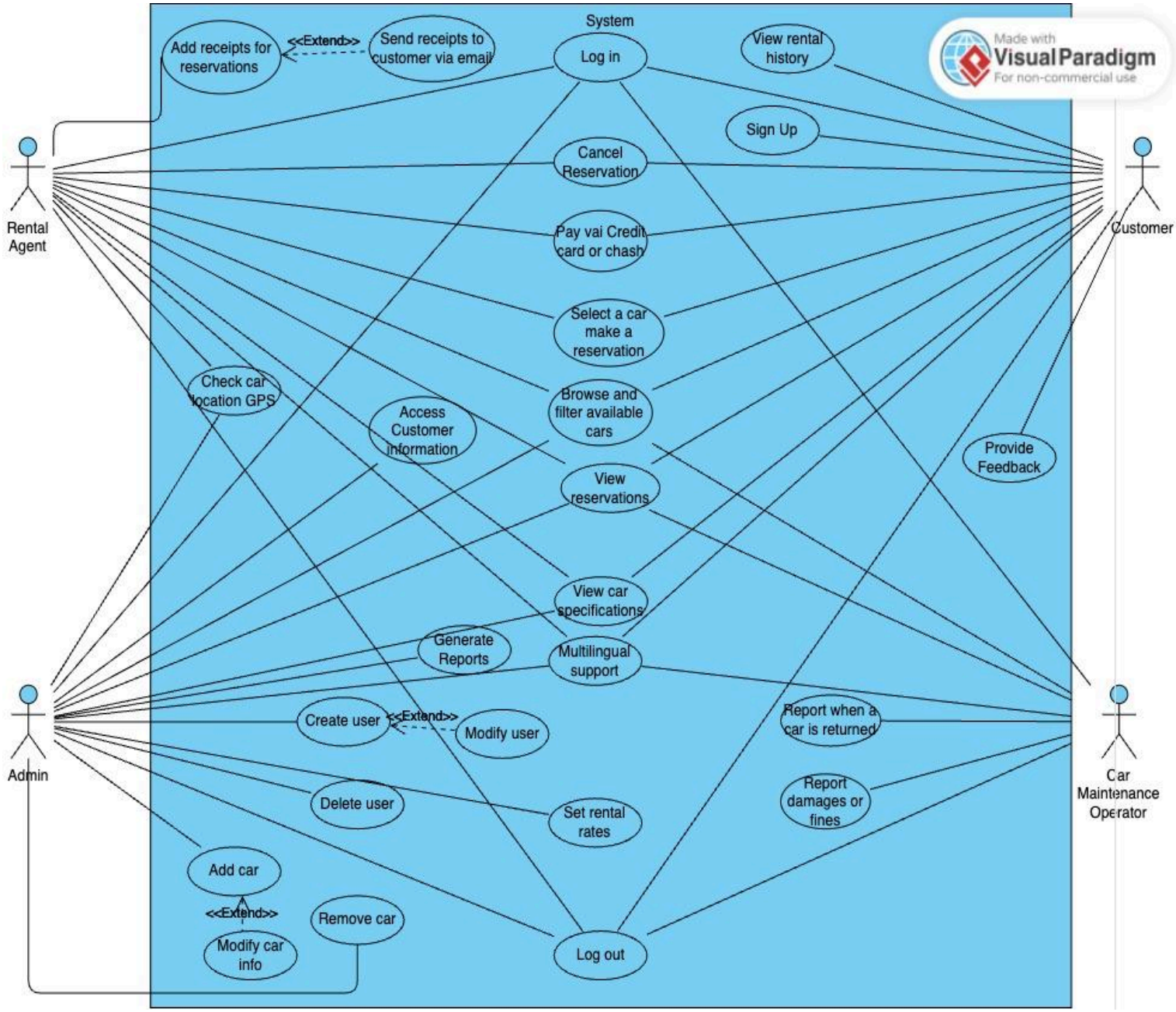
UC Name	***028 Set Rental Rates***

Summary	The system should empower administrators to set and modify rental rates for different car types, taking into account factors like peak seasons and promotional periods.
Dependency	*Market-prices and marketing strategies.*
Actors	*Primary actor: Admin*
Preconditions	*Be signed in as an administrator.*
Description of the Main Sequence	● *Step 1: Go to rental rates menu* ● *Step 2: Select the category of cars you want to change the price of* ● *Step 3: Set the percentage of the rise or the discount*
Non functional requirements	*Can be accessed only by admin. Retrieving data and assigning it to the database accurately.*
Postconditions	*Confirm the procedure and see the new prices assigned to the cars.*

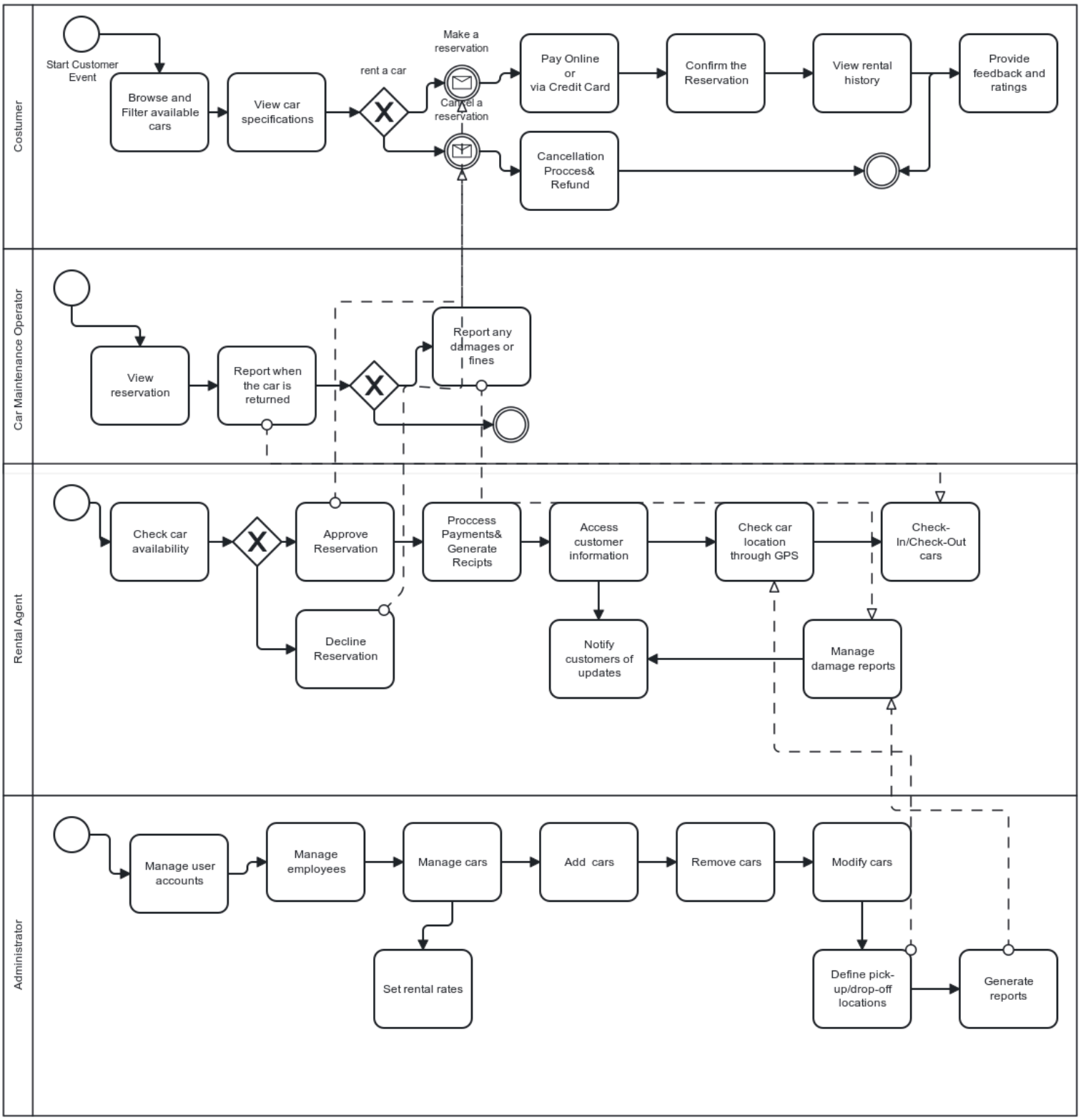
UC Name	***027 Multilingual support***
Summary	The system should have multiple languages so that native customers can navigate easily.
Dependency	Language translation resources and user preferences.
Actors	*Primary actor: Customer* *Secondary actor: all users*
Preconditions	*The system should have built in languages.*
Description of the Main Sequence	● *Step 1: User accesses user interface* ● *Step 2: User selects preferred language from available options* ● *Step 3: System displays interface elements and content in the selected language.*
Non functional requirements	Availability of language translation resources. Interface responsiveness to language changes.
Postconditions	*User interacts with the system in their preferred language.*

UC Name	*029 Log out*
Summary	The users log out of the system when they are not using it anymore.
Dependency	*User must be logged in*
Actors	*Primary actor: All users*
Preconditions	*Be signed in as a user.*
Description of the Main Sequence	• *Step 1: Go to the profile menu* • *Step 2: Click log out*
Non functional requirements	*The device should be connected to the internet.*
Postconditions	*The user is logged out of the system.*

4.2 General Use Case Diagram

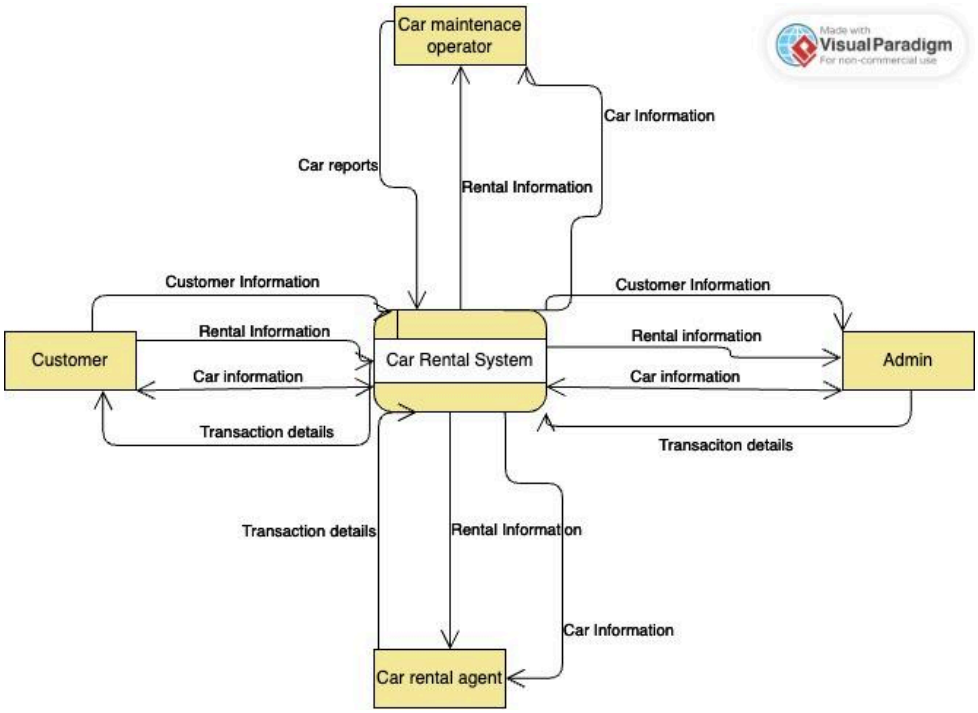

System
Add receipts for reservations
<<Extend>>
Send receipts to customer via email
Log in
View rental history
Made with VisualParadigm For non-commercial use
Sign Up
Cancel Reservation
Pay vai Credit card or chash
Select a car make a reservation
Check car location GPS
Access Customer information
Browse and filter available cars
Provide Feedback
View reservations
View car specifications
Generate Reports
Multilingual support
Create user
<<Extend>>
Modify user
Report when a car is returned
Delete user
Set rental rates
Report damages or fines
Add car
<<Extend>>
Modify car info
Remove car
Log out
Rental Agent
Customer
Admin
Car Maintenance Operator

4.3 BPMN

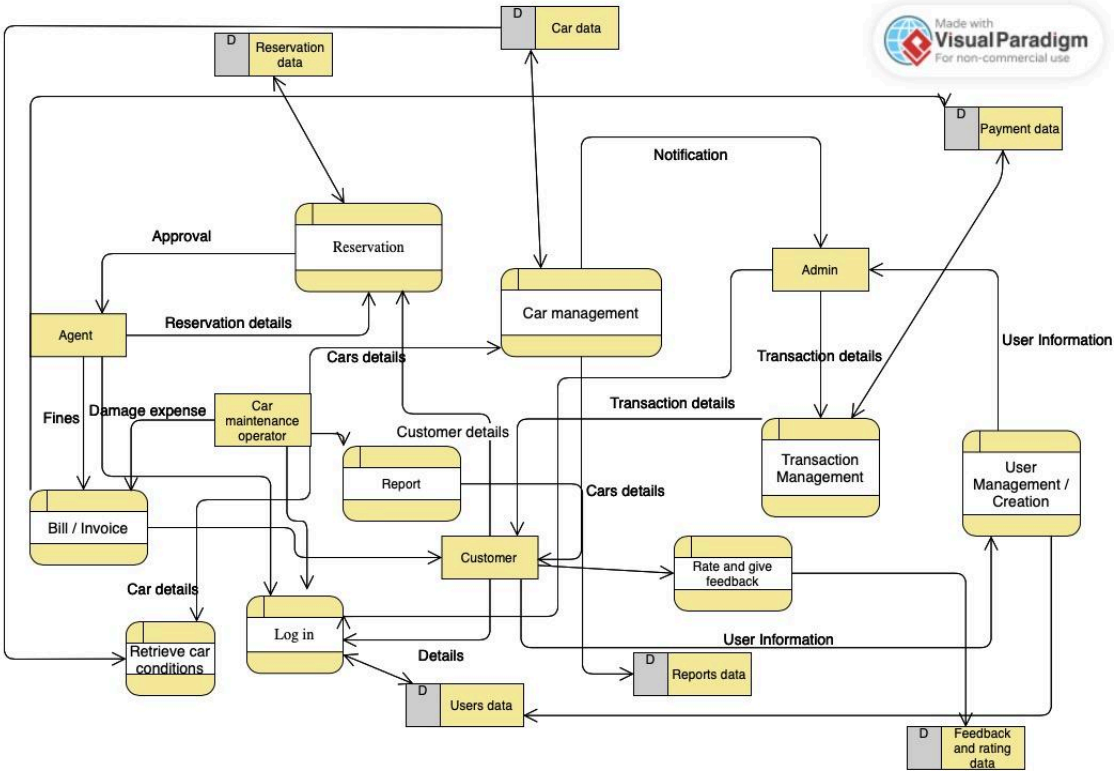


4.4 Data flow diagrams

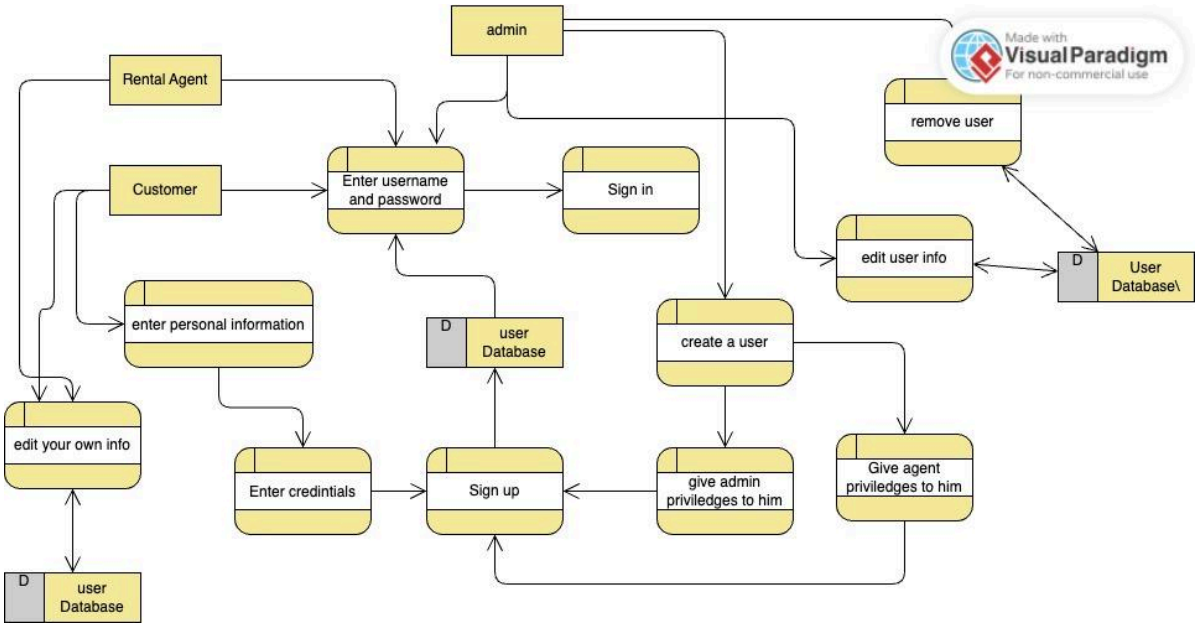
Level 0:



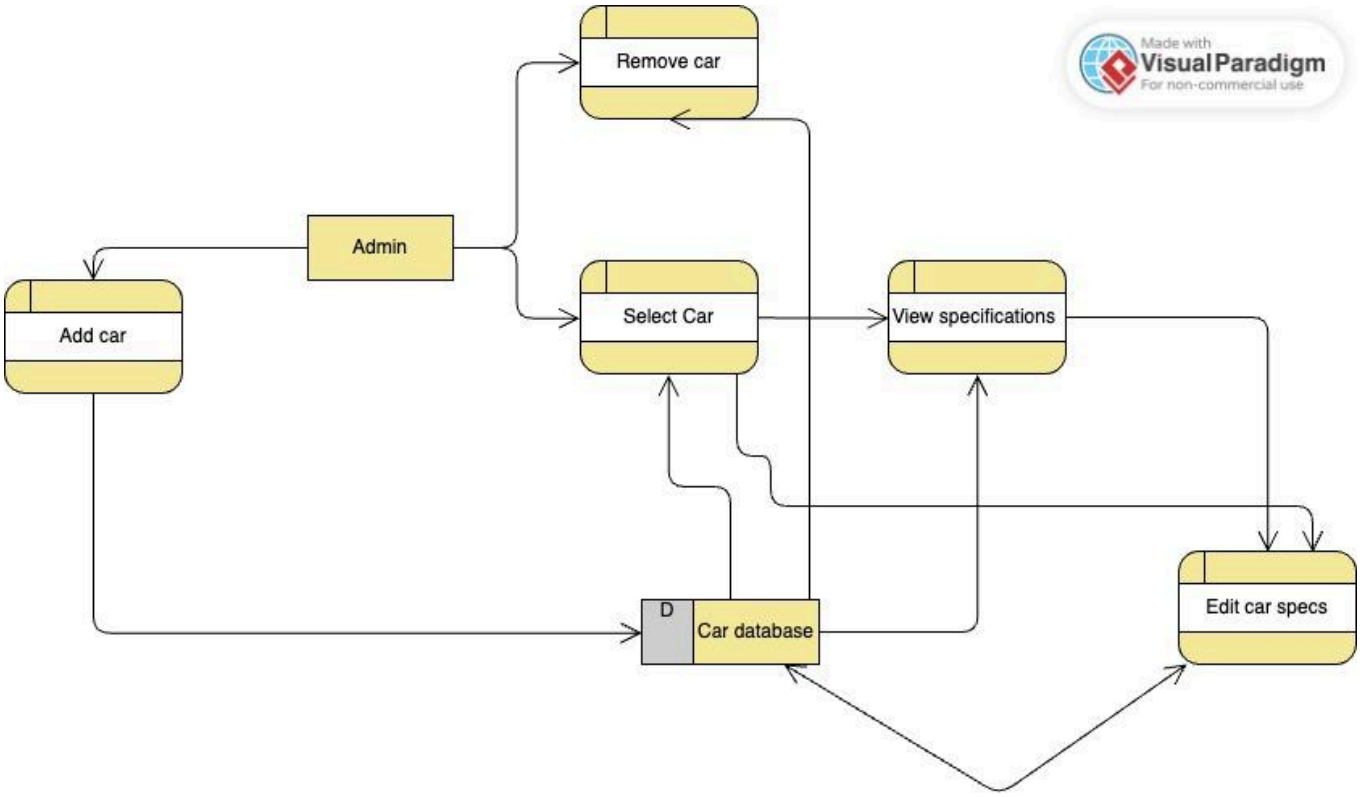
Level 1:



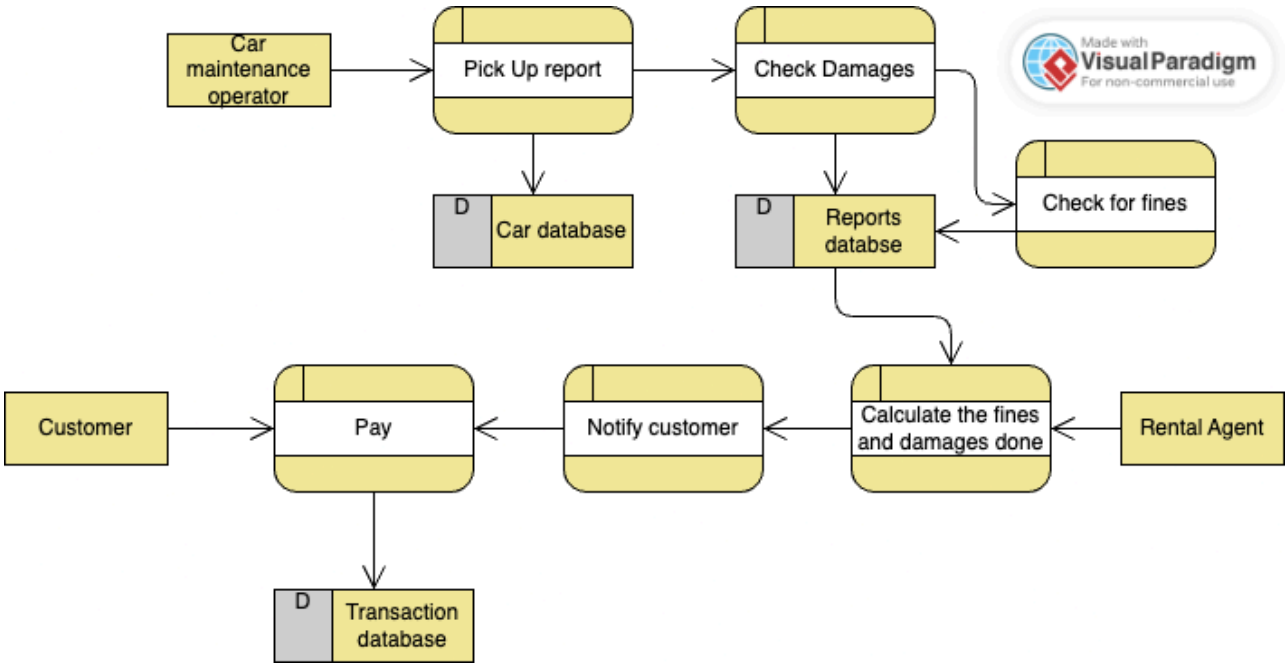
Level 2 (User management):



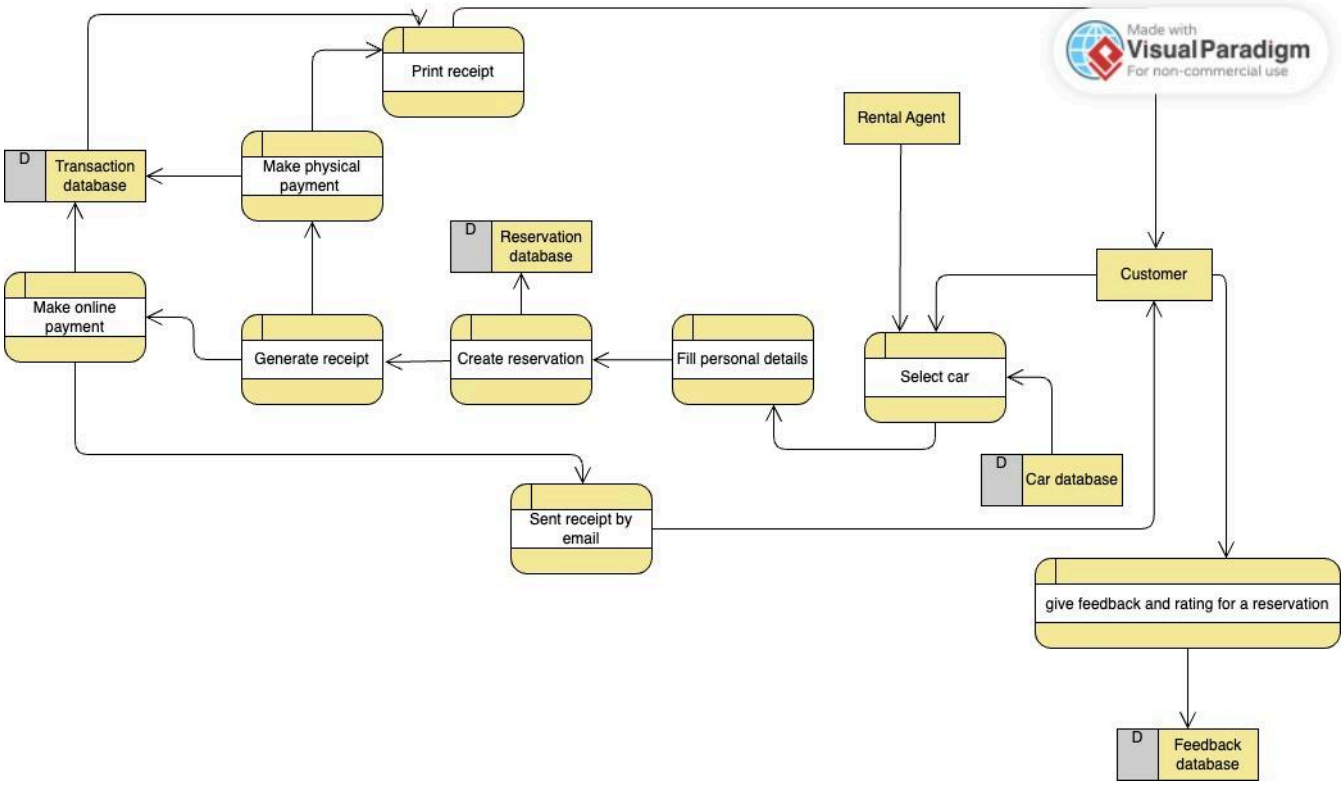
Level 2 (Car management):



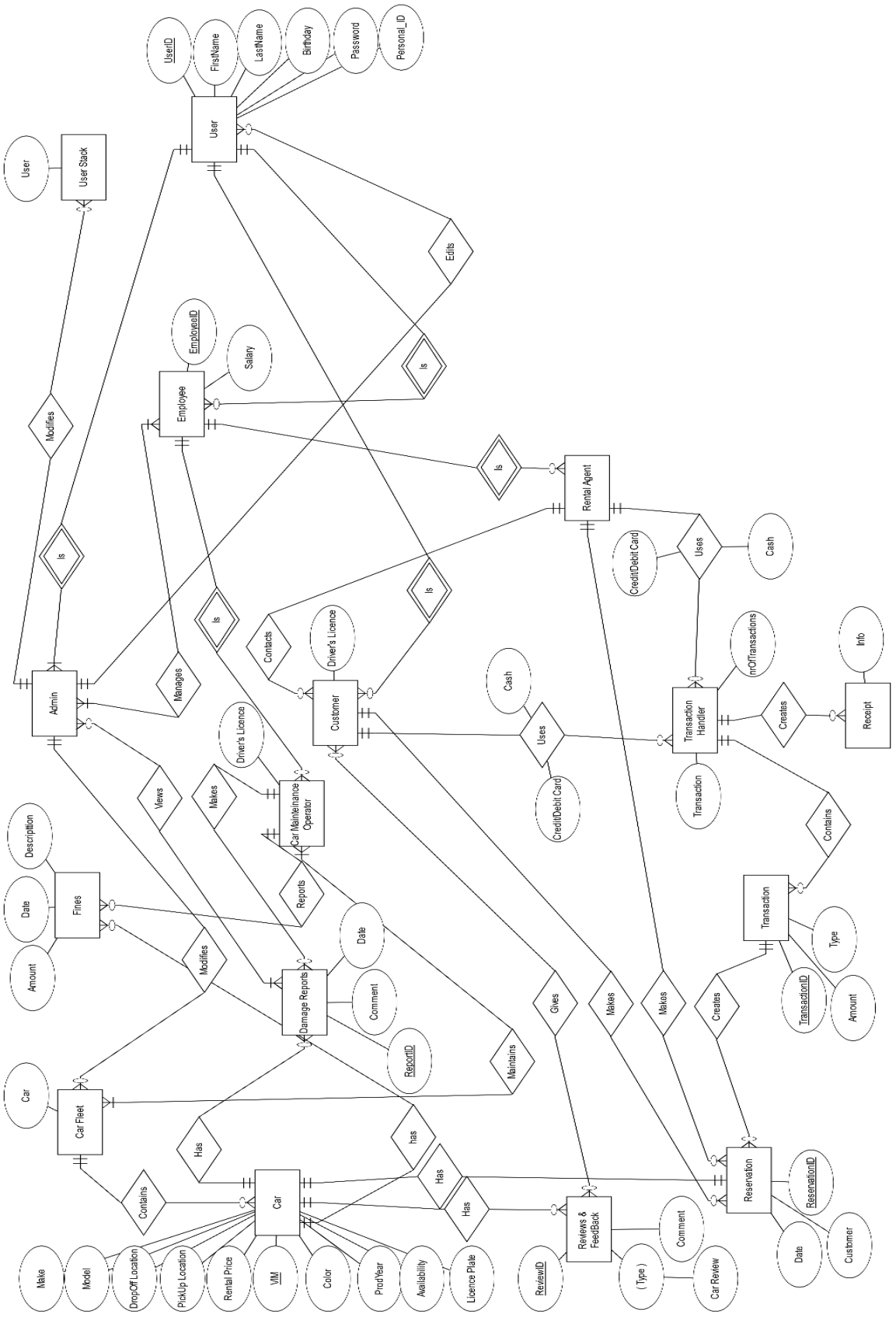
Level 2 (Report):



Level 2 (Reservation):

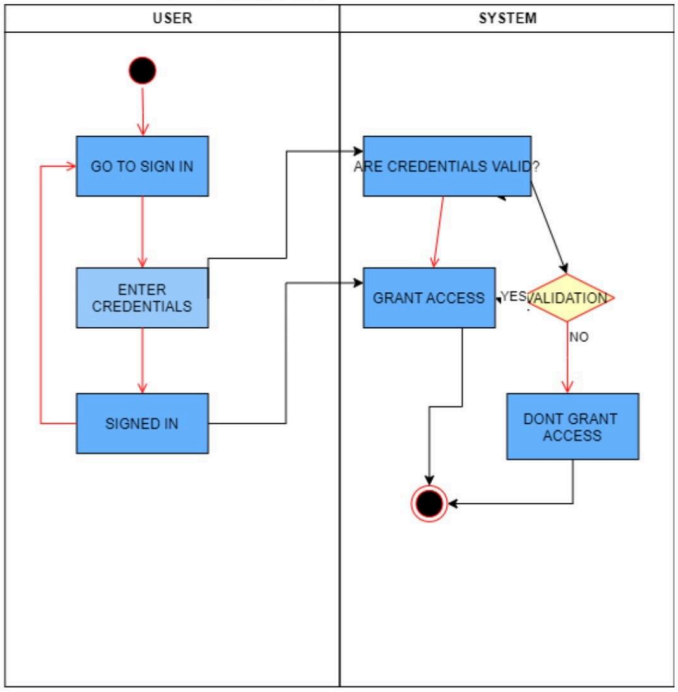


4.4 Er Diagram

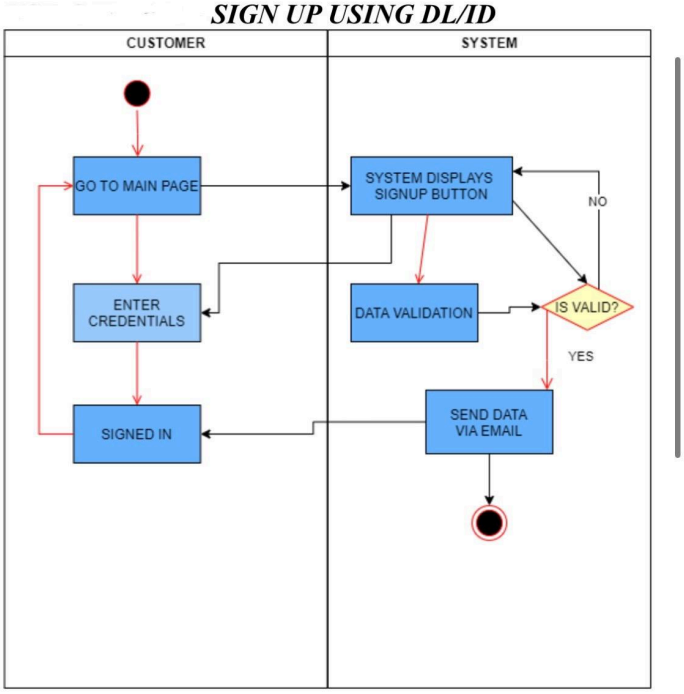


4.6 Activity Diagrams

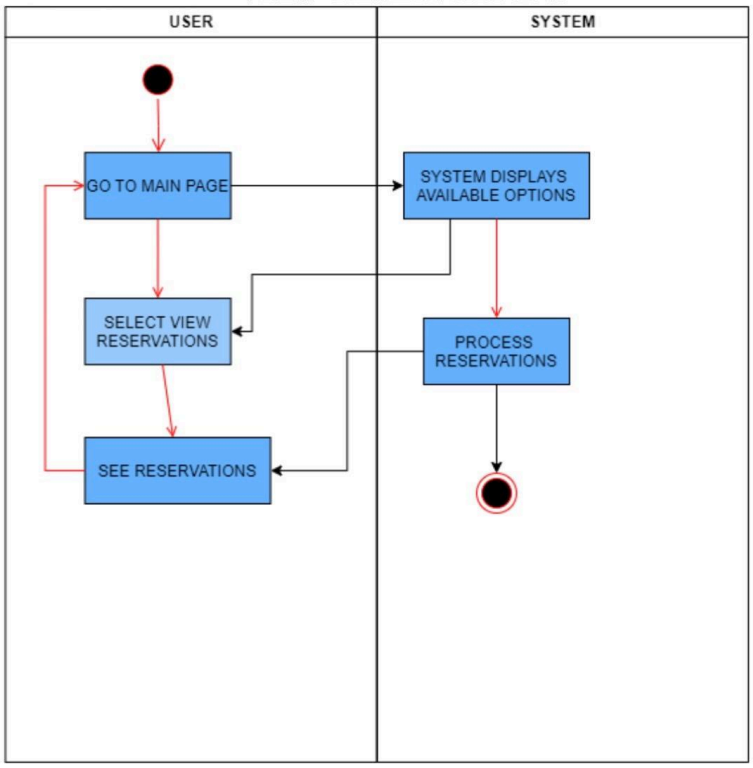
SIGN IN



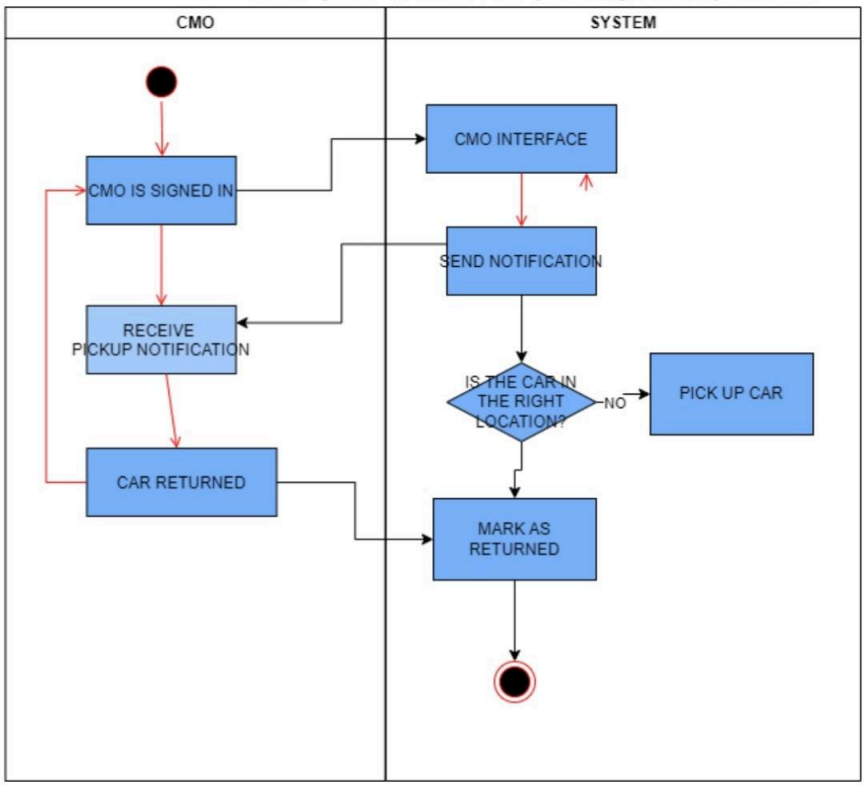
SIGN UP USING DL/ID

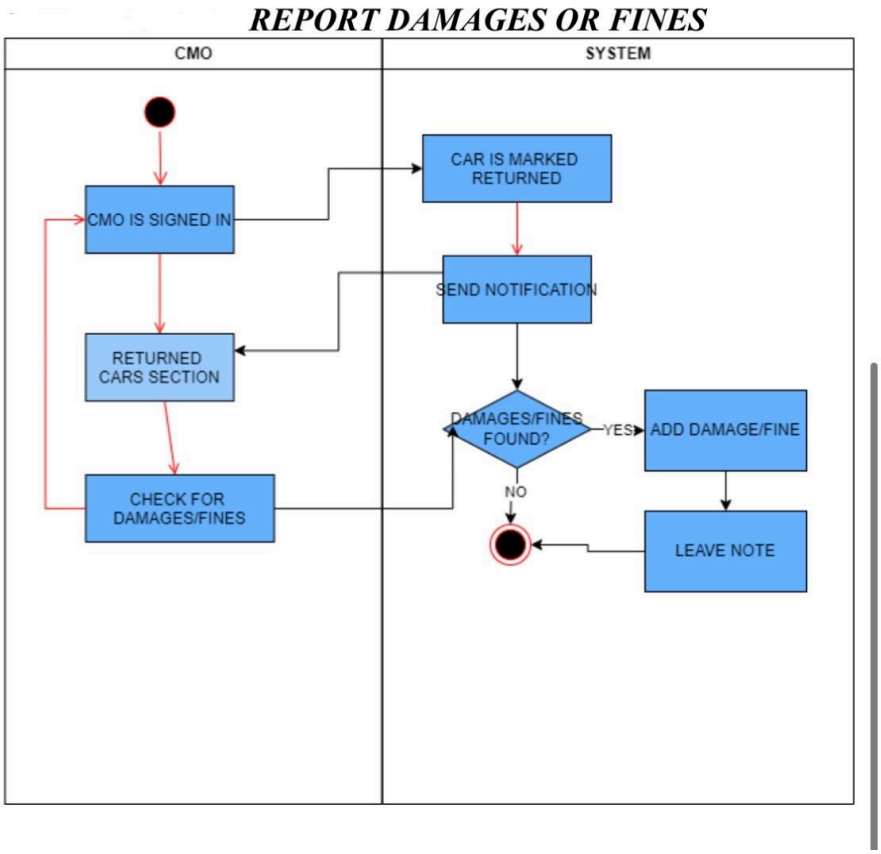


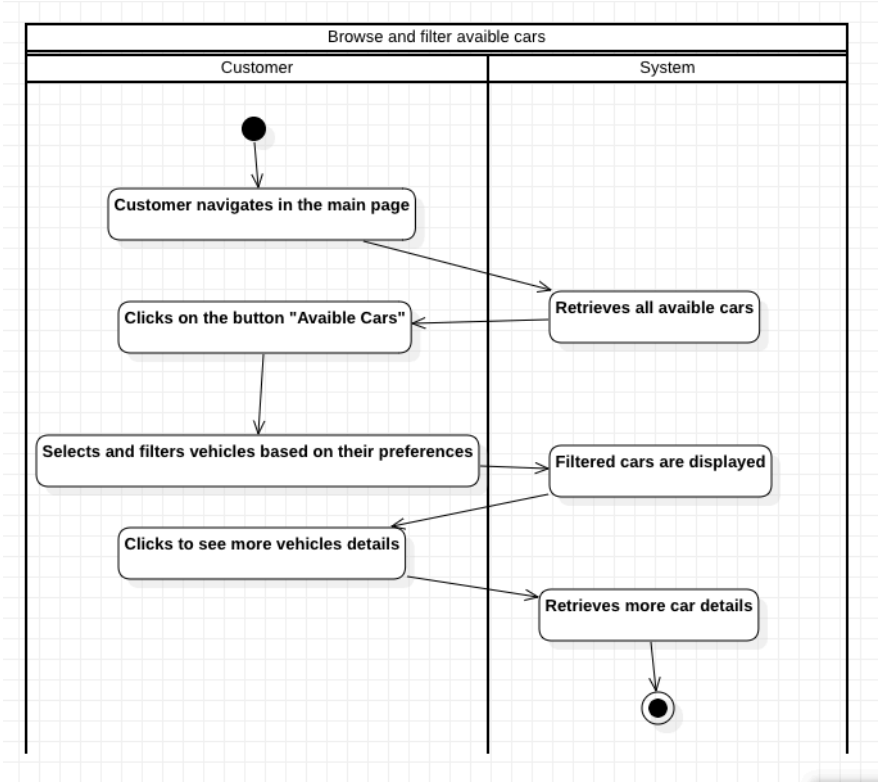
VIEW RESERVATIONS

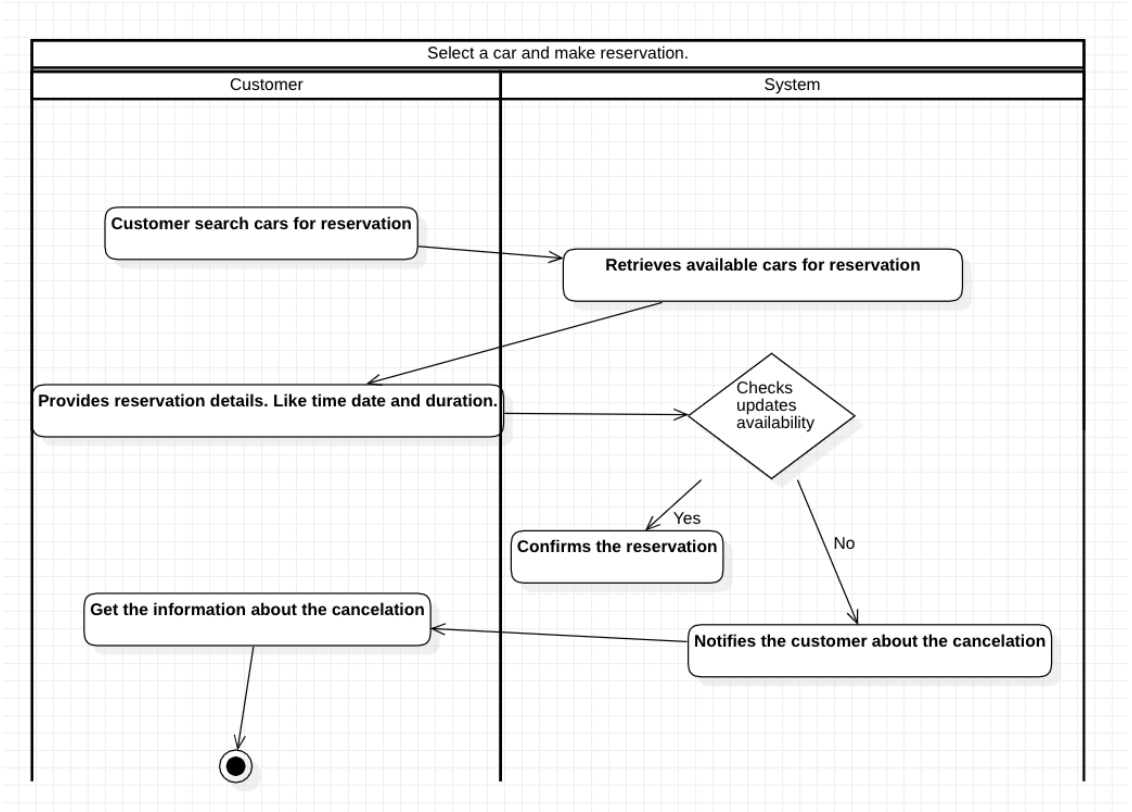


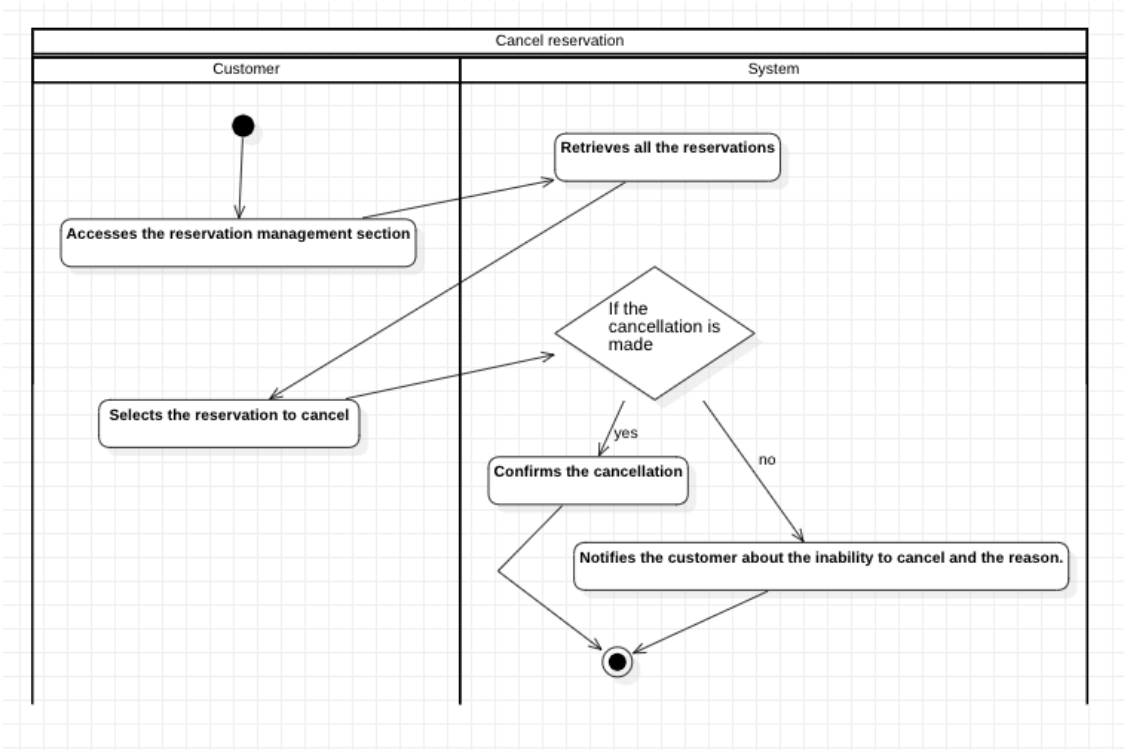
REPORT WHEN A CAR IS RETURNED

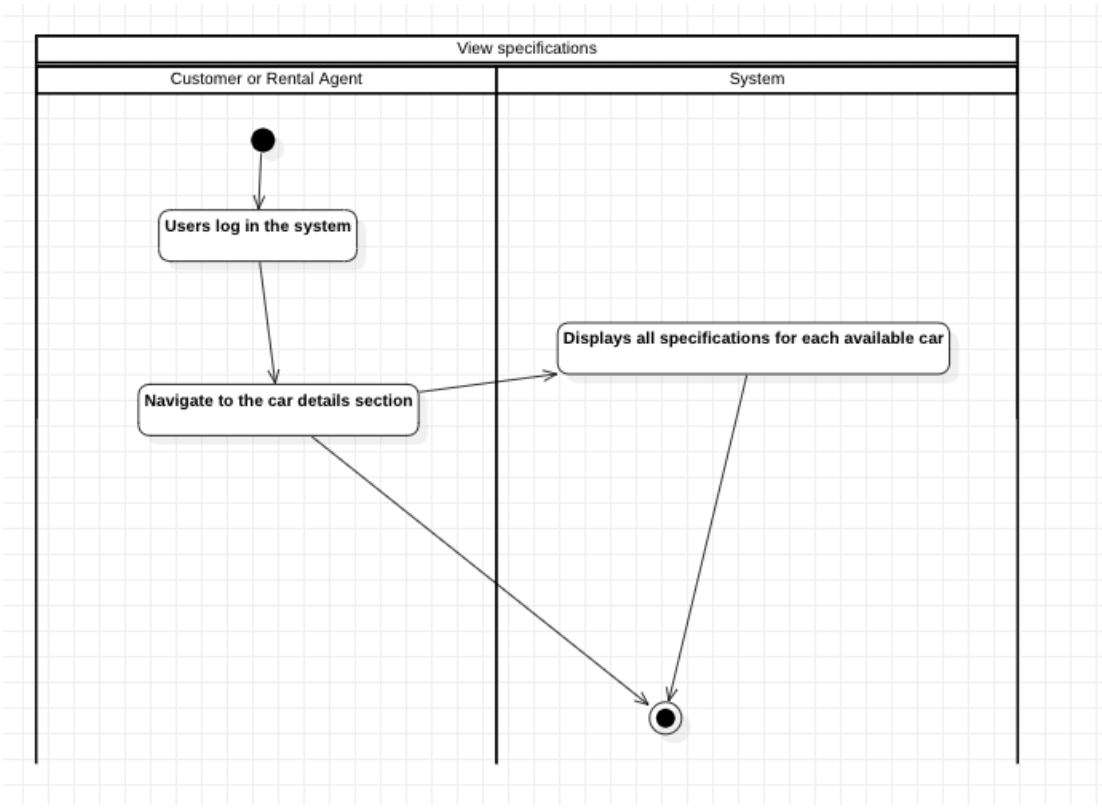


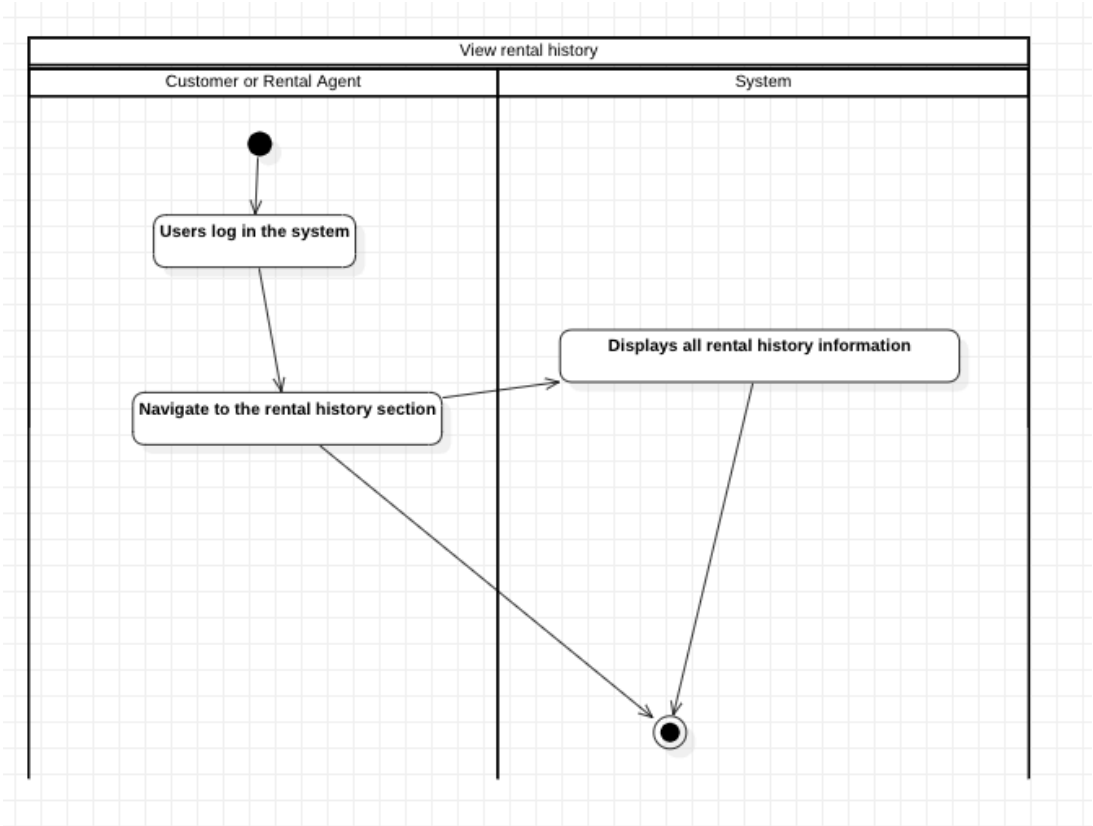

REPORT DAMAGES OR FINES
CMO
SYSTEM
CMO IS SIGNED IN
CAR IS MARKED RETURNED
SEND NOTIFICATION
RETURNED CARS SECTION
DAMAGES/FINES FOUND?
YES
ADD DAMAGE/FINE
NO
CHECK FOR DAMAGES/FINES
LEAVE NOTE


Browse and filter avaible cars
Customer
System
Customer navigates in the main page
Retrieves all avaible cars
Clicks on the button "Avaible Cars"
Selects and filters vehicles based on their preferences
Filtered cars are displayed
Clicks to see more vehicles details
Retrieves more car details

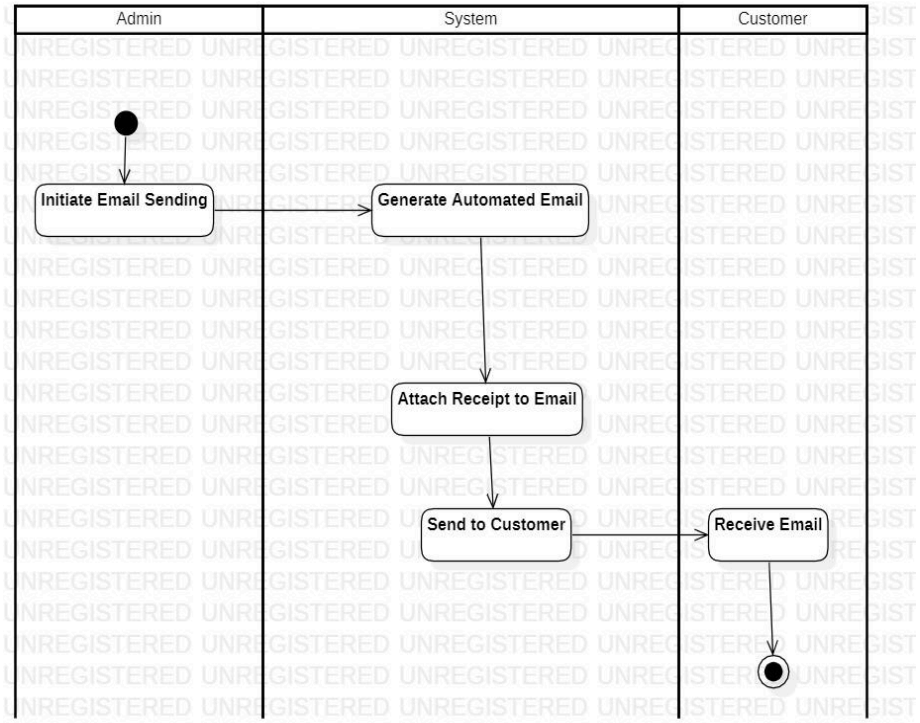

Select a car and make reservation.
Customer
System
Customer search cars for reservation
Retrieves available cars for reservation
Provides reservation details. Like time date and duration.
Checks updates availability
Yes
No
Confirms the reservation
Get the information about the cancelation
Notifies the customer about the cancelation


Cancel reservation
Customer
System
Retrieves all the reservations
Accesses the reservation management section
If the cancellation is made
Selects the reservation to cancel
yes
no
Confirms the cancellation
Notifies the customer about the inability to cancel and the reason.

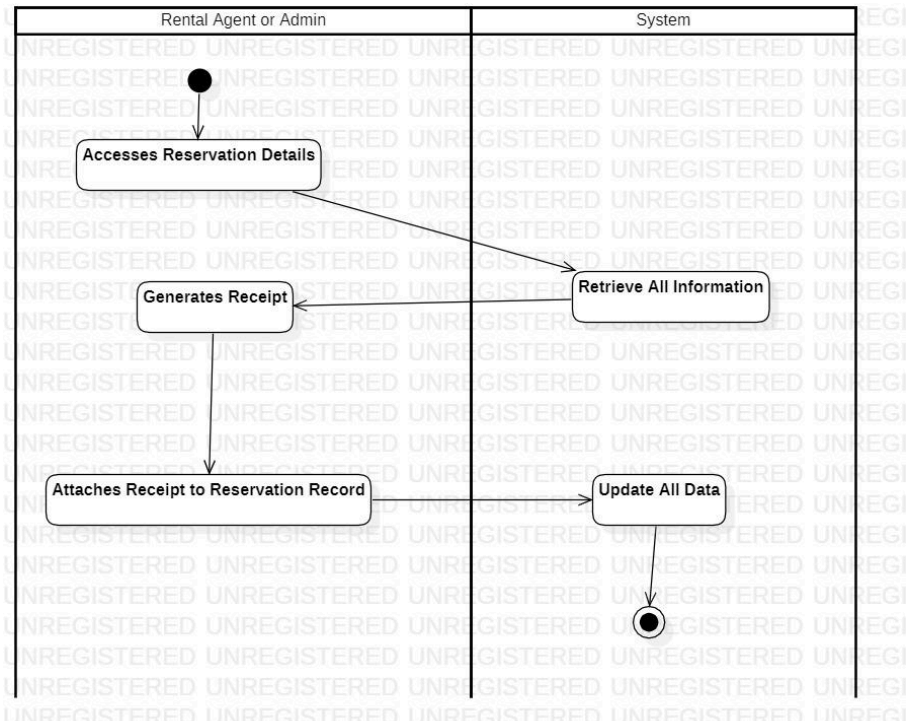

View specifications
Customer or Rental Agent
System
Users log in the system
Displays all specifications for each available car
Navigate to the car details section

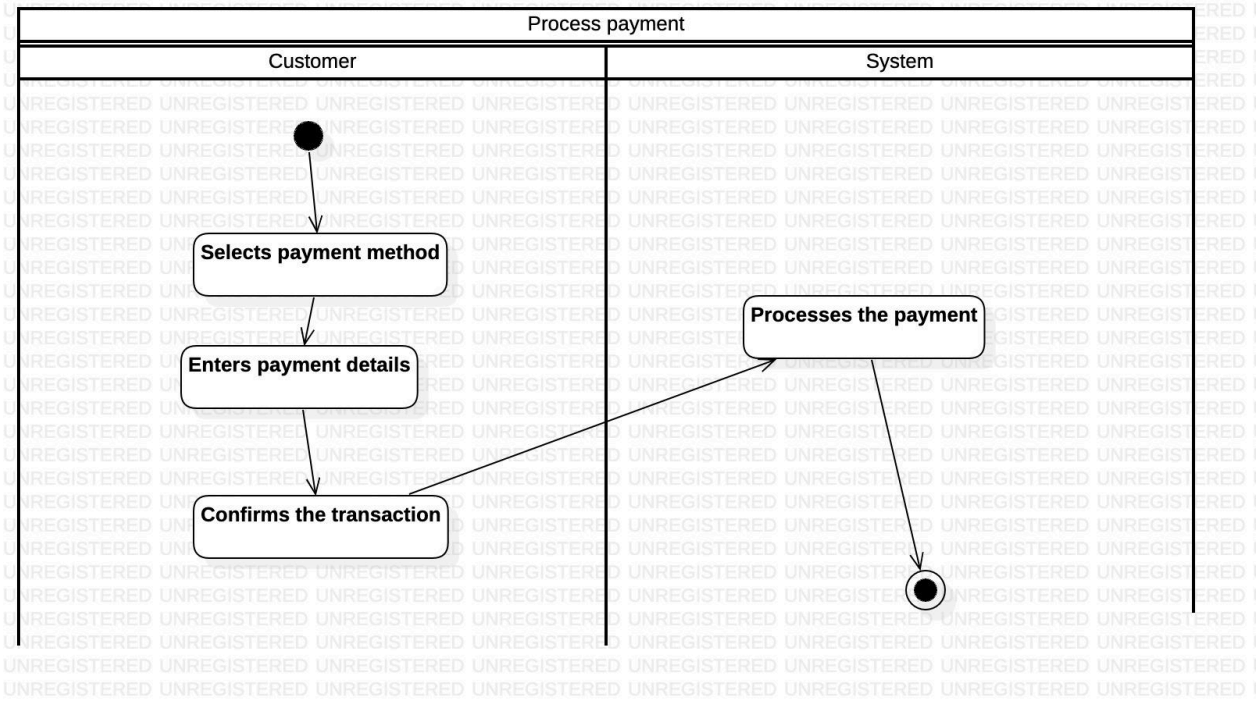

View rental history
Customer or Rental Agent
System
Users log in the system
Displays all rental history information
Navigate to the rental history section

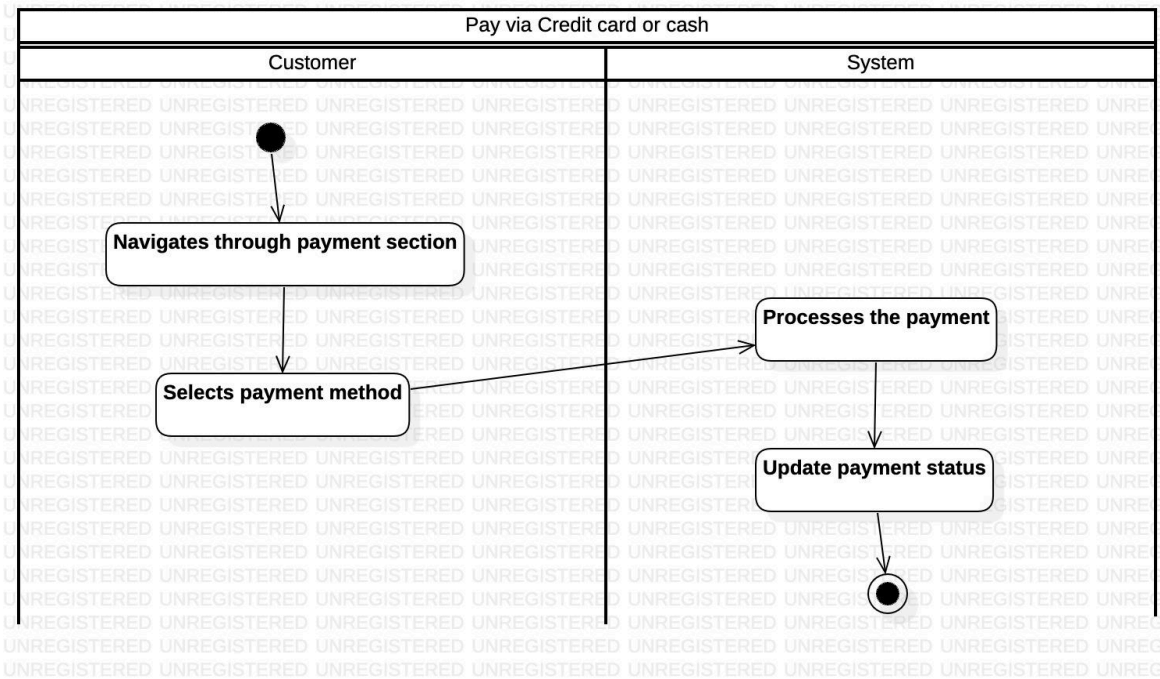
Generate receipt

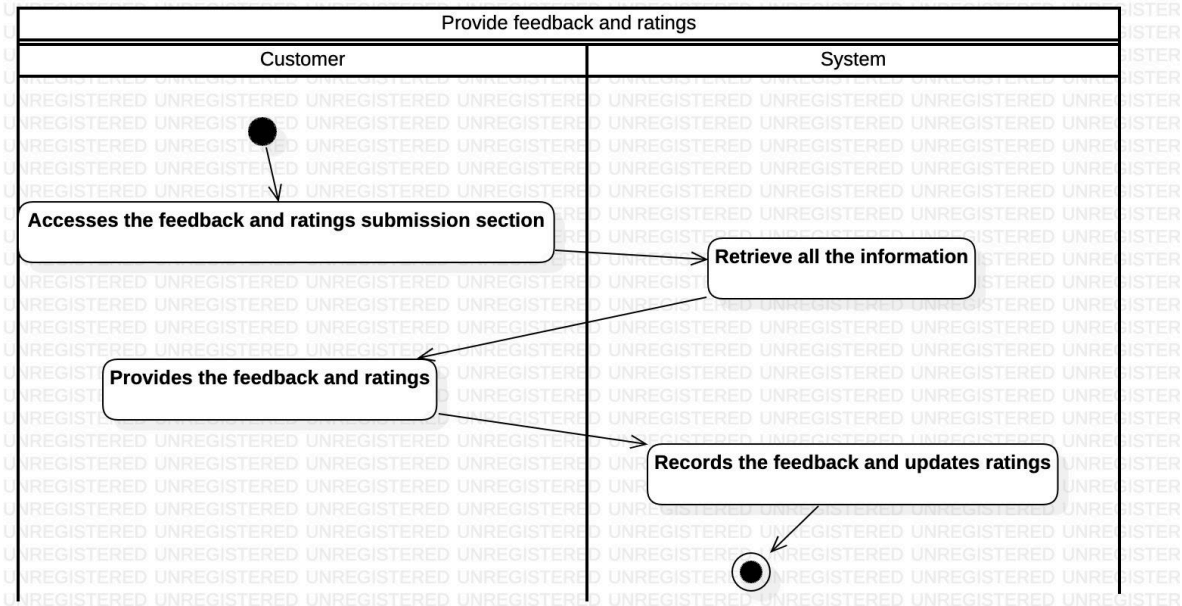


Send email

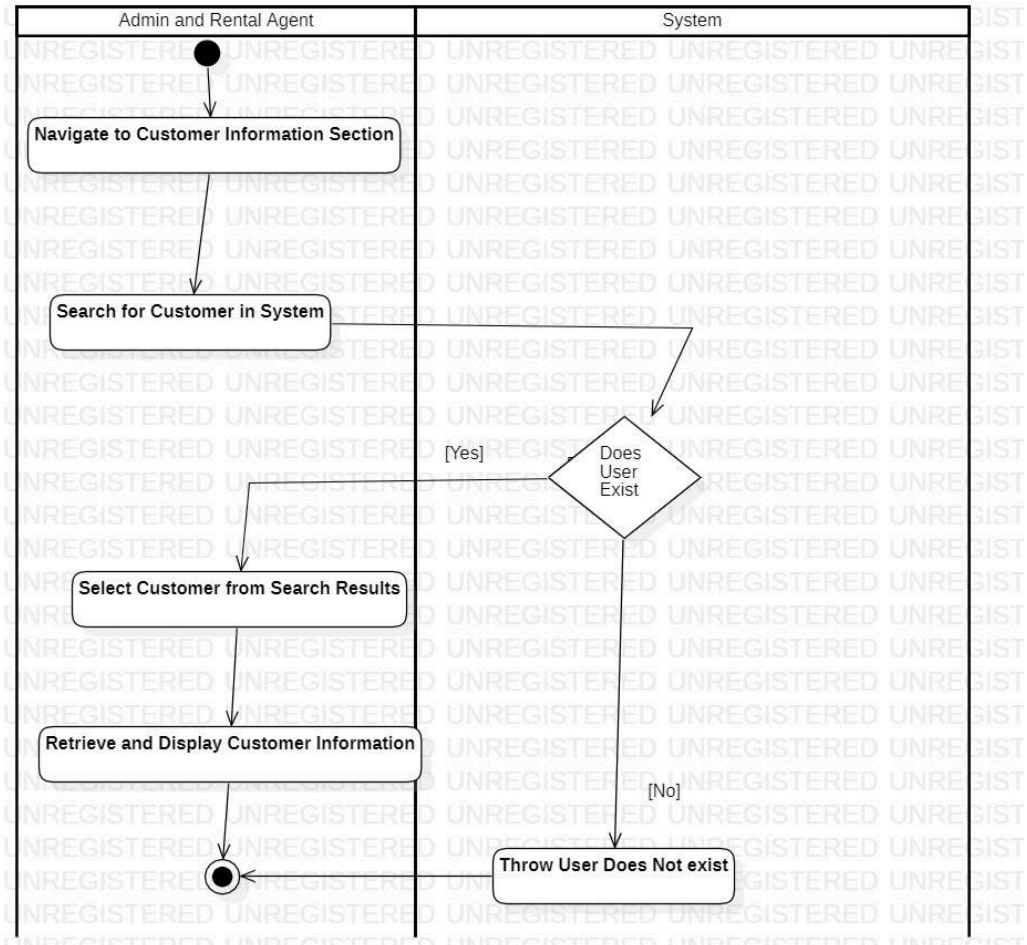



Process payment
Customer
System
Selects payment method
Enters payment details
Confirms the transaction
Processes the payment

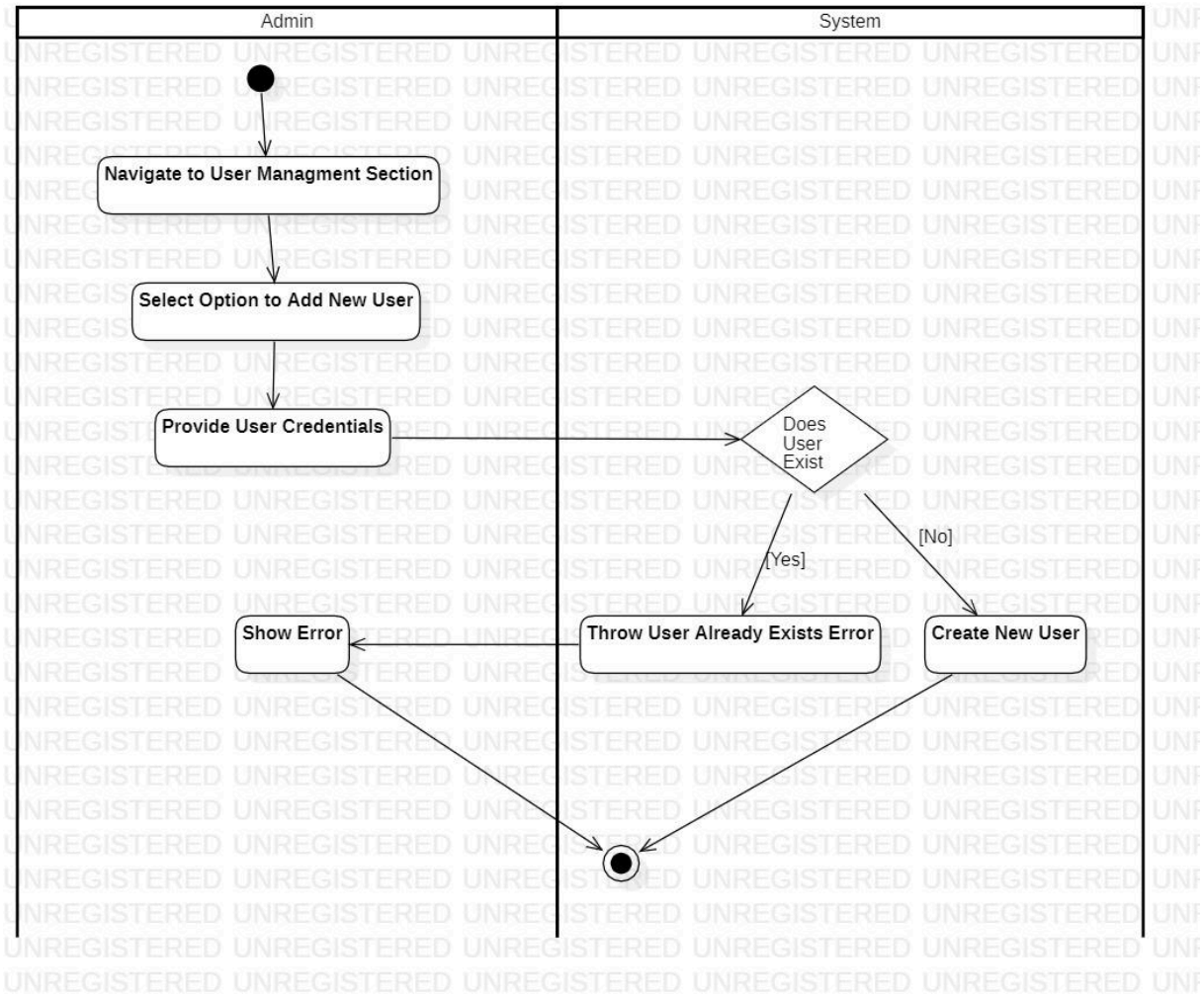

Pay via Credit card or cash
Customer
System
Navigates through payment section
Selects payment method
Processes the payment
Update payment status



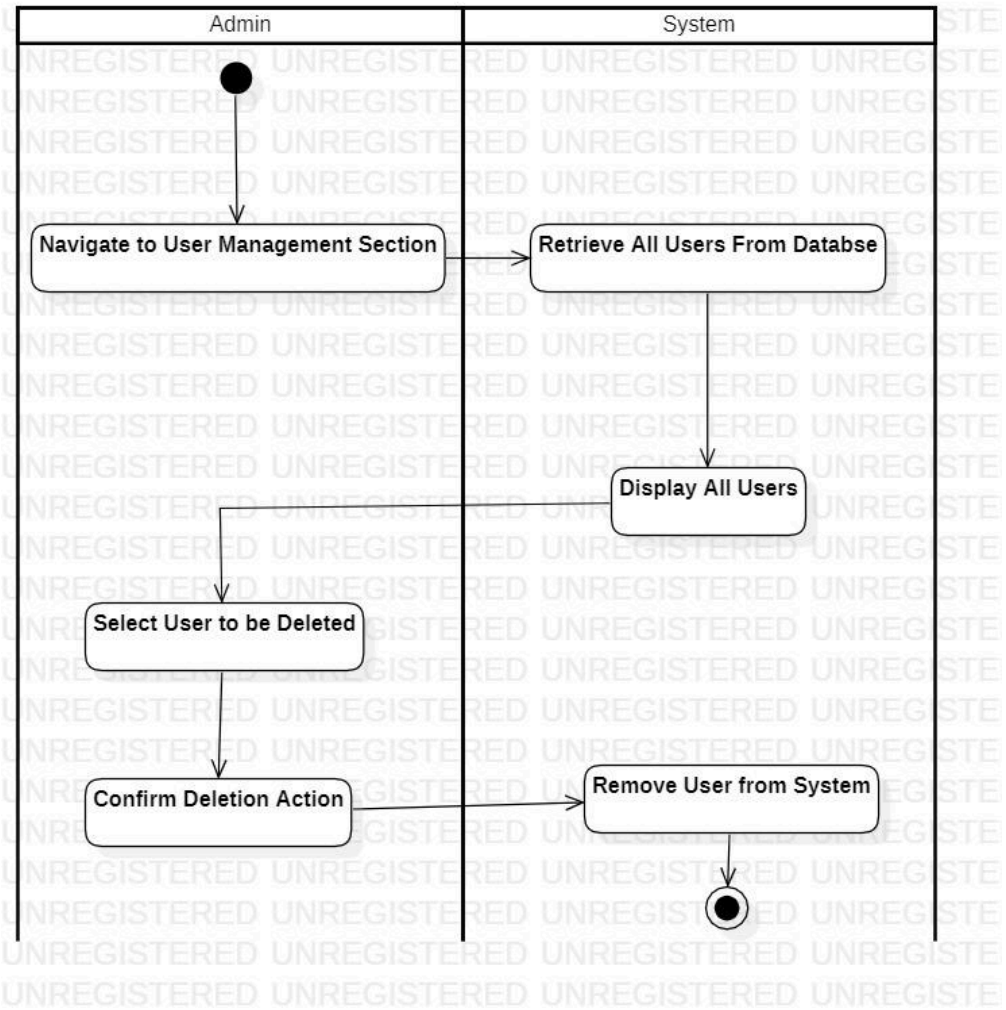
Check user info



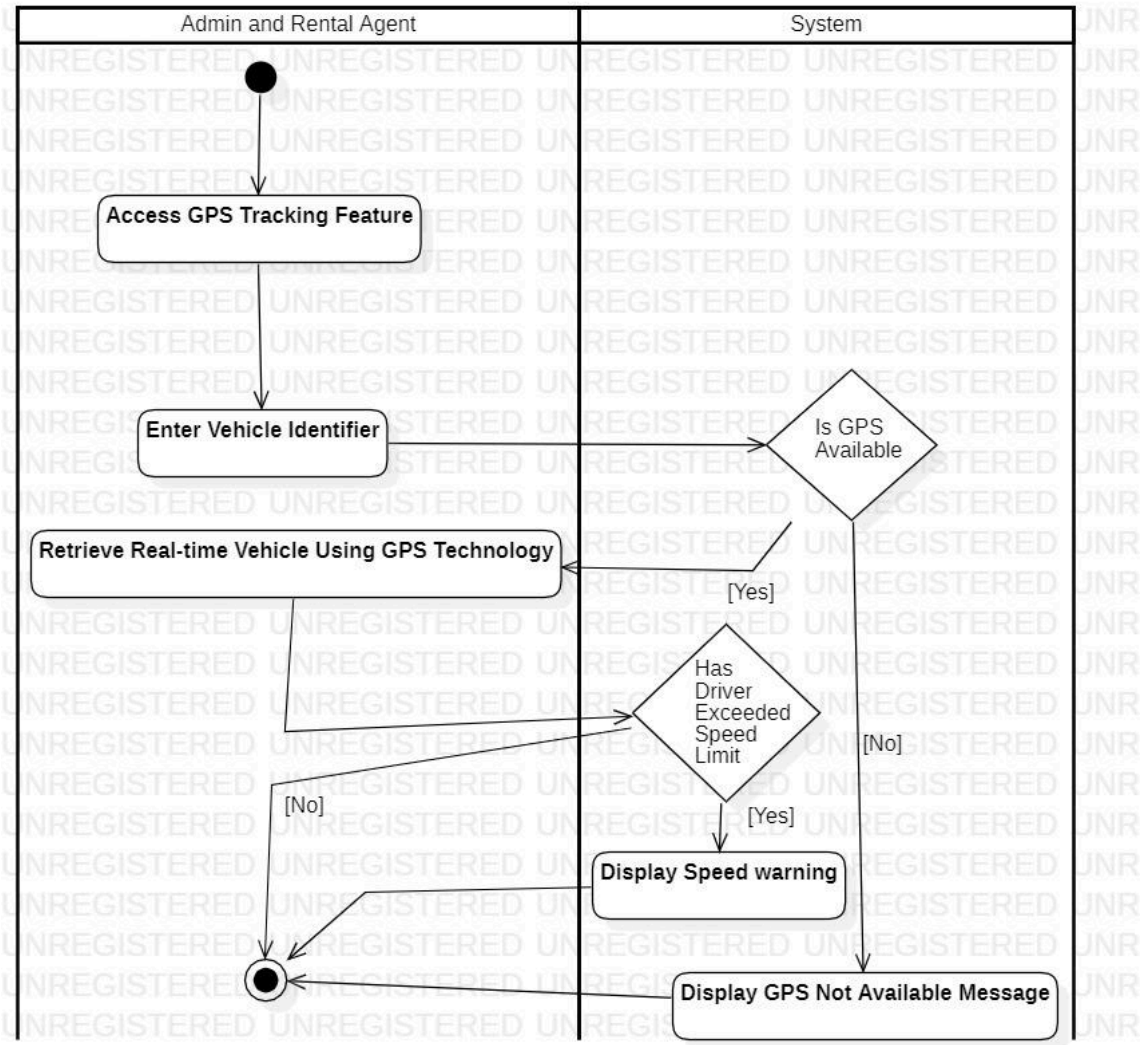
Create user


Admin
System
Navigate to User Managment Section
Select Option to Add New User
Provide User Credentials
Does User Exist
[Yes]
[No]
Show Error
Throw User Already Exists Error
Create New User

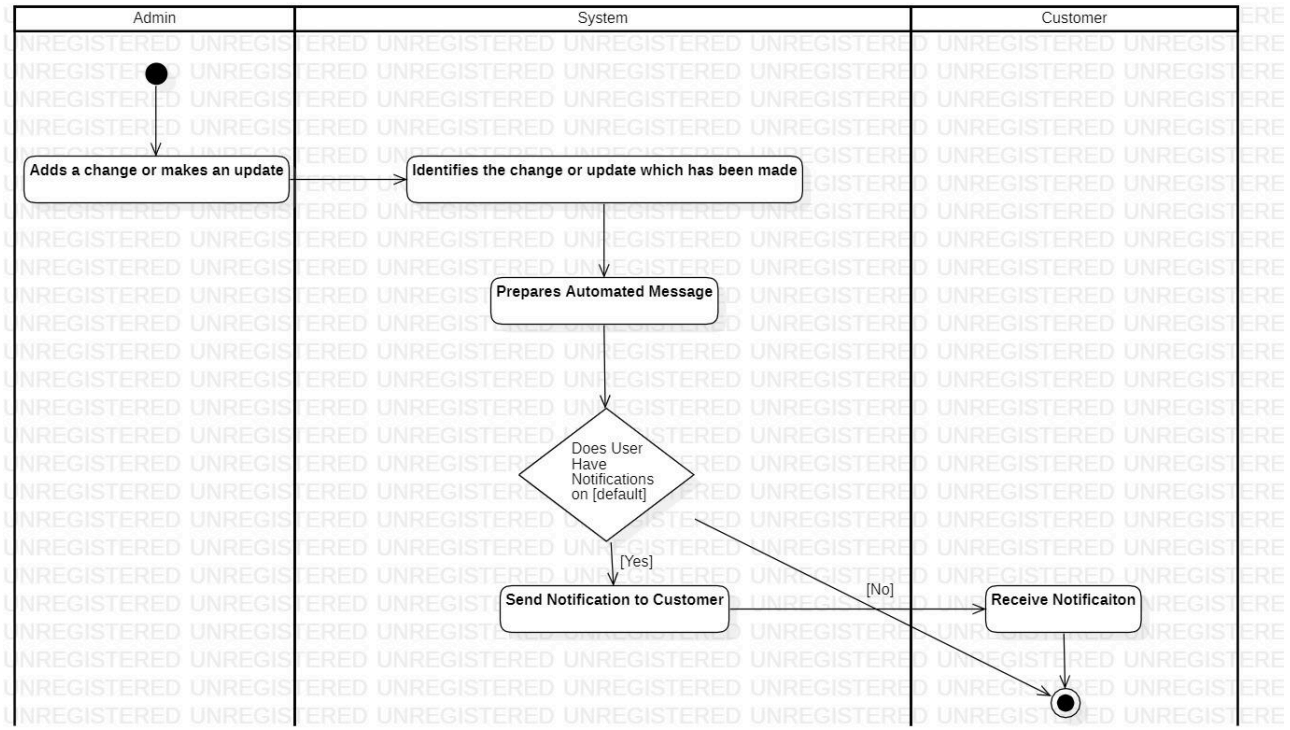
Delete User


Admin
System
Navigate to User Management Section
Retrieve All Users From Databse
Display All Users
Select User to be Deleted
Confirm Deletion Action
Remove User from System

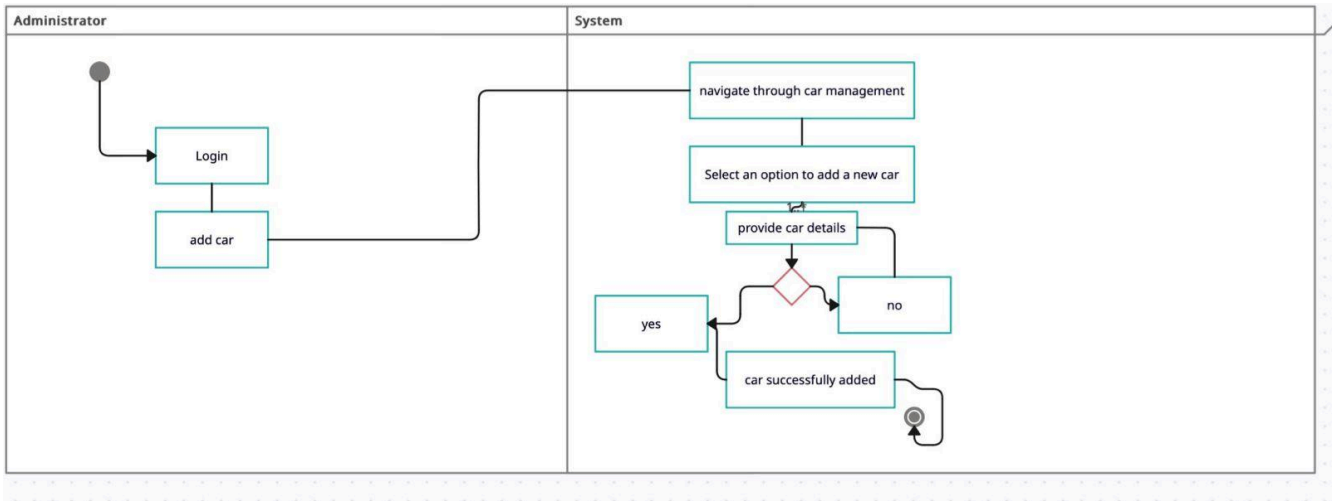
GPS


Admin and Rental Agent
System
Access GPS Tracking Feature
Enter Vehicle Identifier
Is GPS Available
Retrieve Real-time Vehicle Using GPS Technology
[Yes]
Has Driver Exceeded Speed Limit
[No]
[No]
[Yes]
Display Speed warning
Display GPS Not Available Message

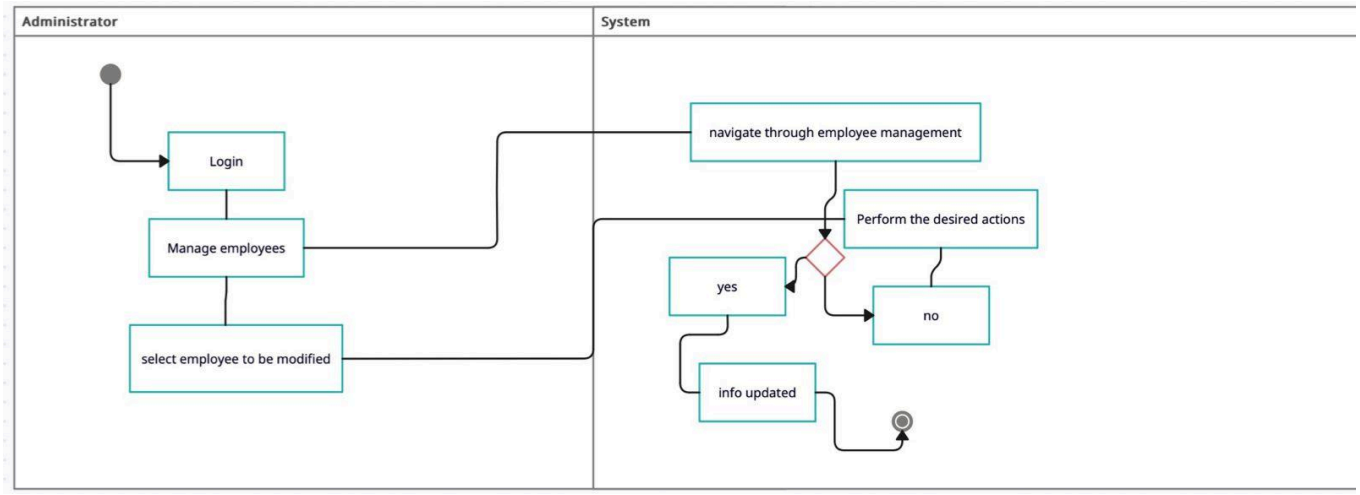
Receive notification



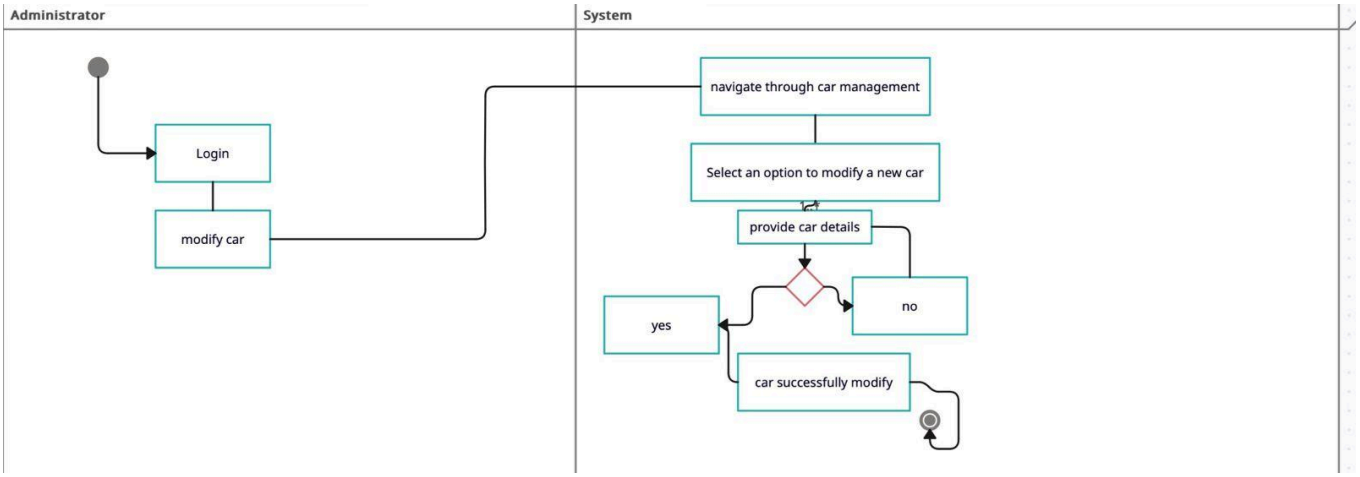
Add Car



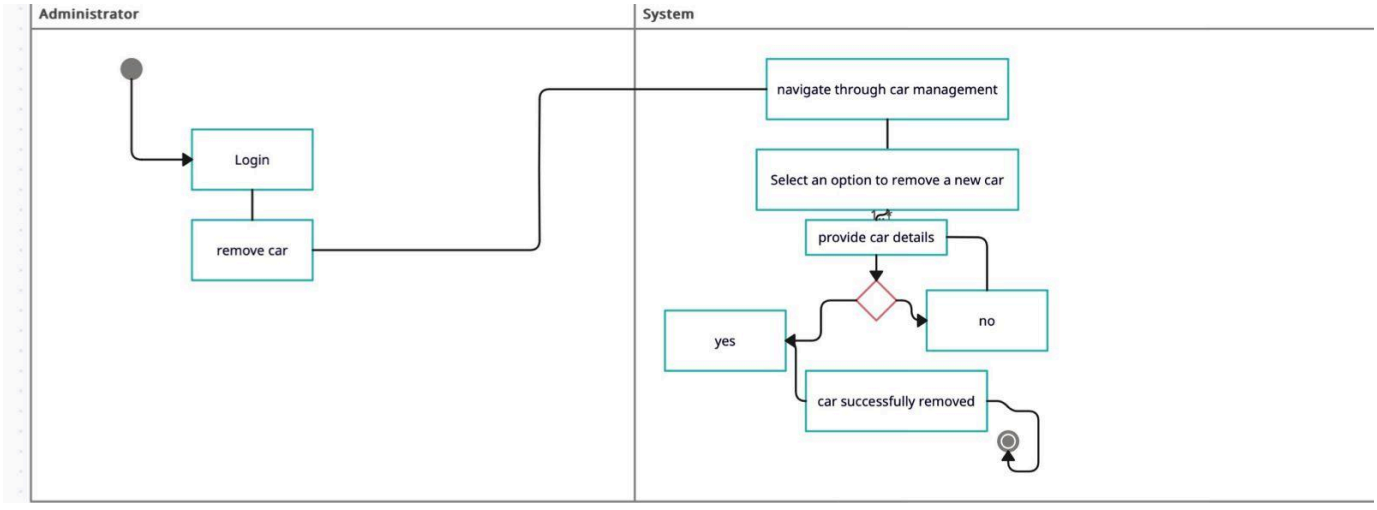
Manage Employees



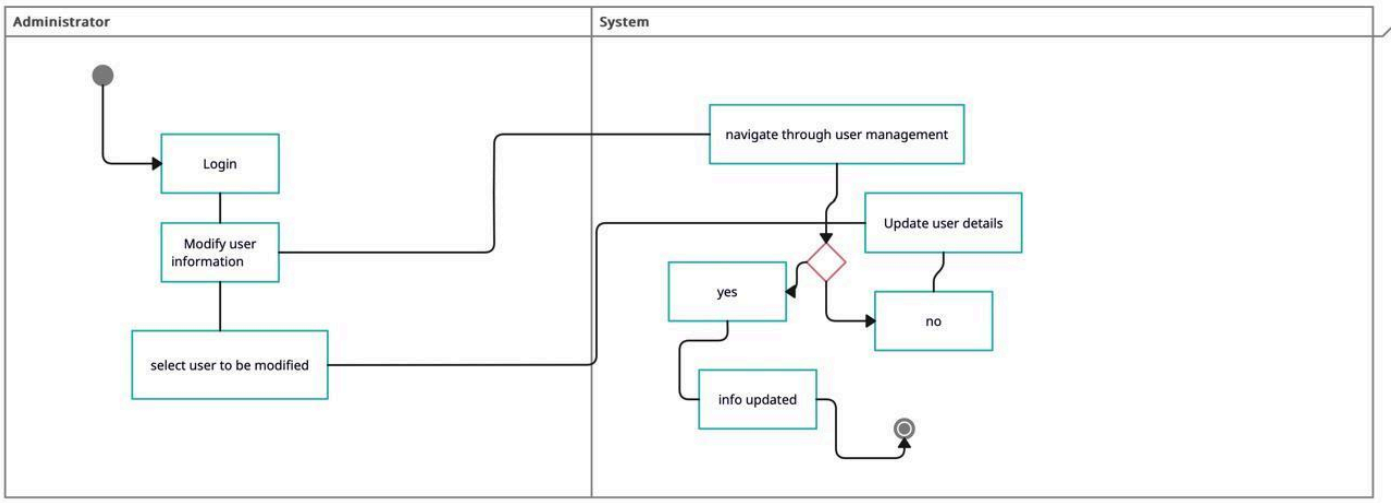
Modify Car

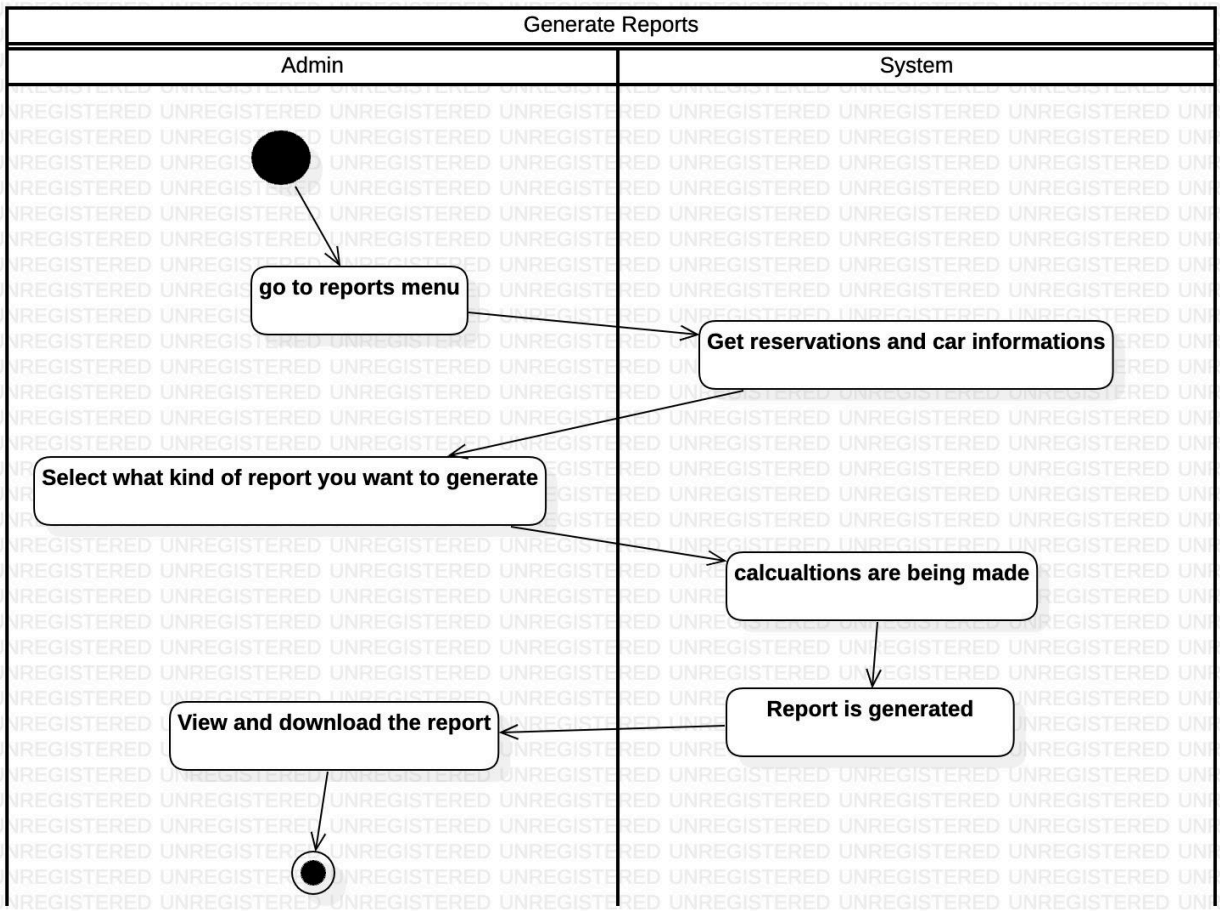


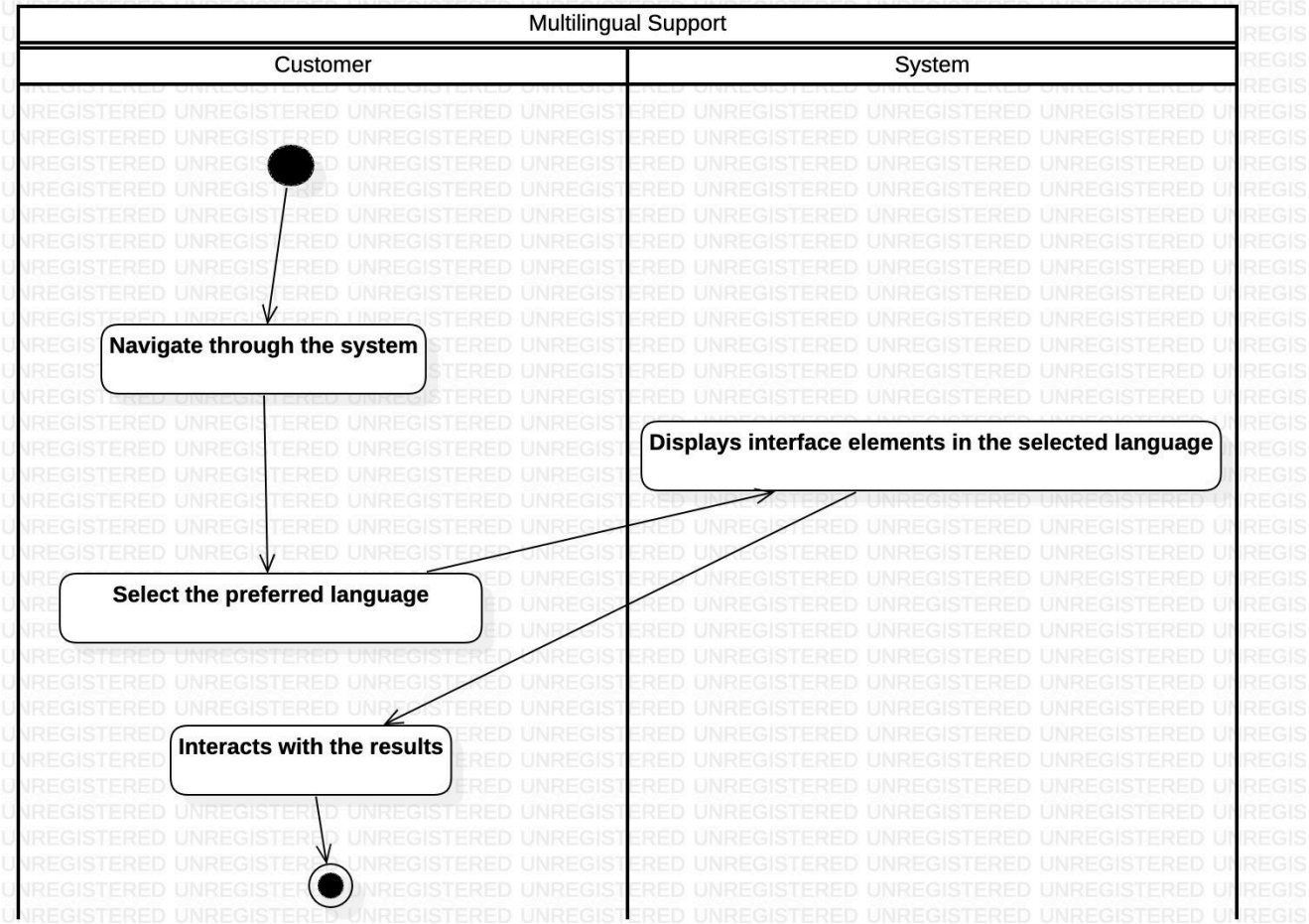
Remove Car

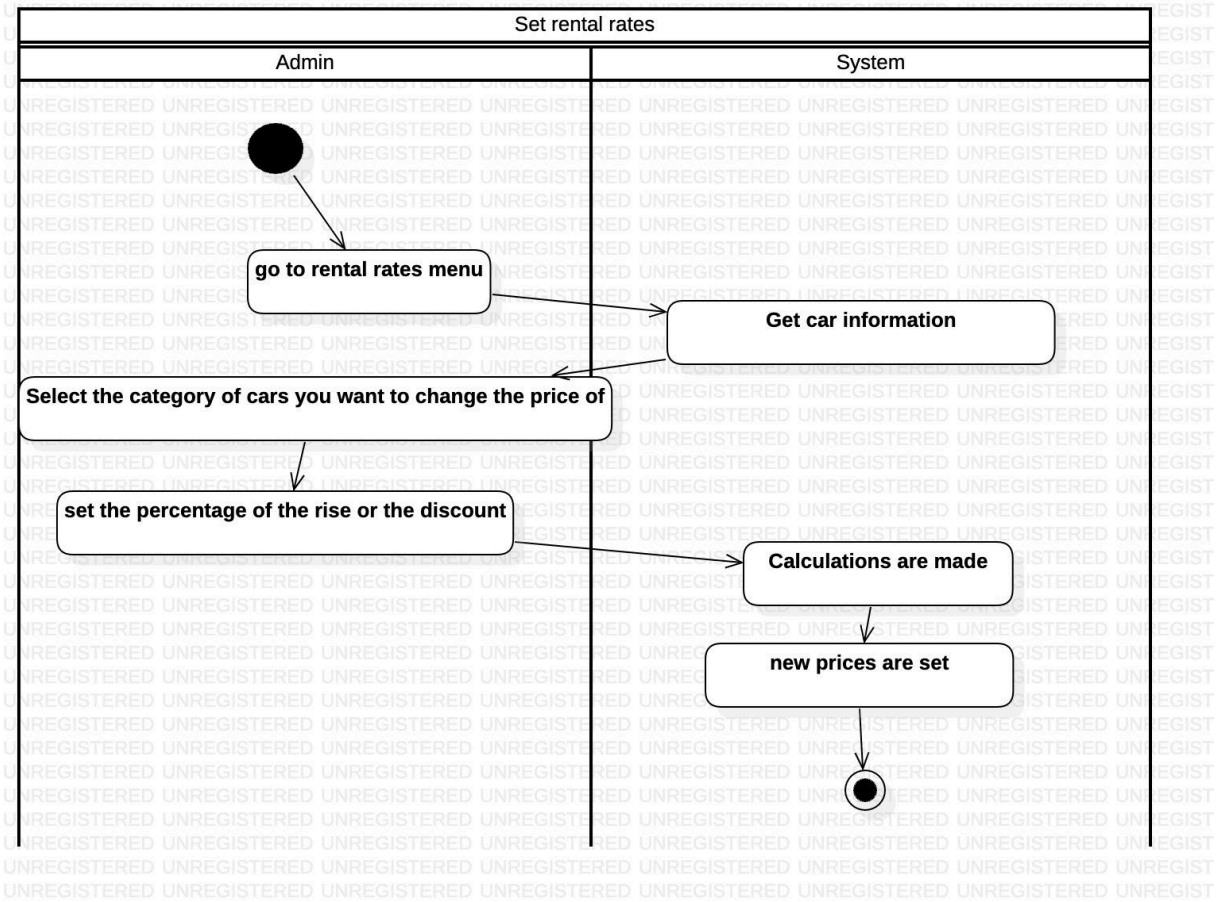


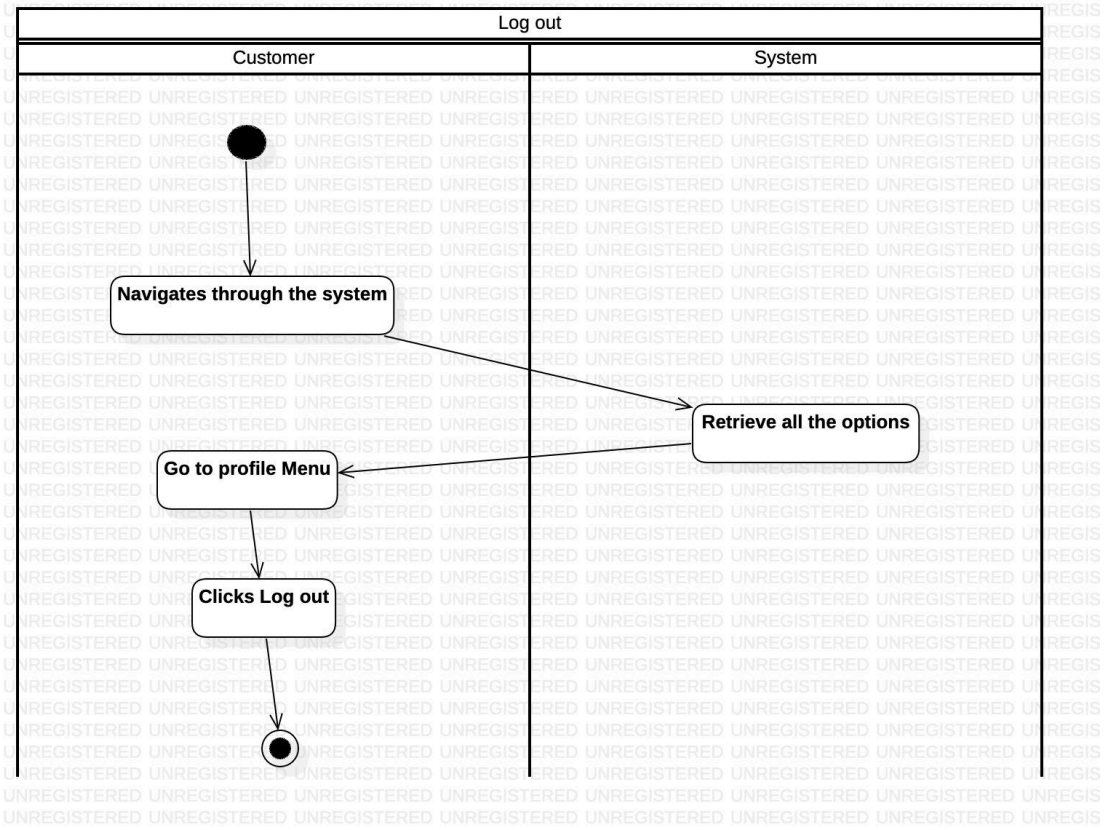
User Management





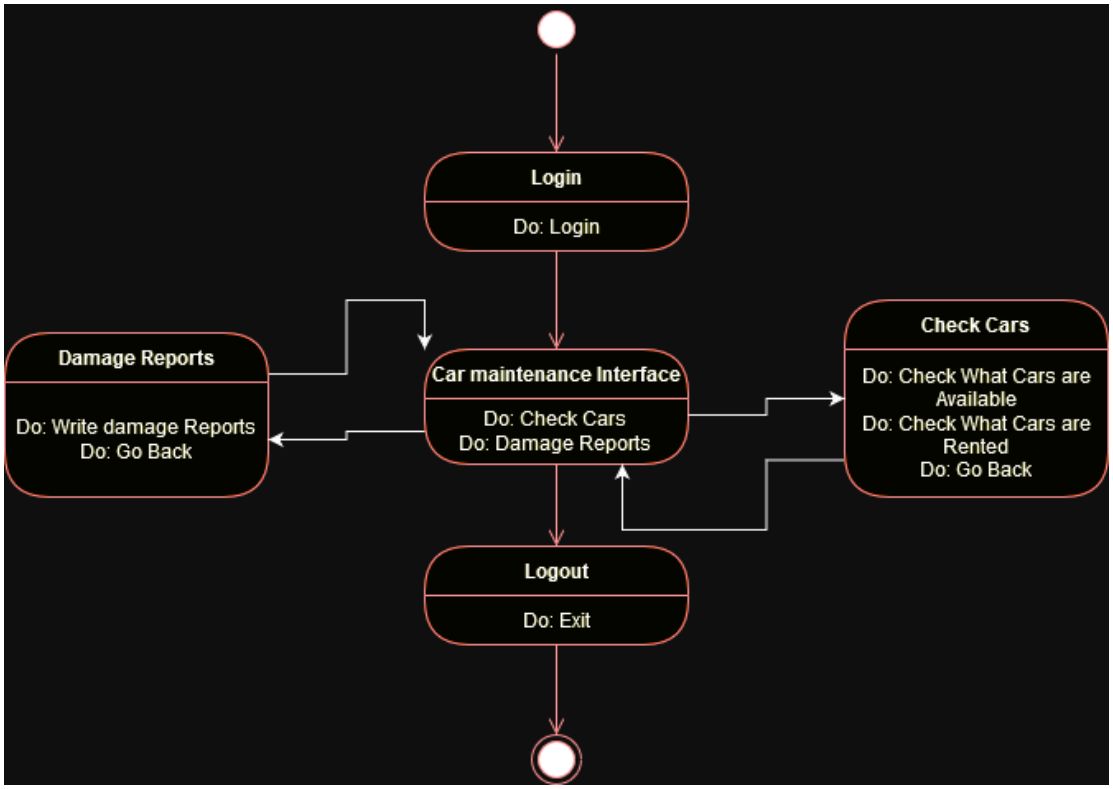

Multilingual Support
Customer
System
Navigate through the system
Displays interface elements in the selected language
Select the preferred language
Interacts with the results

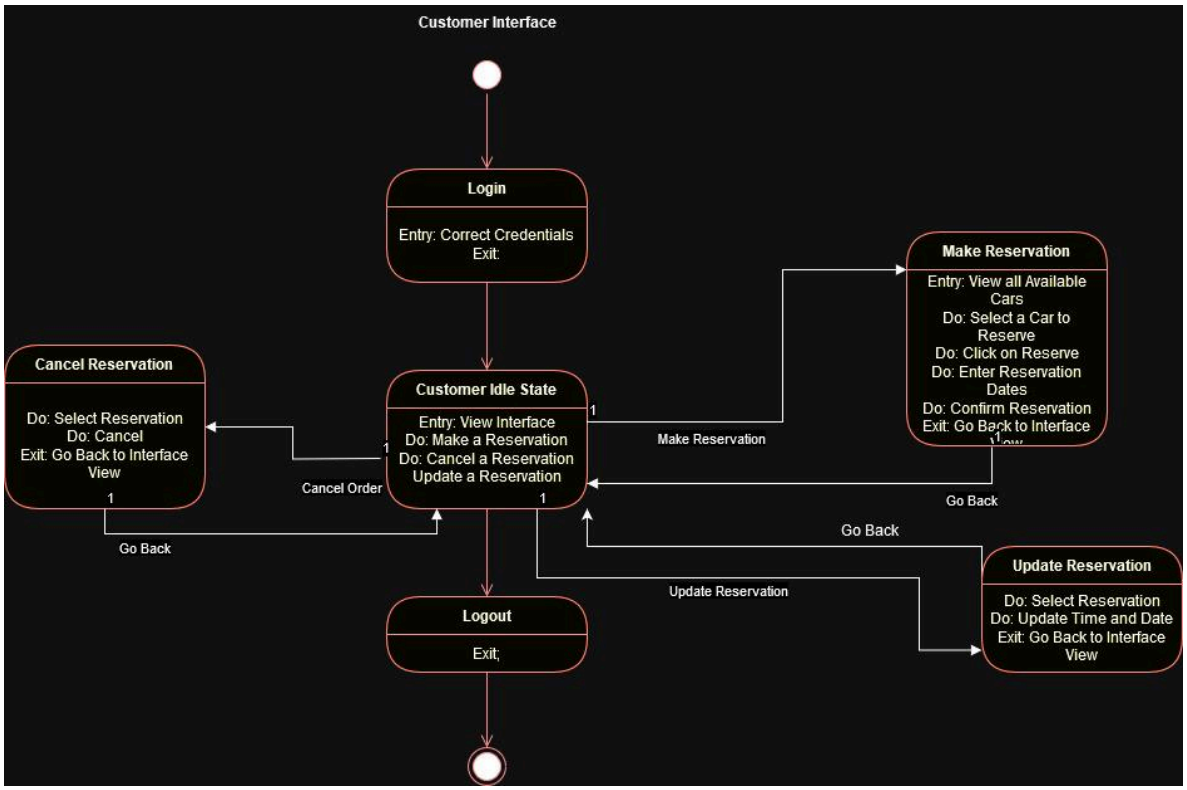

Set rental rates
Admin
System
go to rental rates menu
Get car information
Select the category of cars you want to change the price of
set the percentage of the rise or the discount
Calculations are made
new prices are set


Log out
Customer
System
Navigates through the system
Retrieve all the options
Go to profile Menu
Clicks Log out

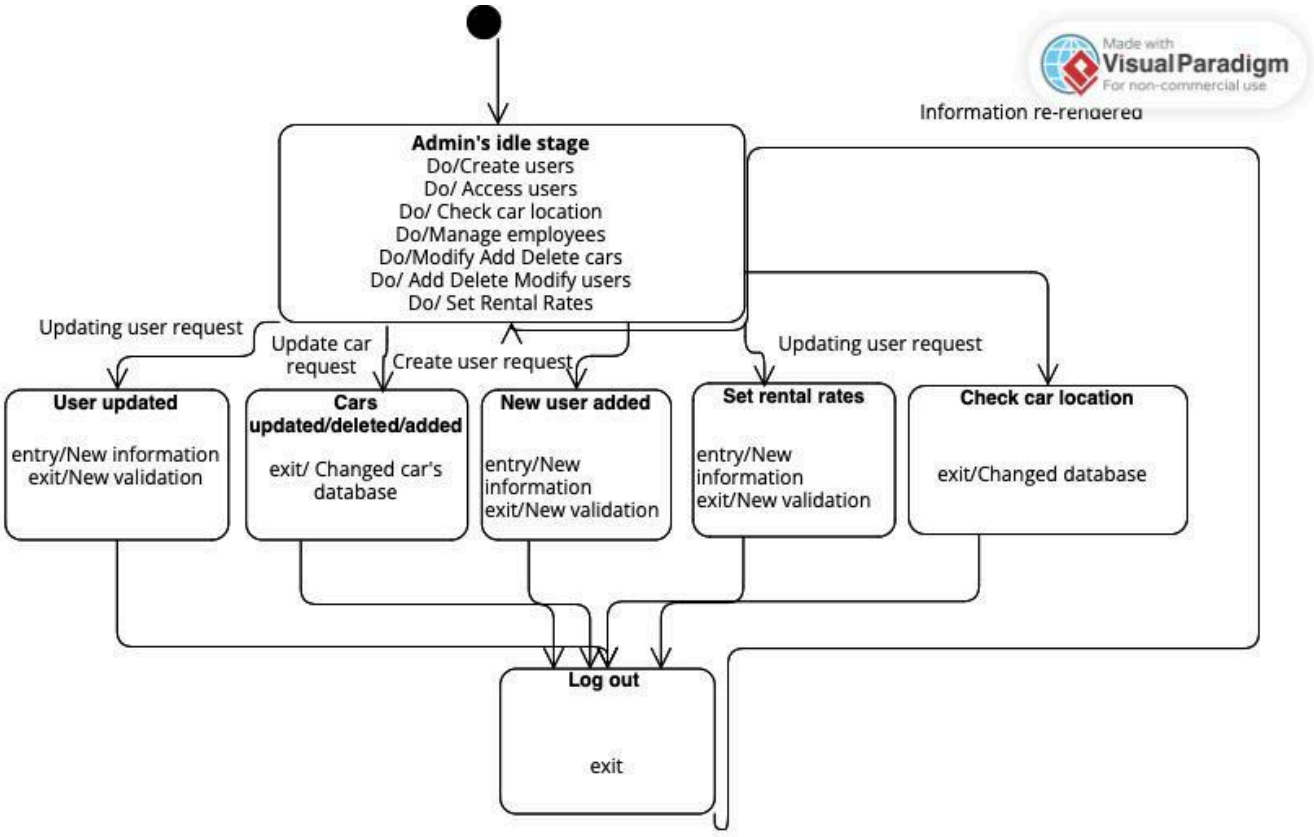
4.7 State Diagrams

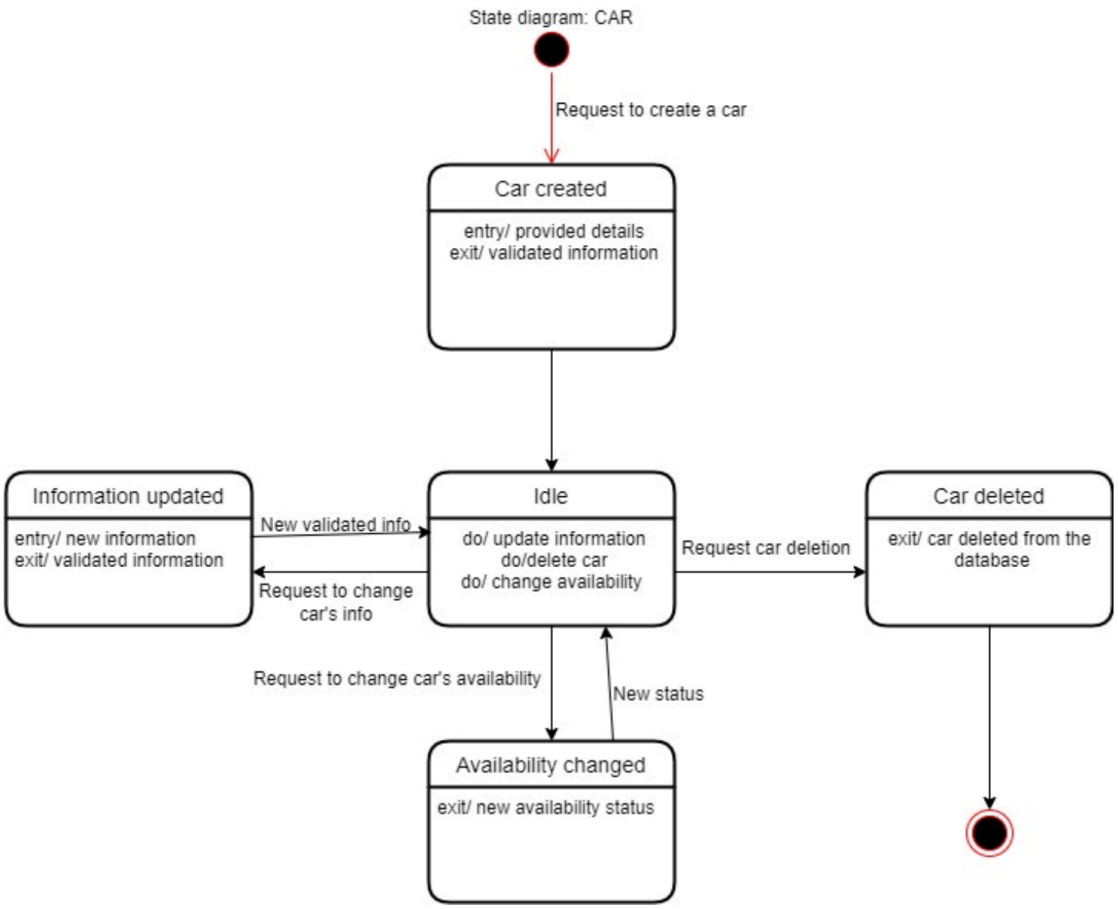
Car maintenance operator



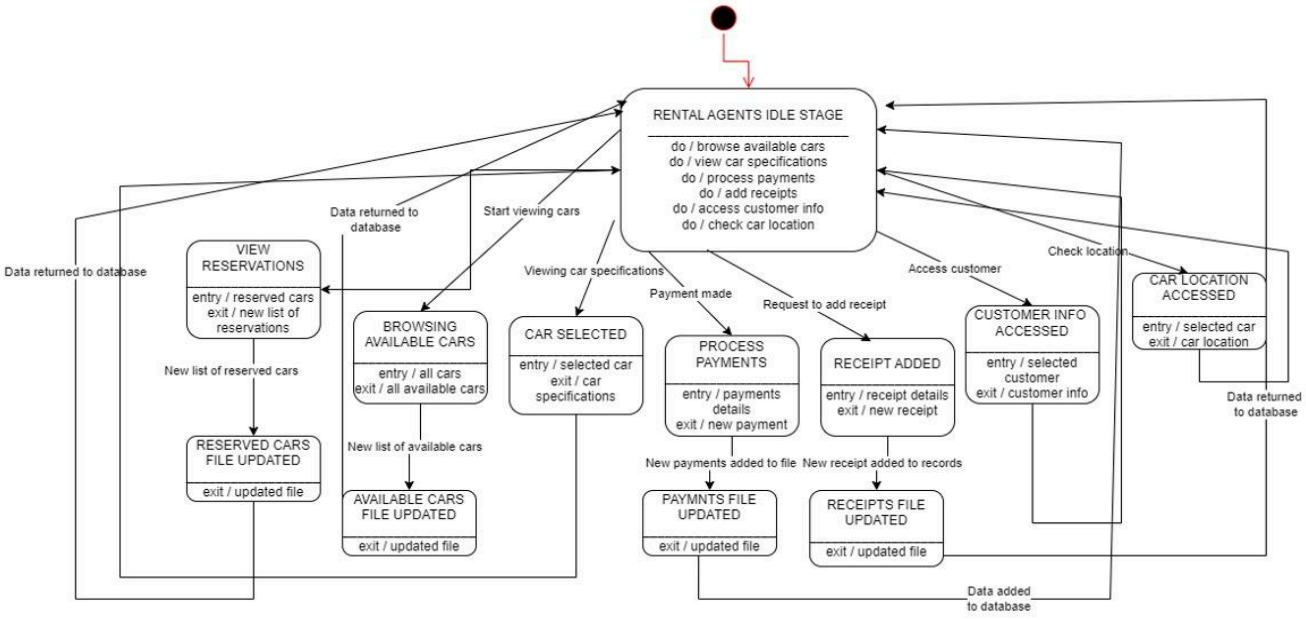


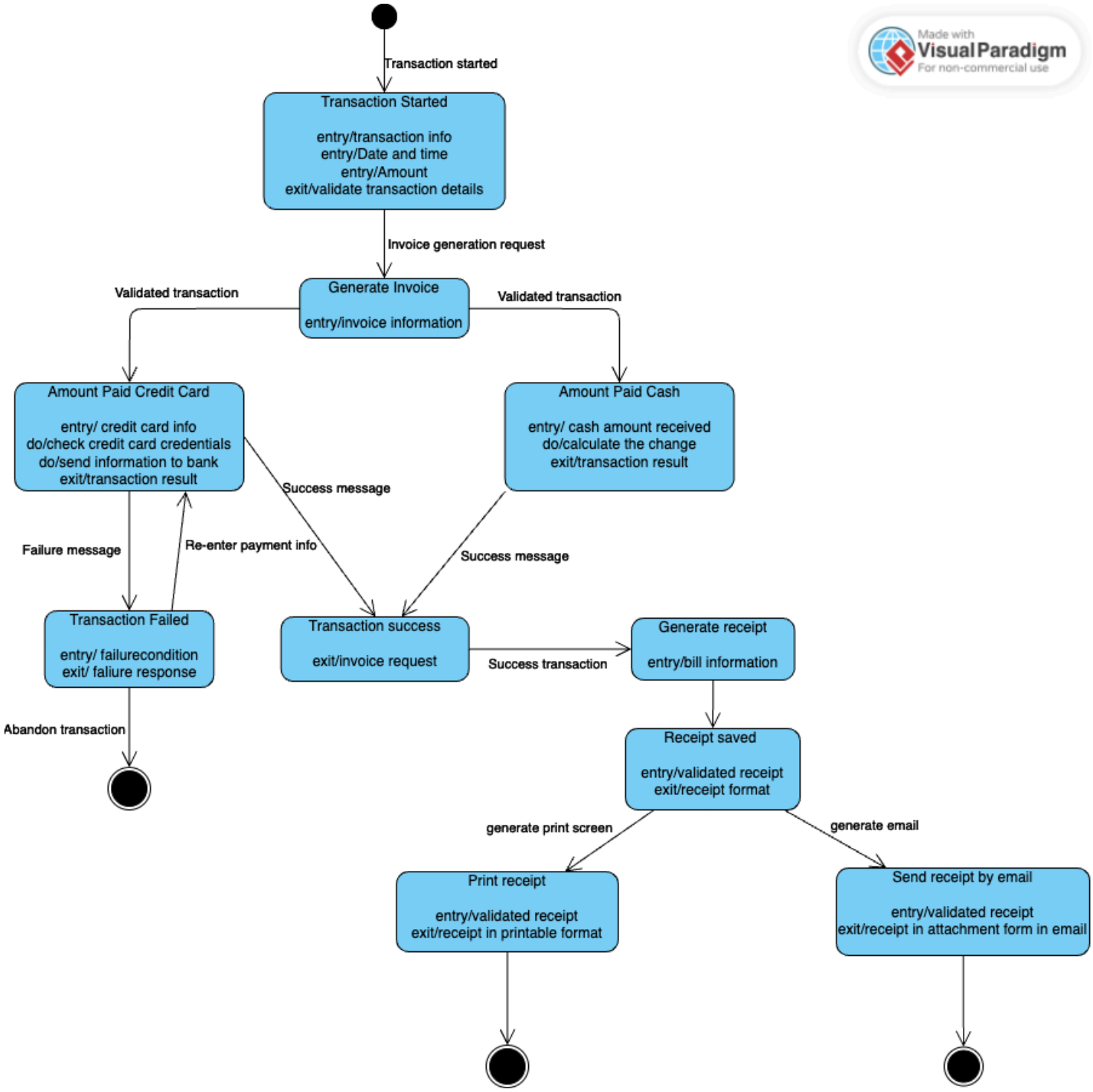
Admin

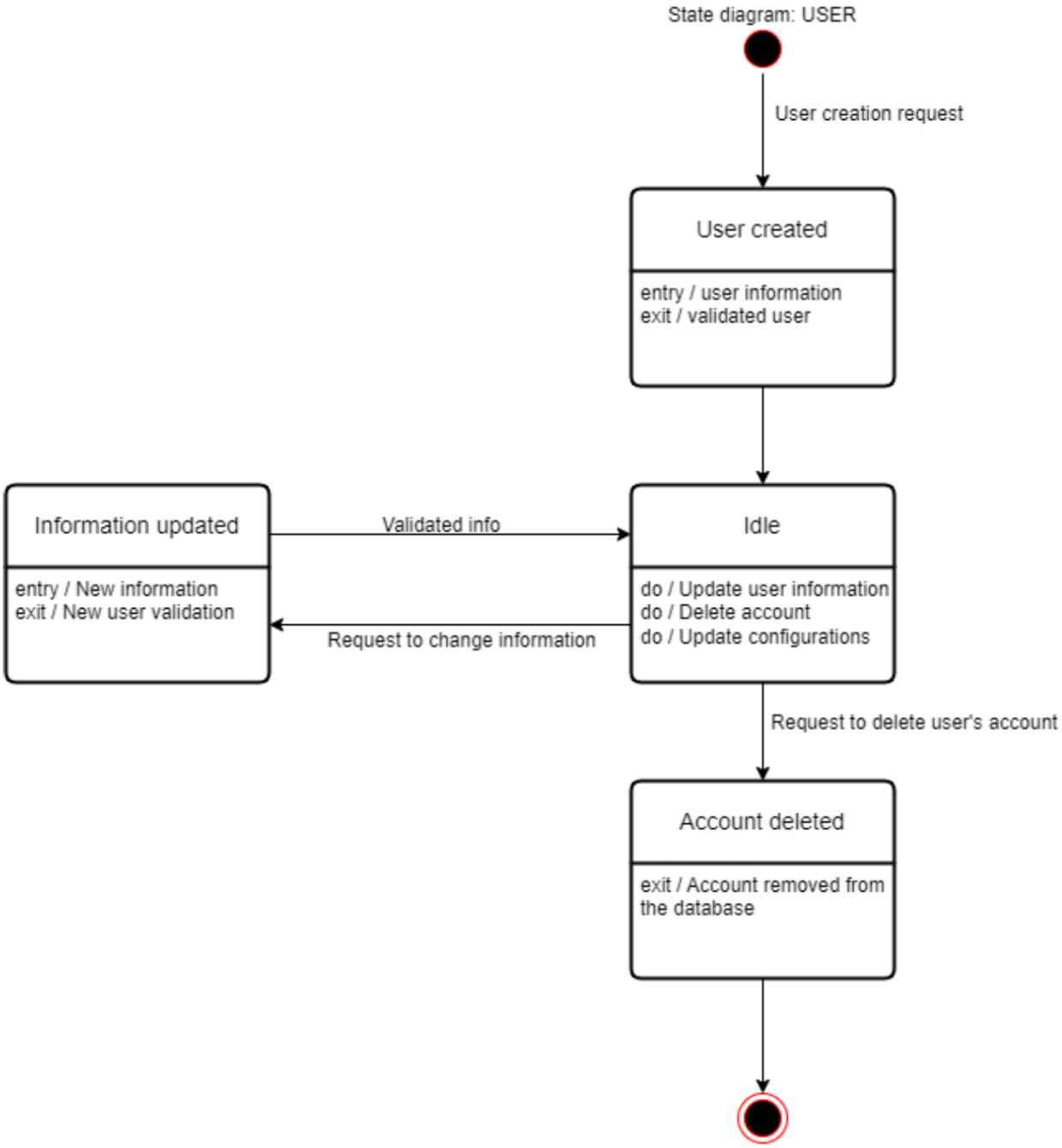

Made with Visual Paradigm
For non-commercial use
Information re-rendered
Admin's idle stage
Do/Create users
Do/ Access users
Do/ Check car location
Do/Manage employees
Do/Modify Add Delete cars
Do/ Add Delete Modify users
Do/ Set Rental Rates
Updating user request
Update car request
Create user request
Updating user request
User updated
entry/New information
exit/New validation
Cars updated/deleted/added
exit/ Changed car's database
New user added
entry/New information
exit/New validation
Set rental rates
entry/New information
exit/New validation
Check car location
exit/Changed database
Log out
exit



Rental Agent

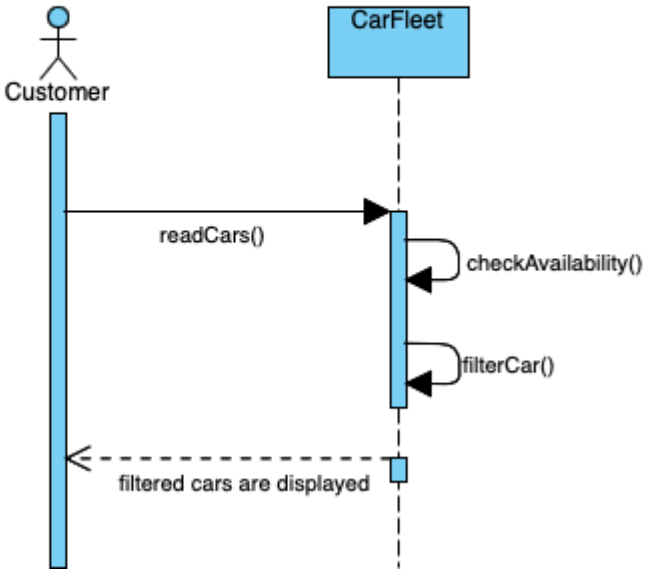



Transaction started
Transaction Started
entry/transaction info
entry/Date and time
entry/Amount
exit/validate transaction details
Invoice generation request
Generate Invoice
entry/invoice information
Validated transaction
Validated transaction
Amount Paid Credit Card
entry/ credit card info
do/check credit card credentials
do/send information to bank
exit/transaction result
Amount Paid Cash
entry/ cash amount received
do/calculate the change
exit/transaction result
Success message
Failure message
Re-enter payment info
Success message
Transaction Failed
entry/ failurecondition
exit/ faliure response
Transaction success
exit/invoice request
Success transaction
Generate receipt
entry/bill information
Abandon transaction
Receipt saved
entry/validated receipt
exit/receipt format
generate print screen
generate email
Print receipt
entry/validated receipt
exit/receipt in printable format
Send receipt by email
entry/validated receipt
exit/receipt in attachment form in email

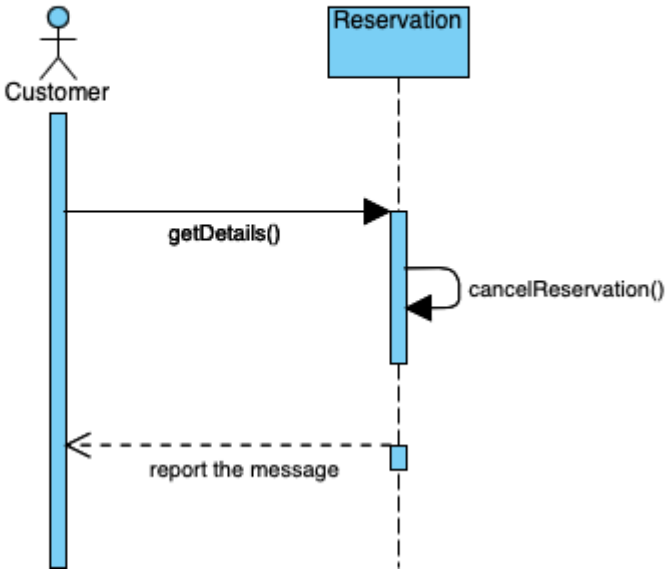

State diagram: USER
User creation request
User created
entry / user information
exit / validated user
Information updated
entry / New information
exit / New user validation
Validated info
Request to change information
Idle
do / Update user information
do / Delete account
do / Update configurations
Request to delete user's account
Account deleted
exit / Account removed from the database

4.8 Sequence Diagrams

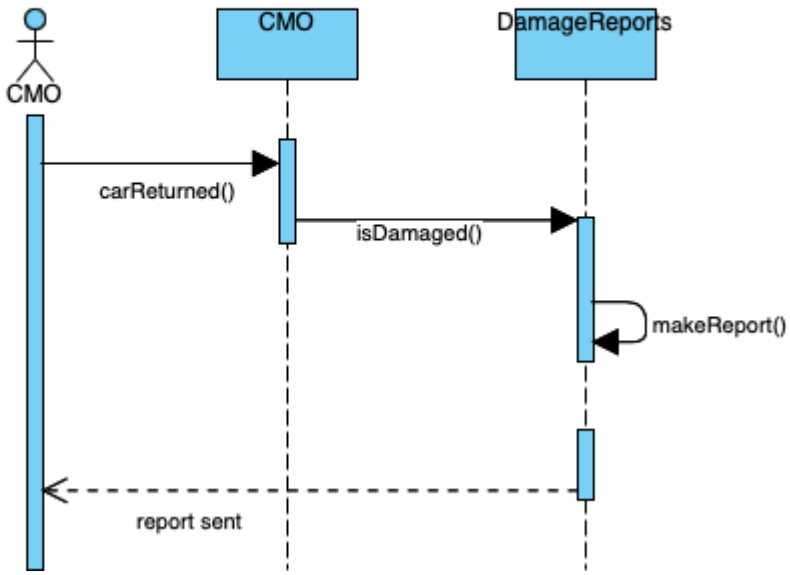
Browse and filter available ca



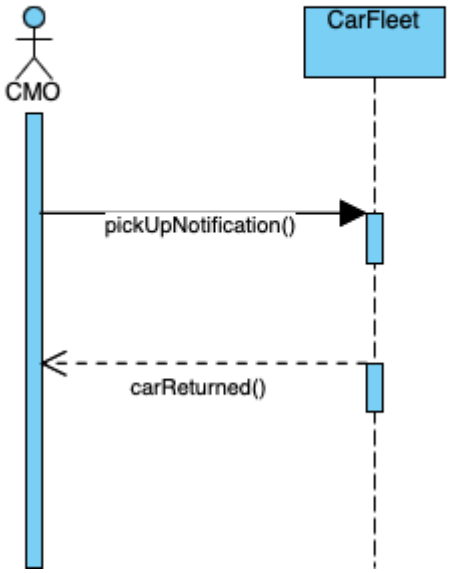
Cancel reservation



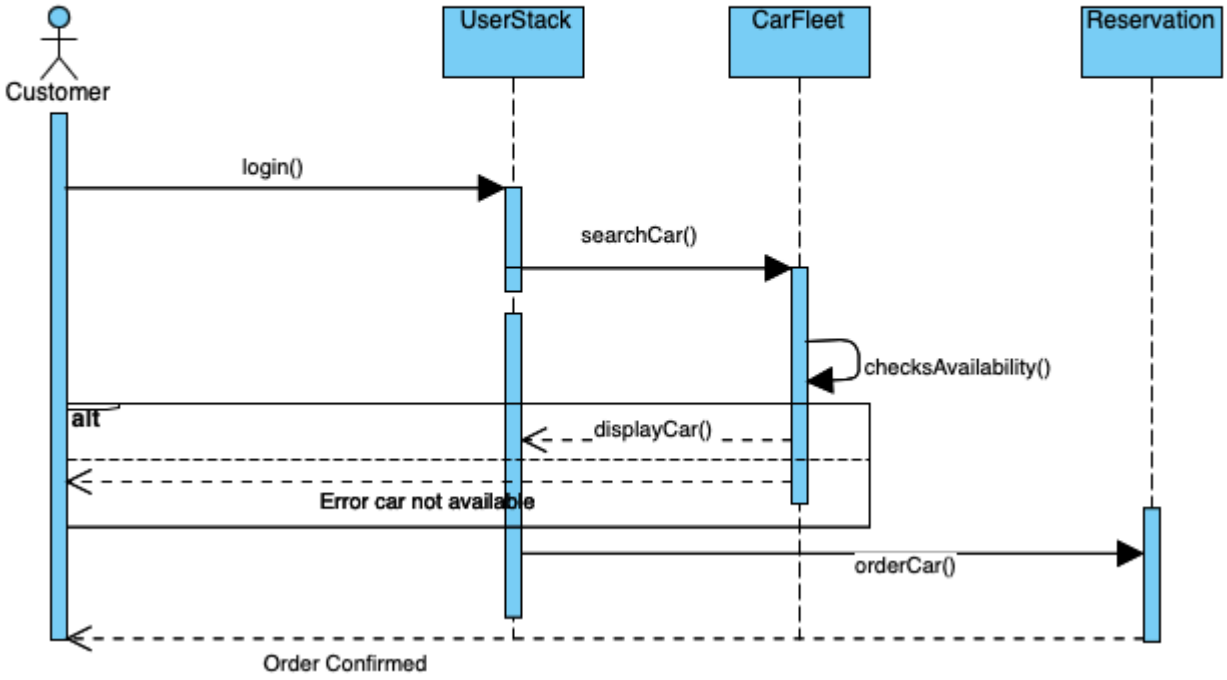
Report damages or fines



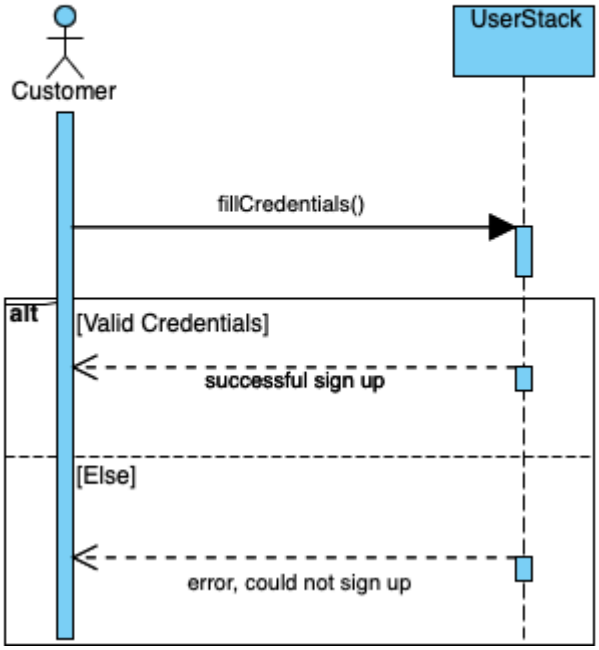
Report when a car is returned



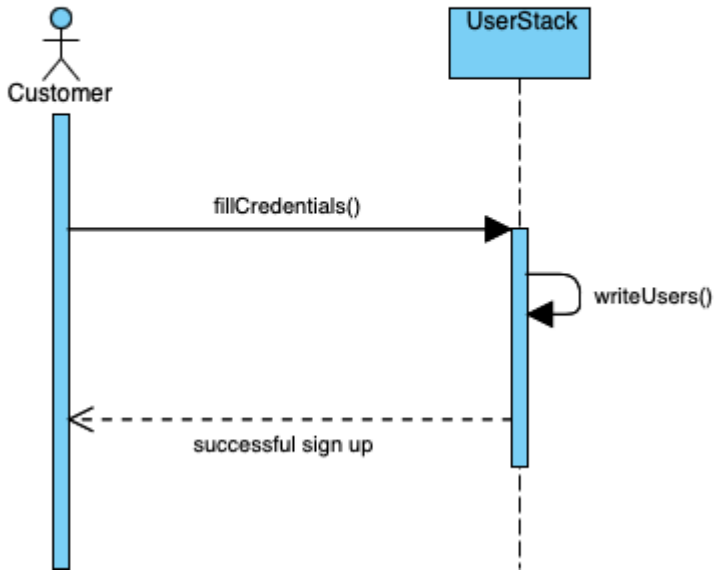
Select car and make reservation



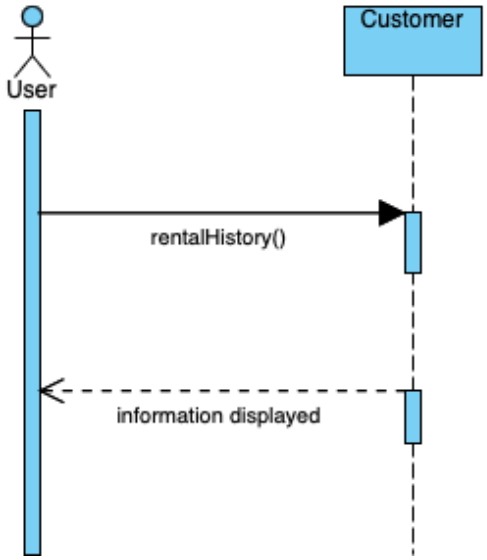
Sign in



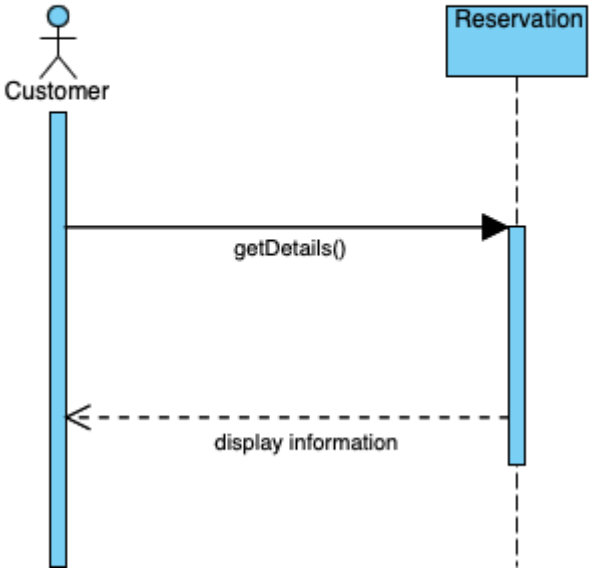
Sign up



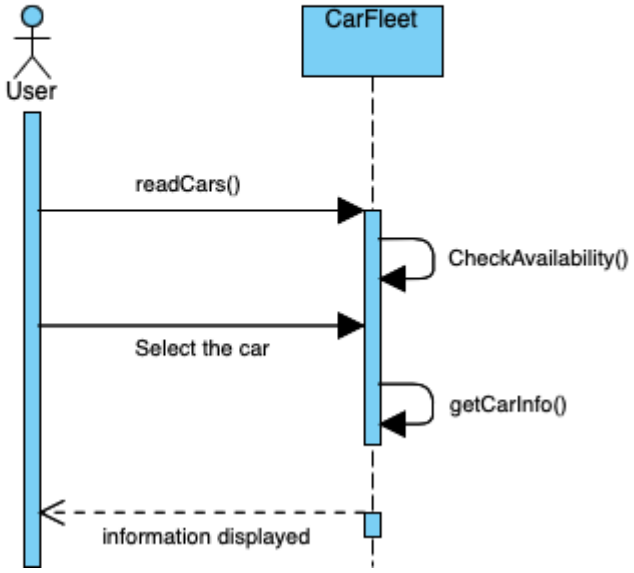
View rental history

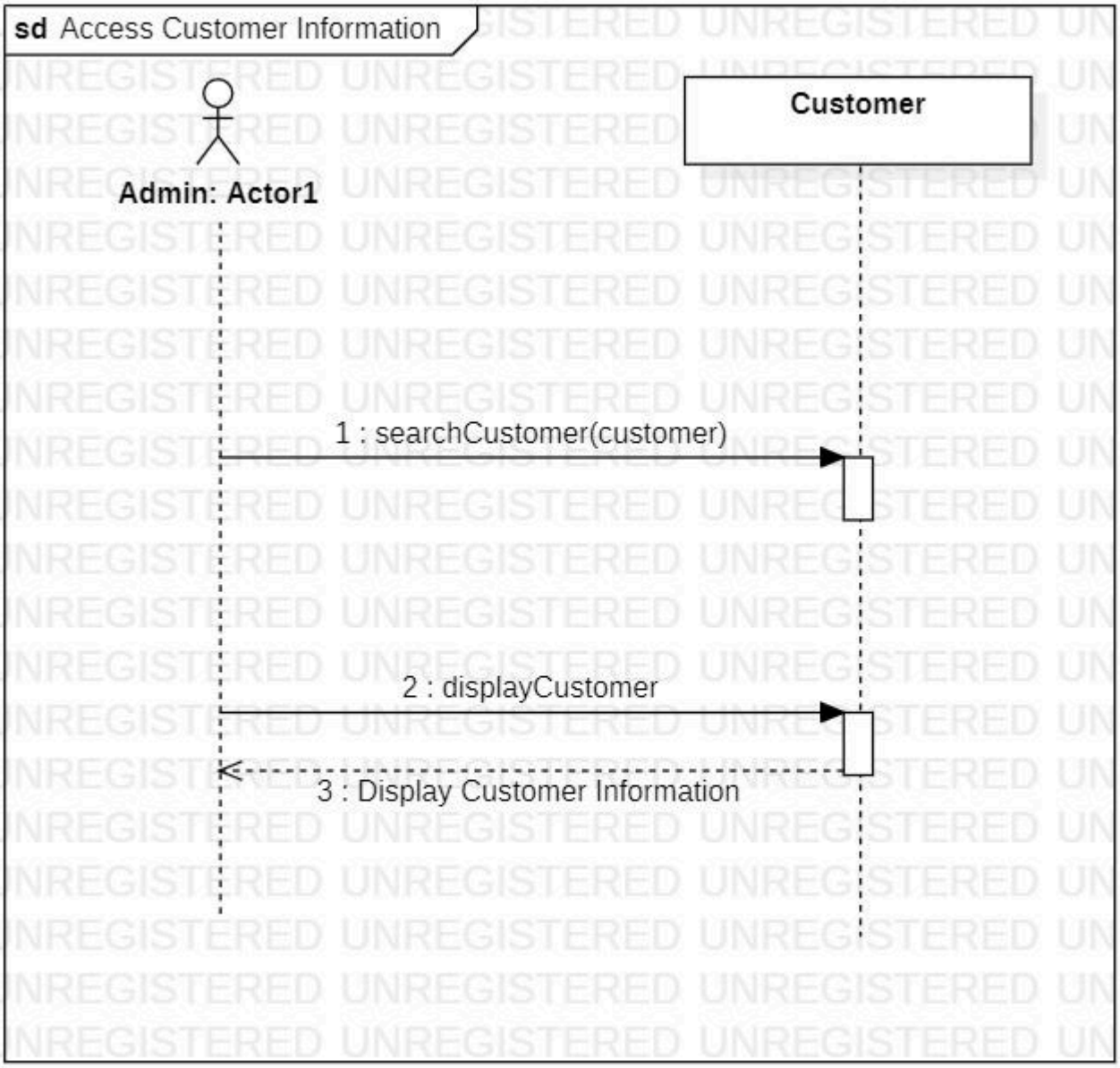


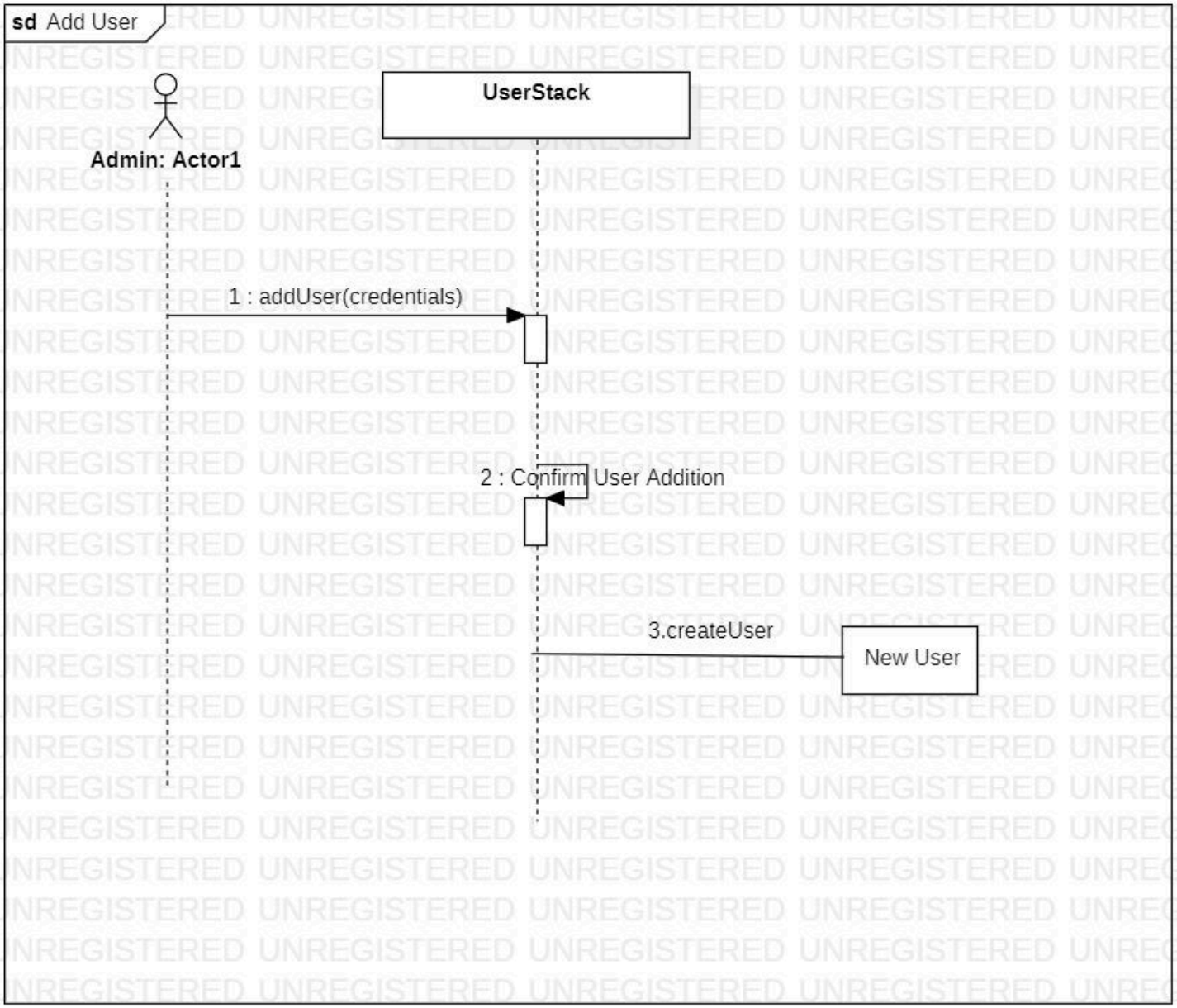
View reservation

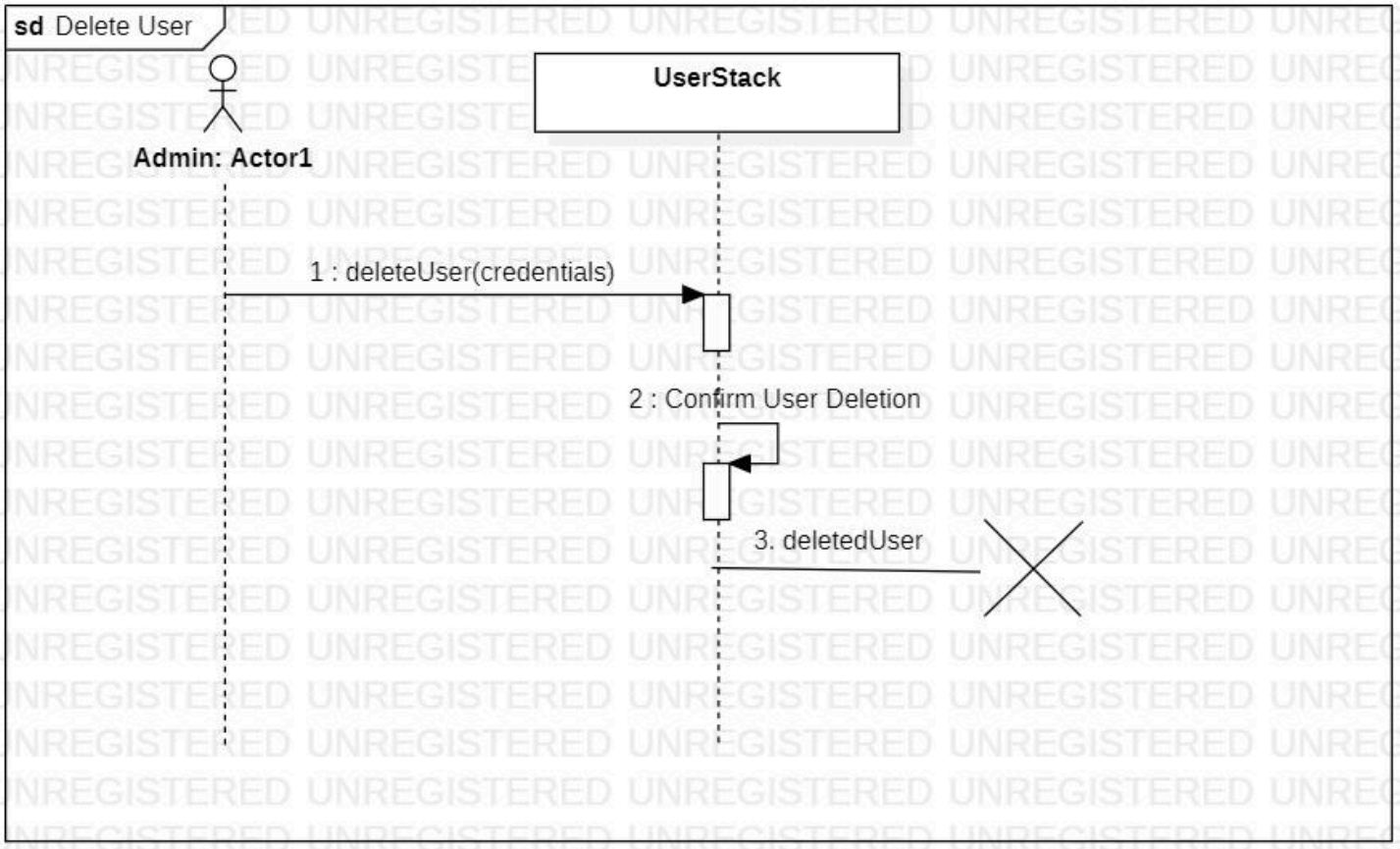


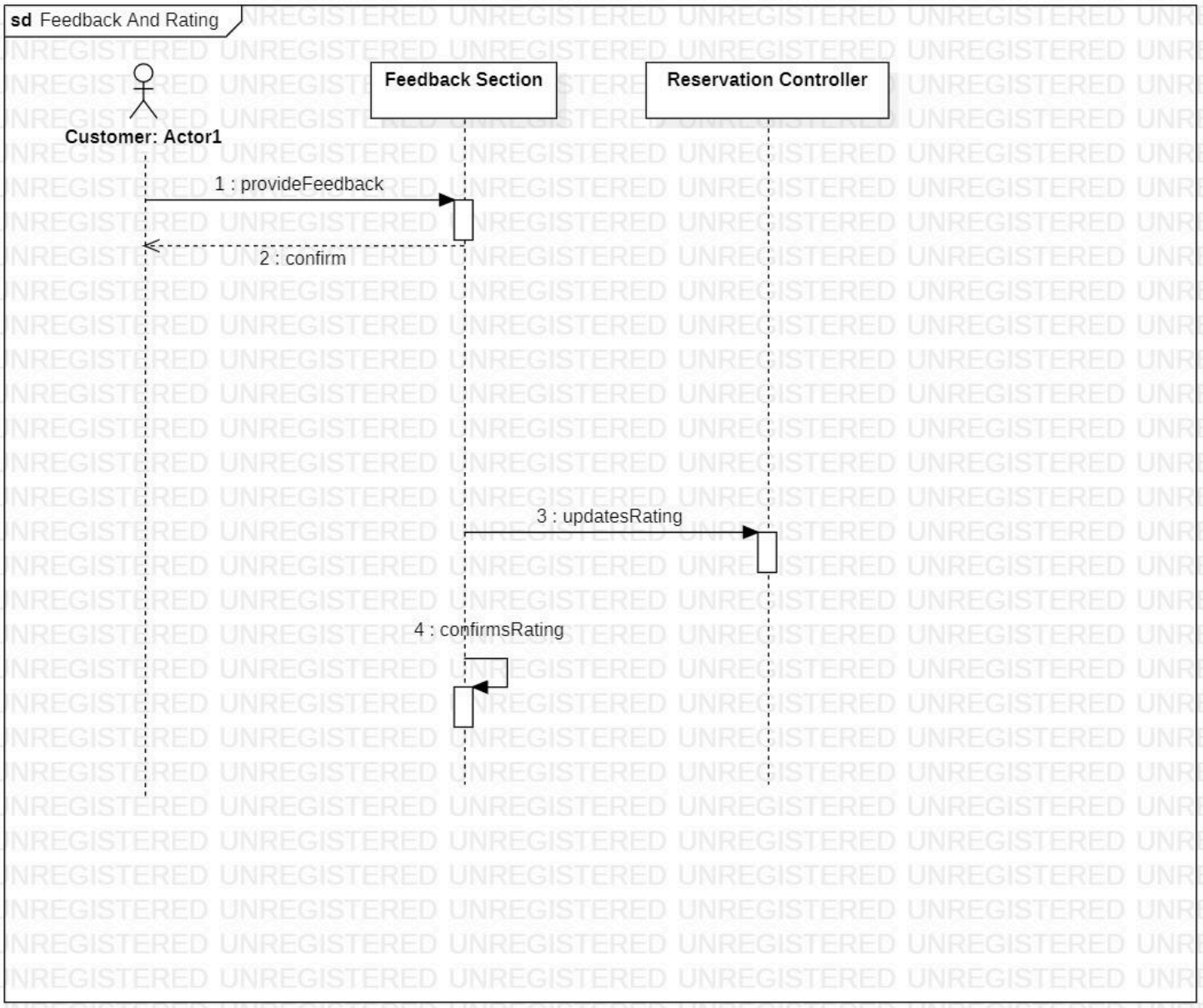
View specifications

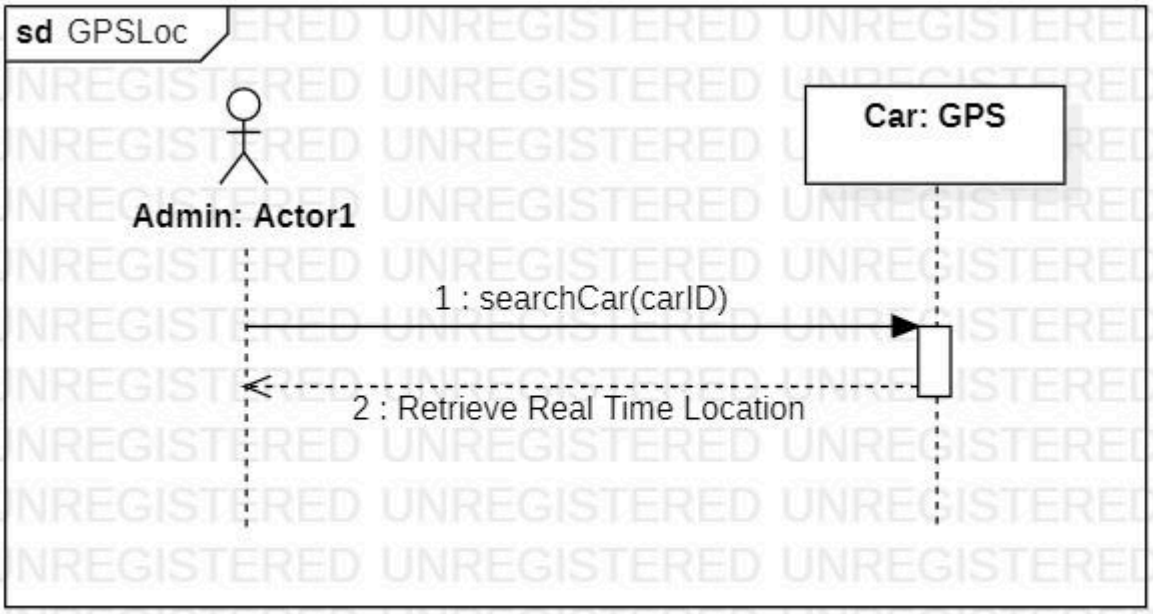


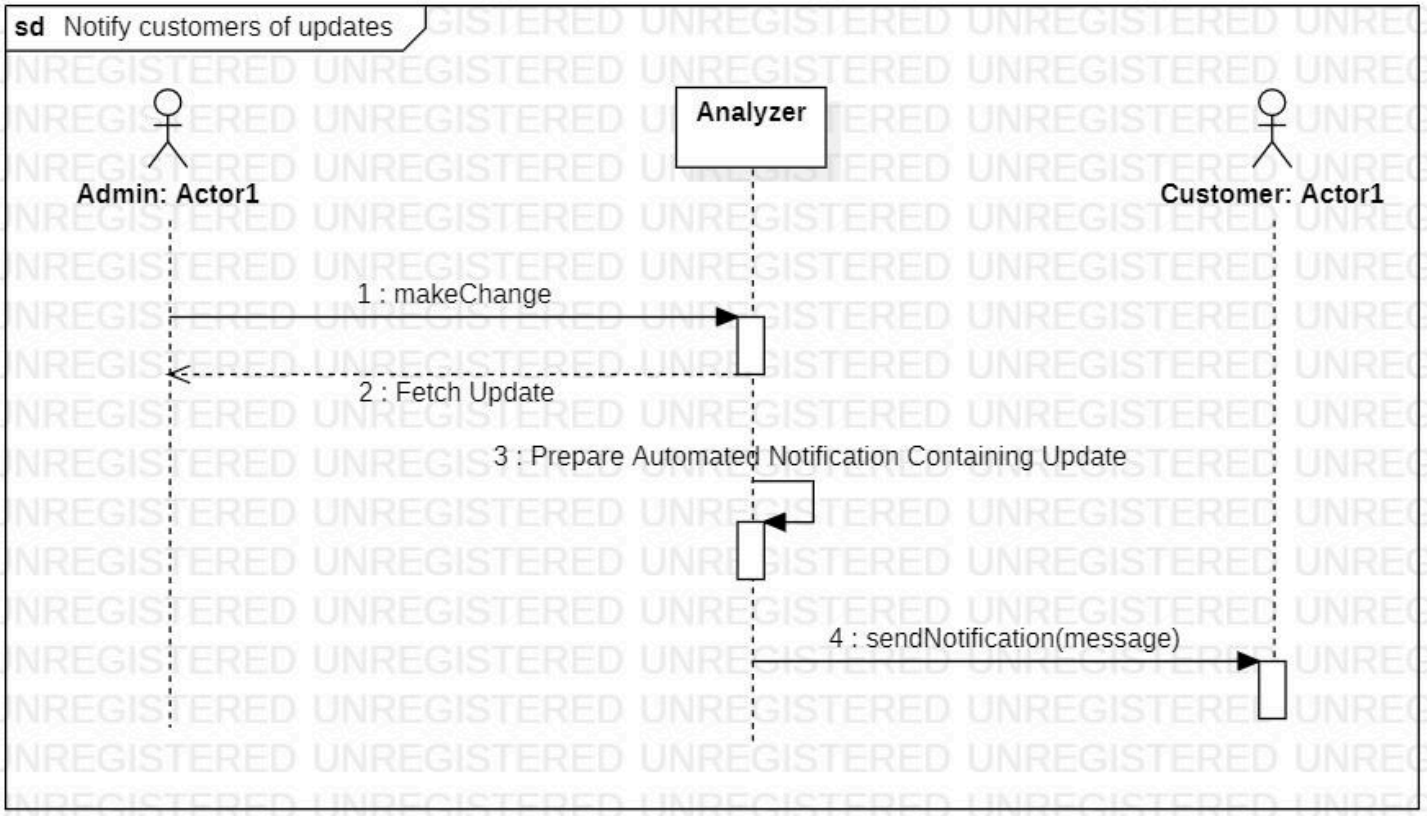


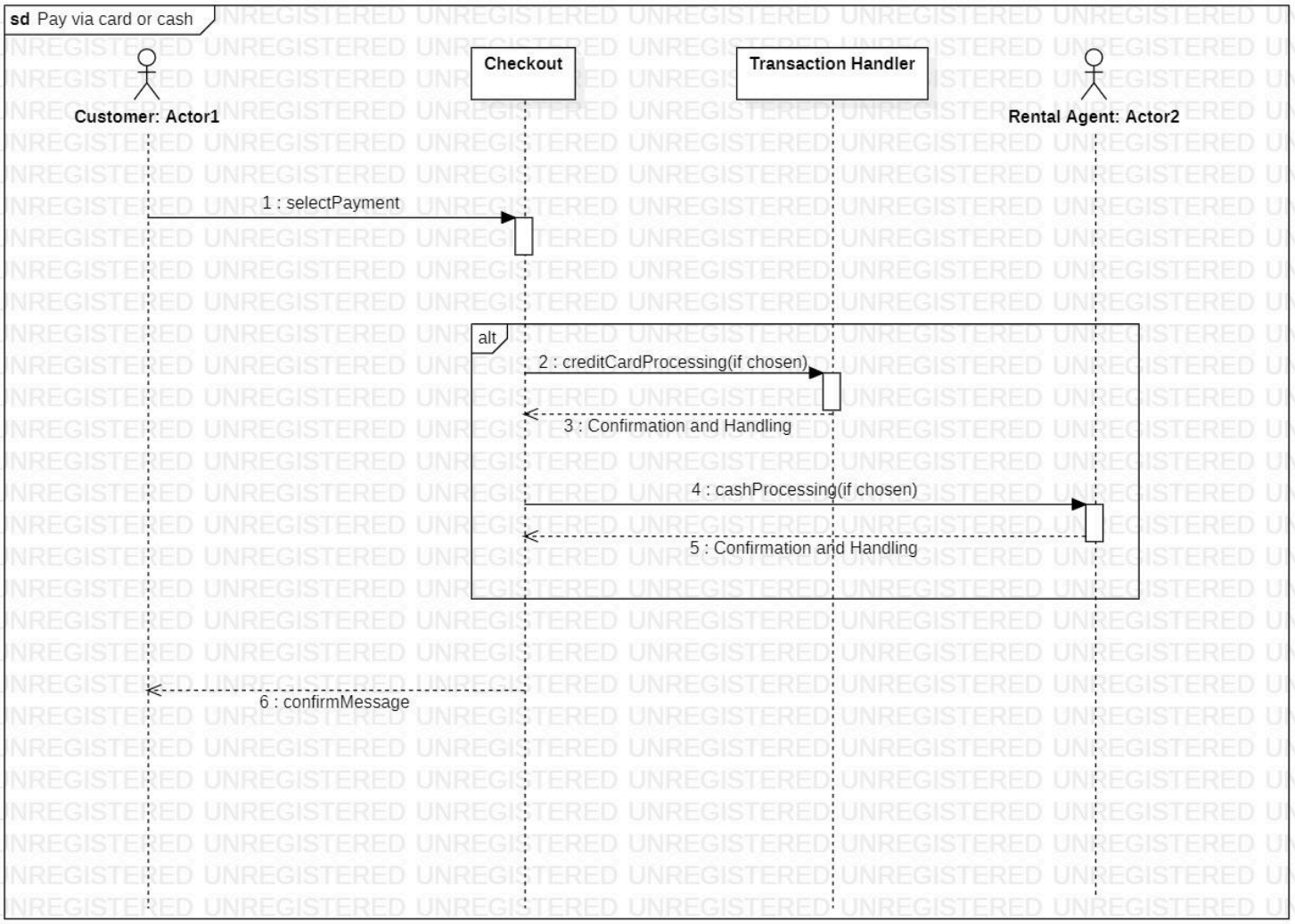

sd Add User
UserStack
Admin: Actor1
1 : addUser(credentials)
2 : Confirm User Addition
3.createUser
New User

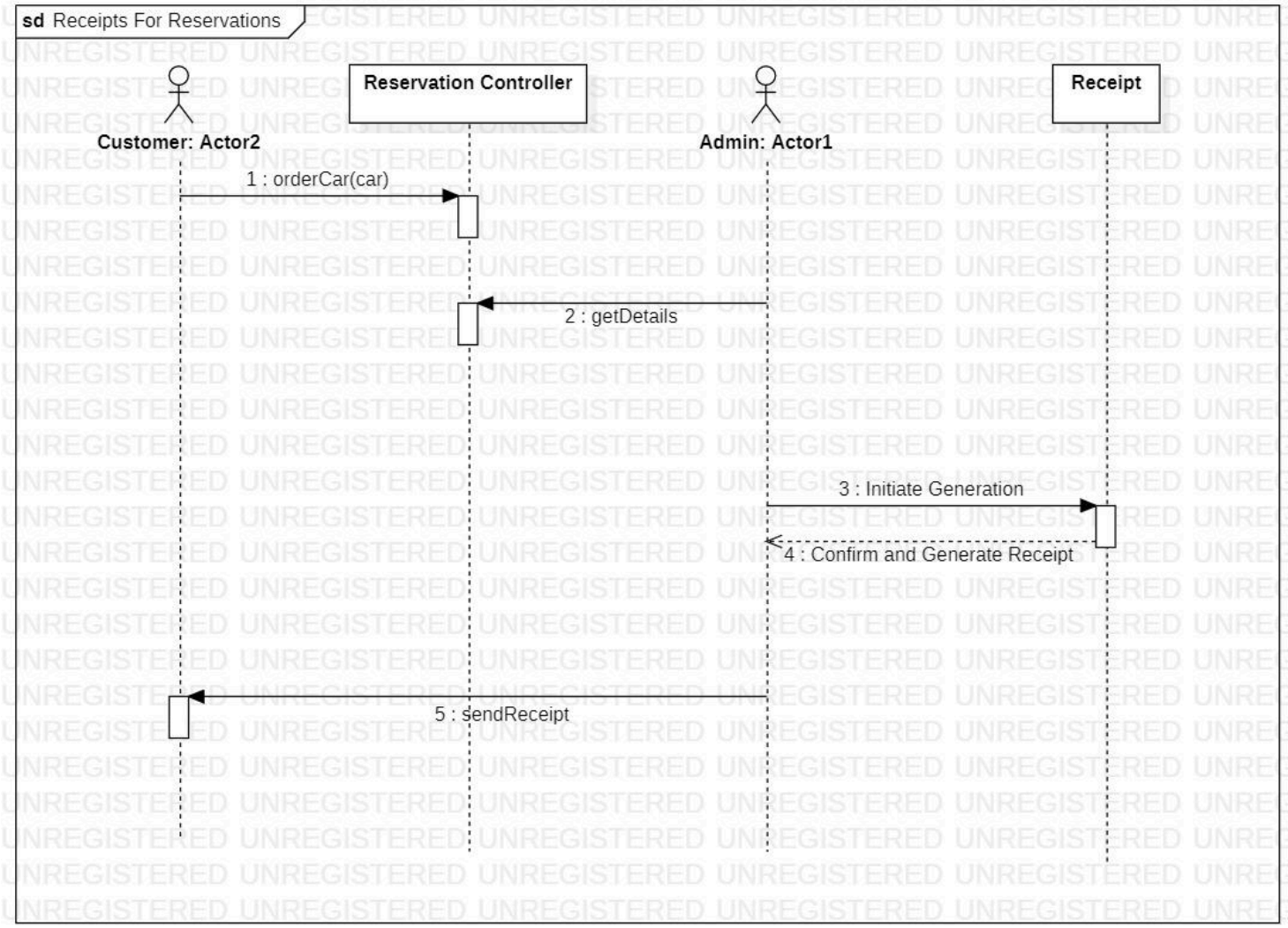


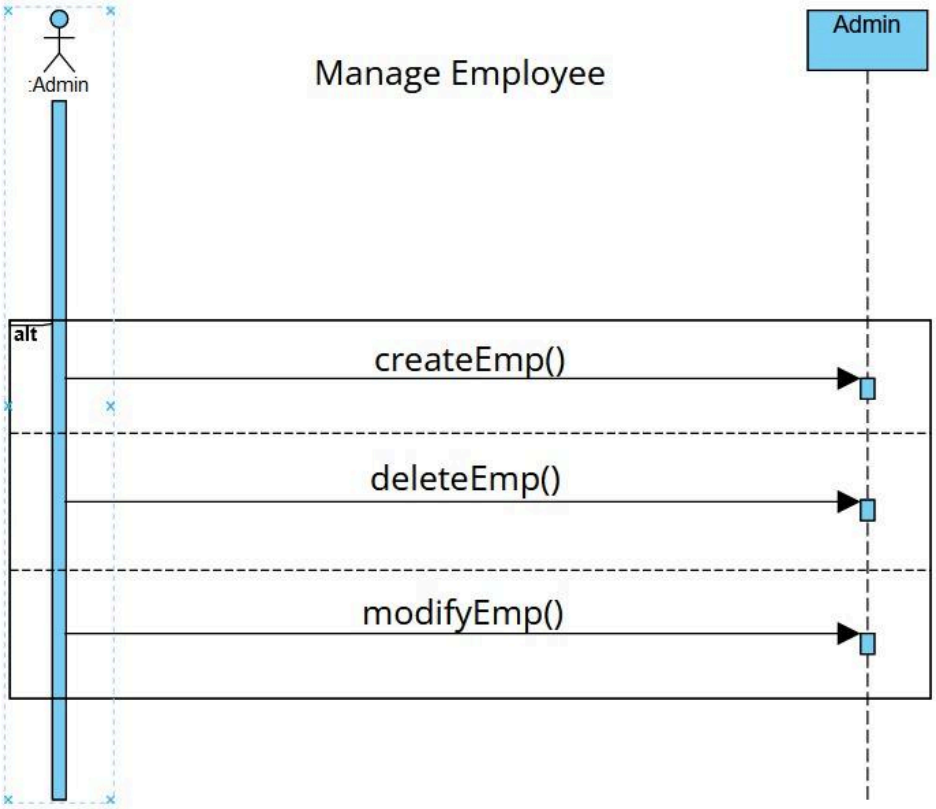

sd Feedback And Rating
Customer: Actor1
Feedback Section
Reservation Controller
1 : provideFeedback
2 : confirm
3 : updatesRating
4 : confirmsRating

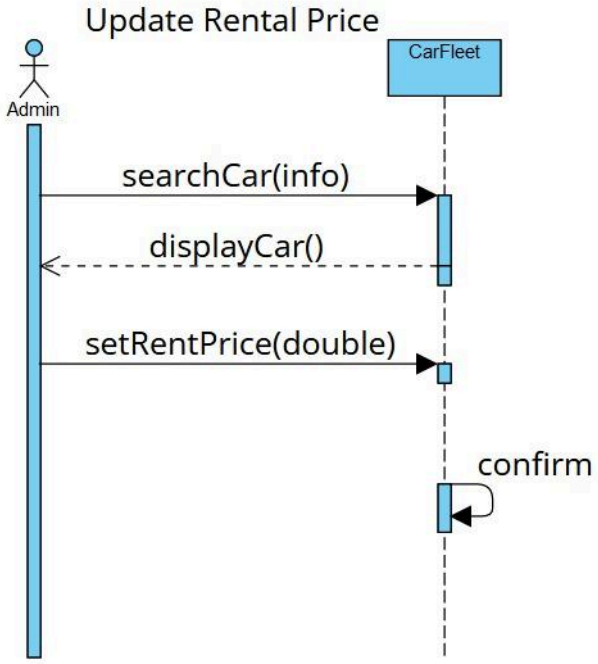

sd GPSLoc
Admin: Actor1
Car: GPS
1 : searchCar(carID)
2 : Retrieve Real Time Location

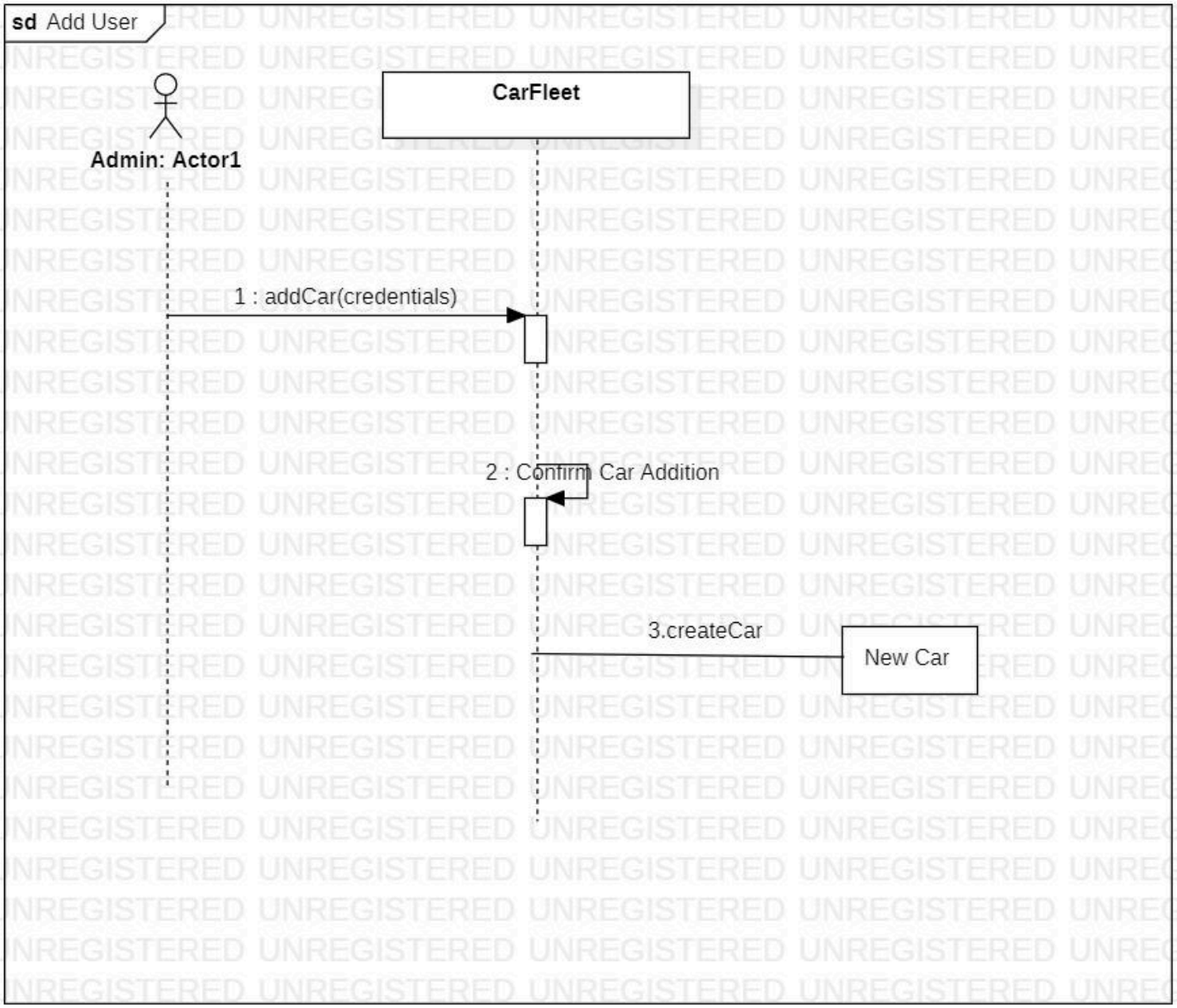

sd Notify customers of updates
Admin: Actor1
Analyzer
Customer: Actor1
1 : makeChange
2 : Fetch Update
3 : Prepare Automated Notification Containing Update
4 : sendNotification(message)

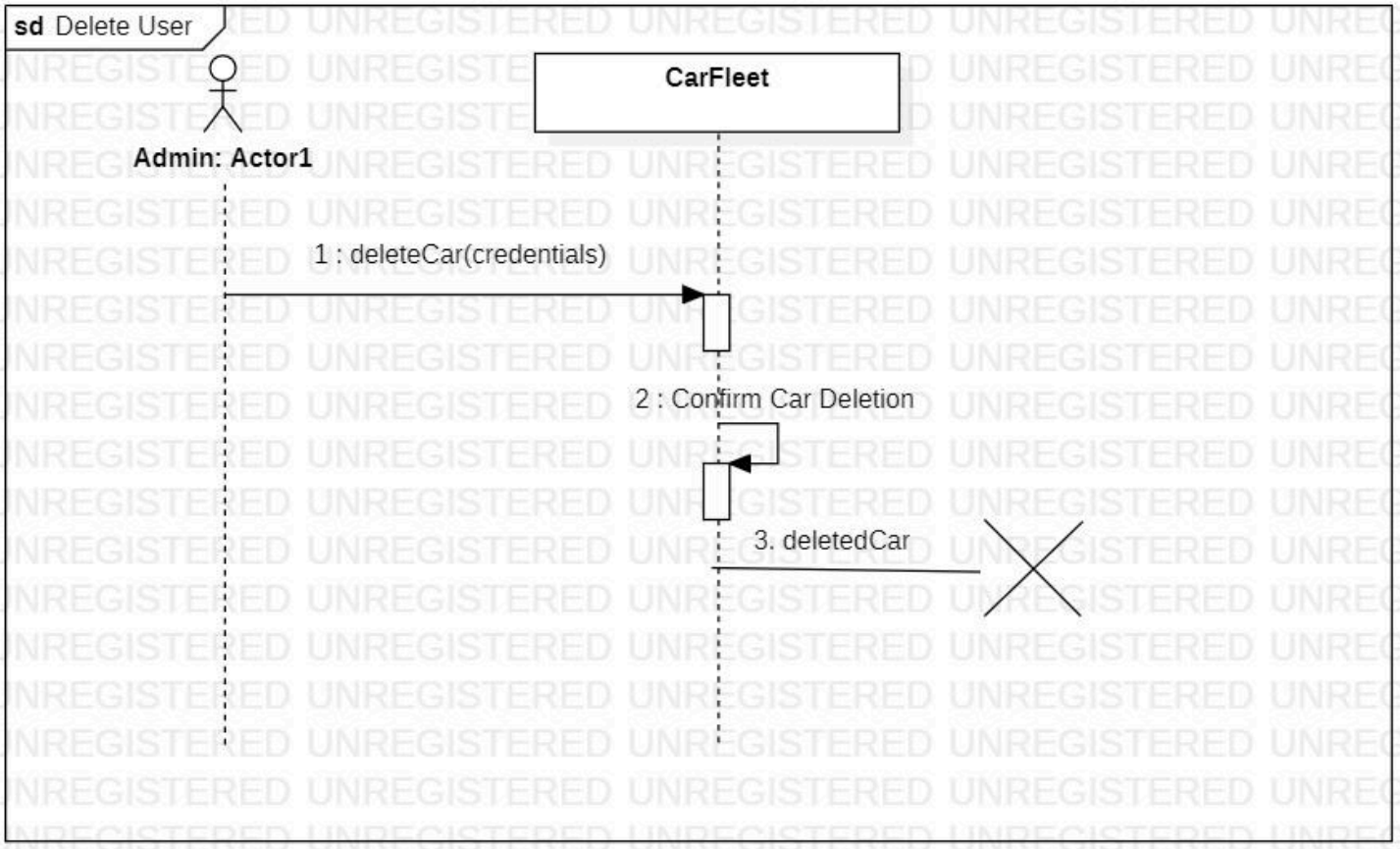

sd Pay via card or cash
Customer: Actor1
Checkout
Transaction Handler
Rental Agent: Actor2
1 : selectPayment
alt
2 : creditCardProcessing(if chosen)
3 : Confirmation and Handling
4 : cashProcessing(if chosen)
5 : Confirmation and Handling
6 : confirmMessage

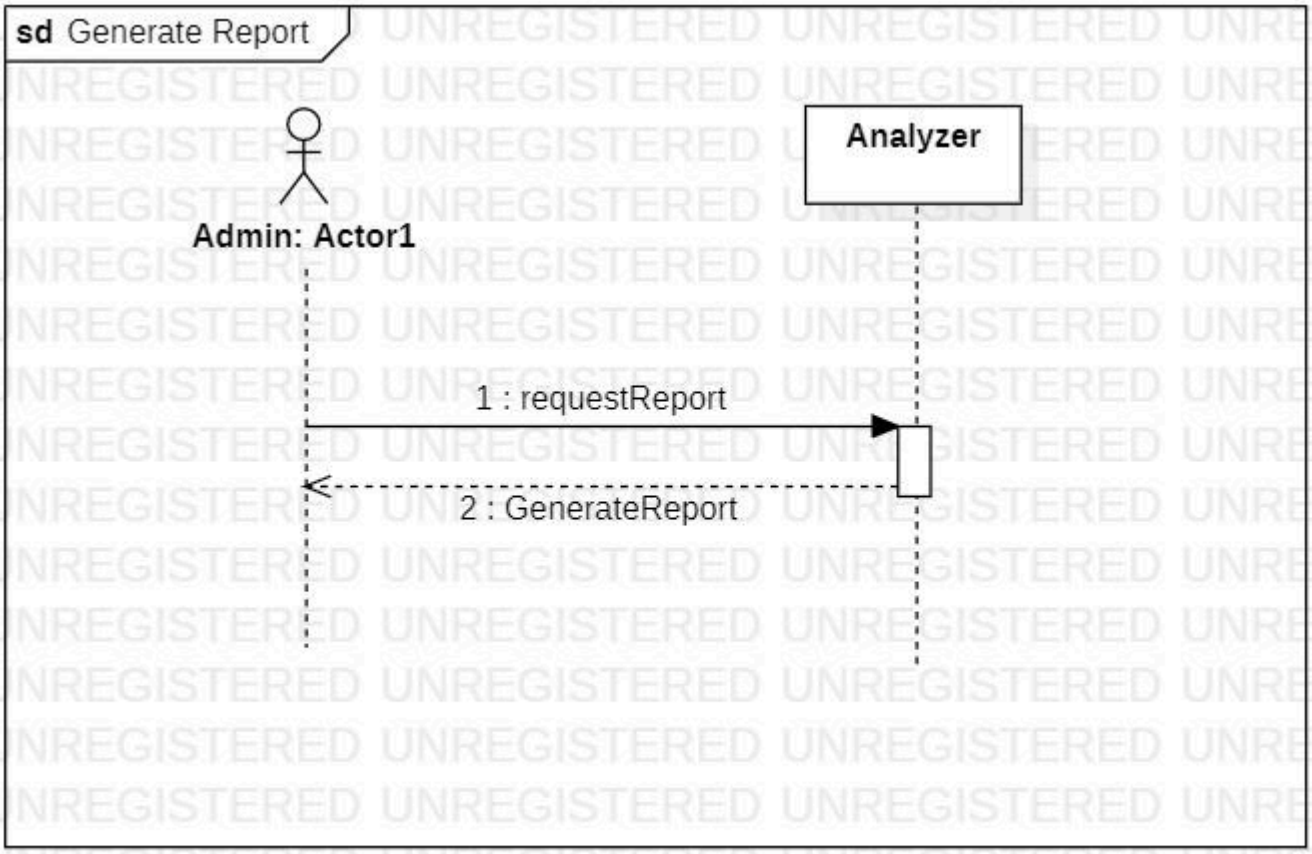


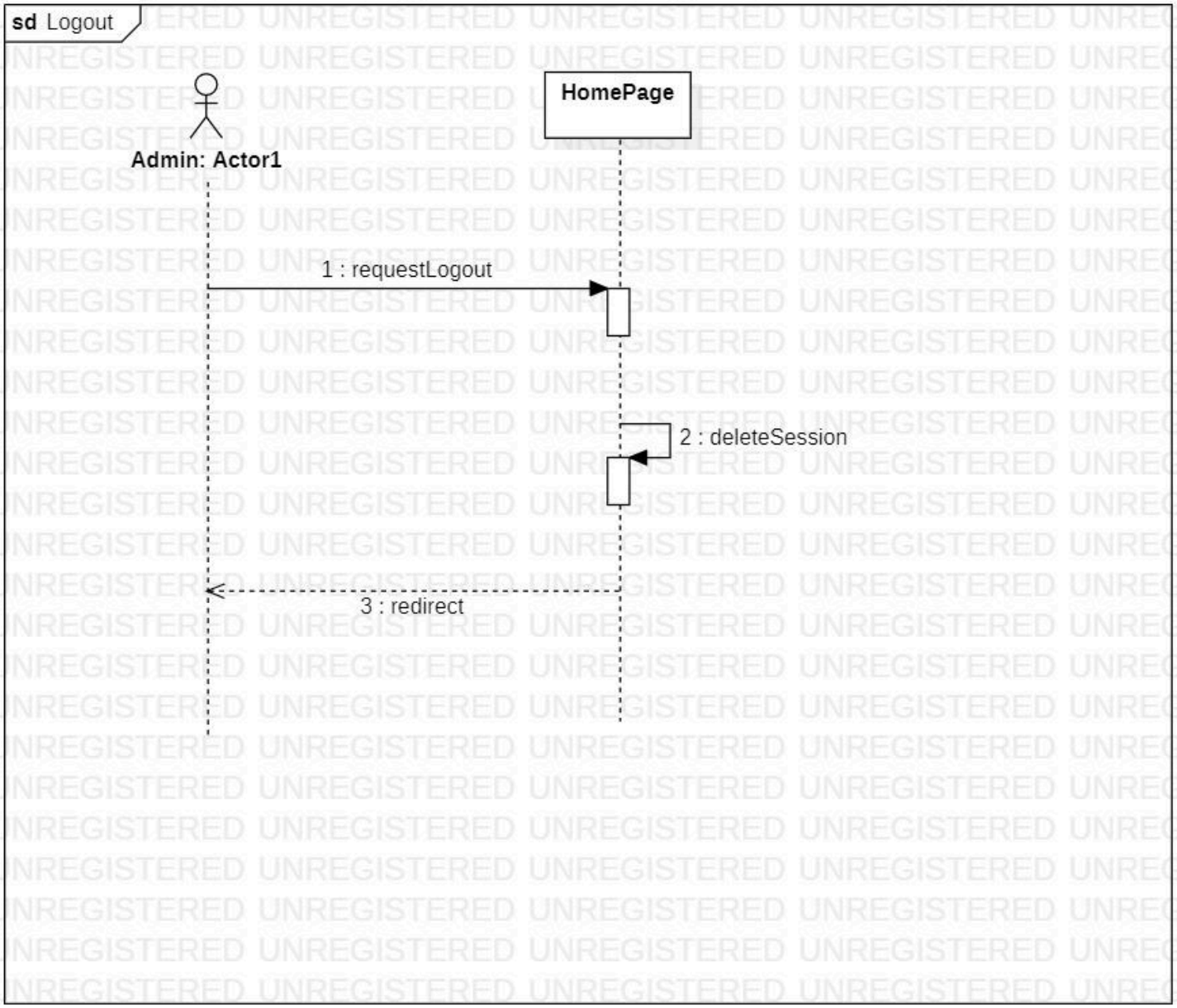

Manage Employee
:Admin
Admin
alt
createEmp()
deleteEmp()
modifyEmp()

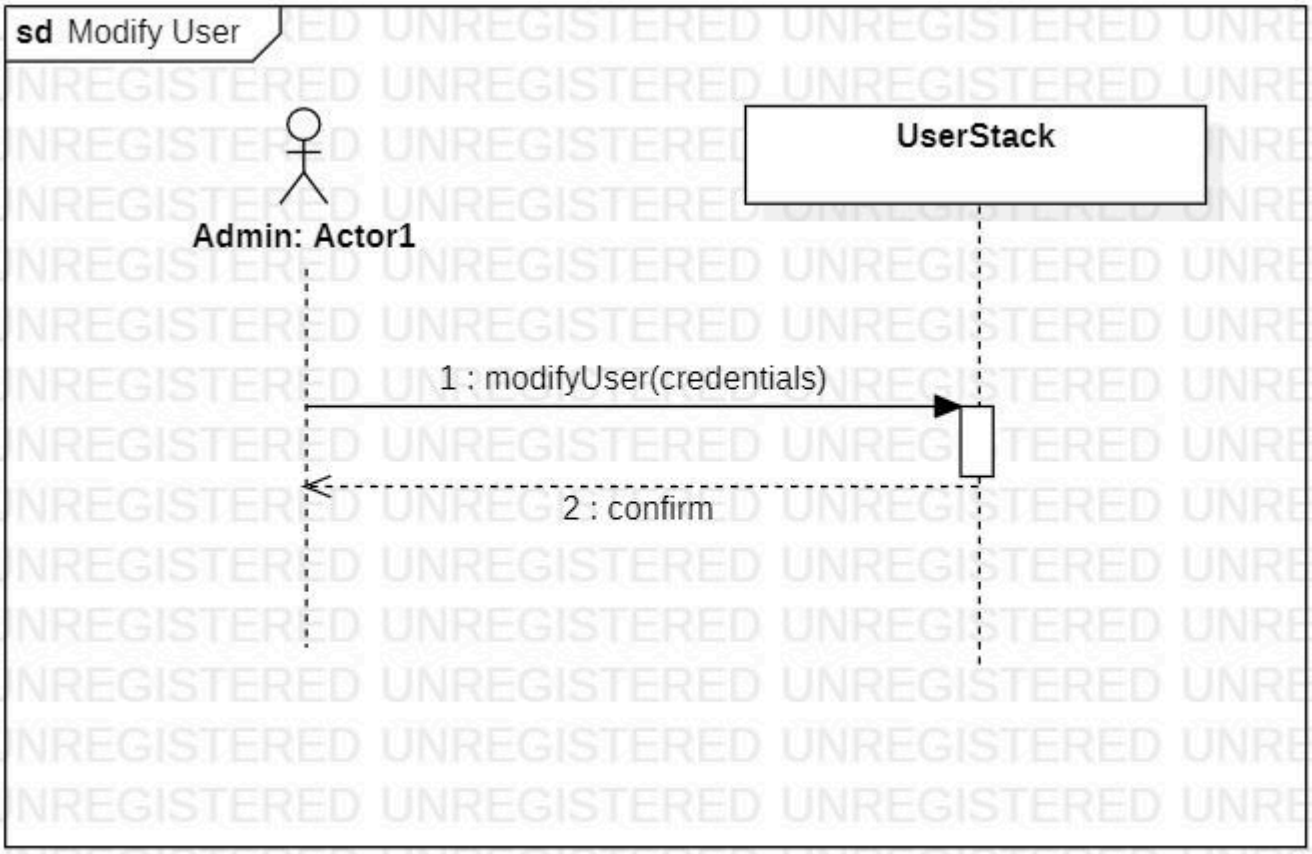

Update Rental Price
Admin
CarFleet
searchCar(info)
displayCar()
setRentPrice(double)
confirm

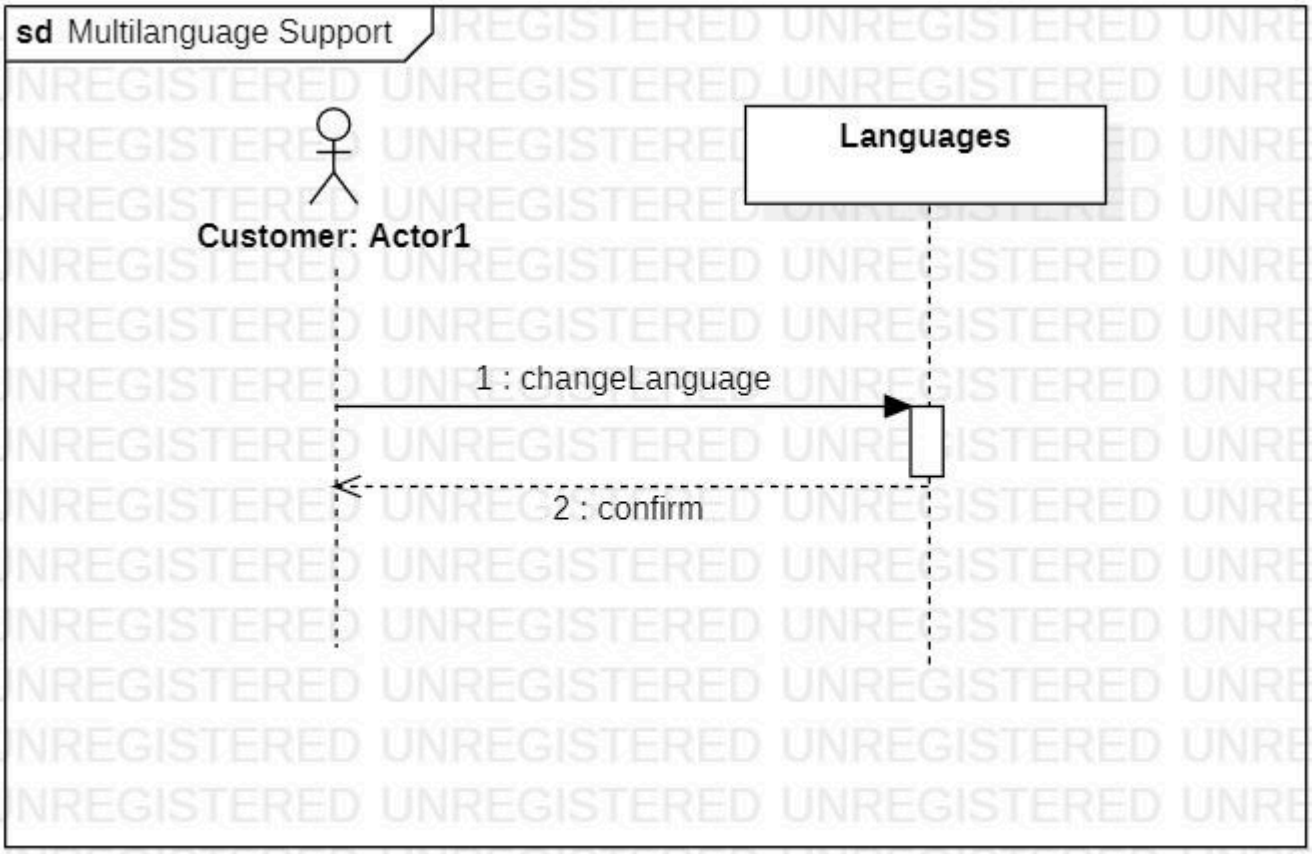

sd Add User
CarFleet
Admin: Actor1
1 : addCar(credentials)
2 : Confirm Car Addition
3.createCar
New Car

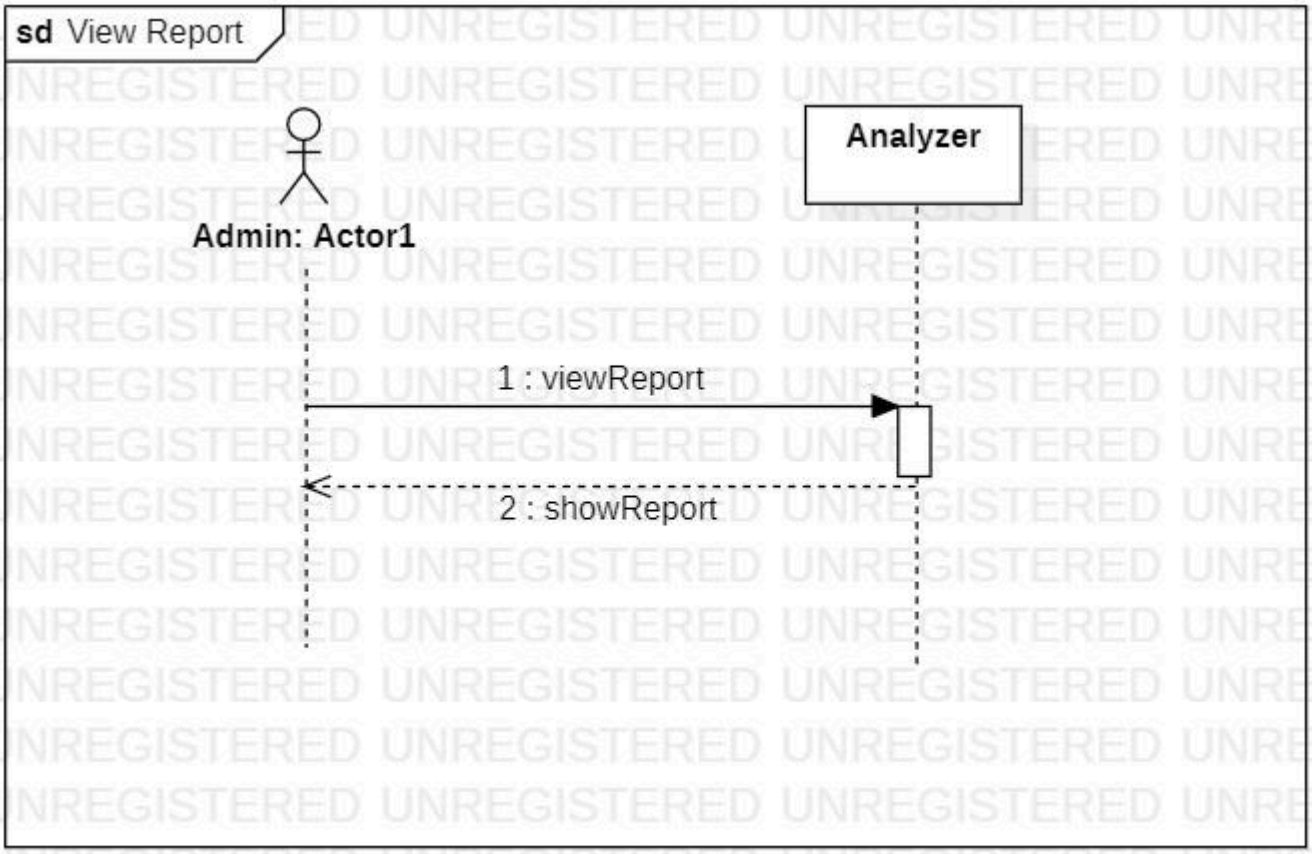






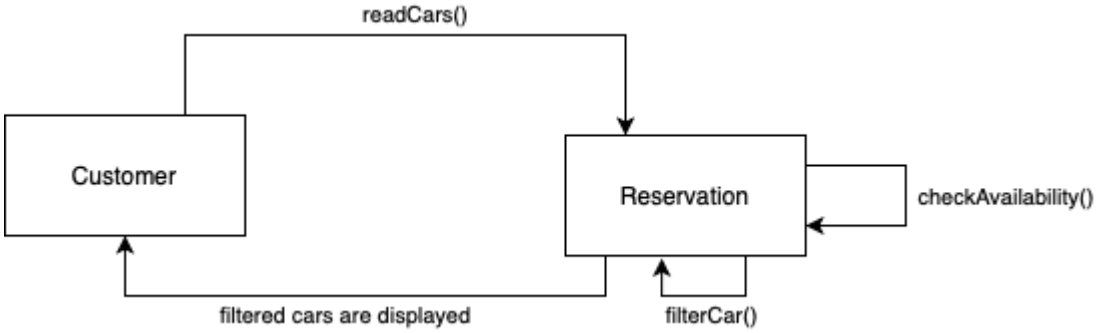




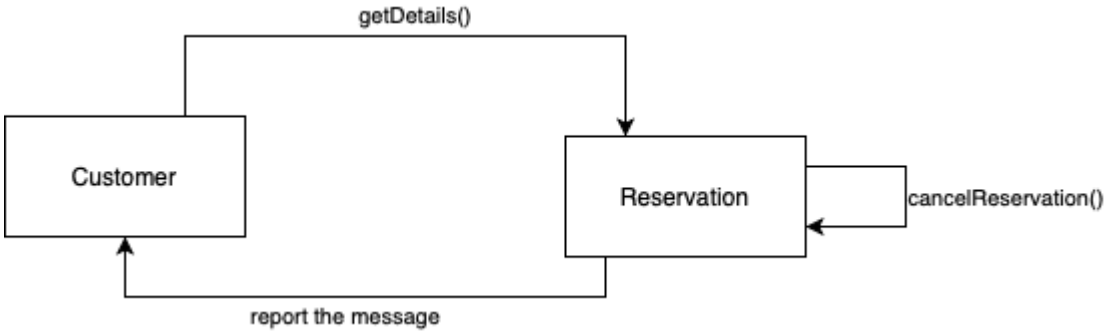


4.9 Collaboration diagrams

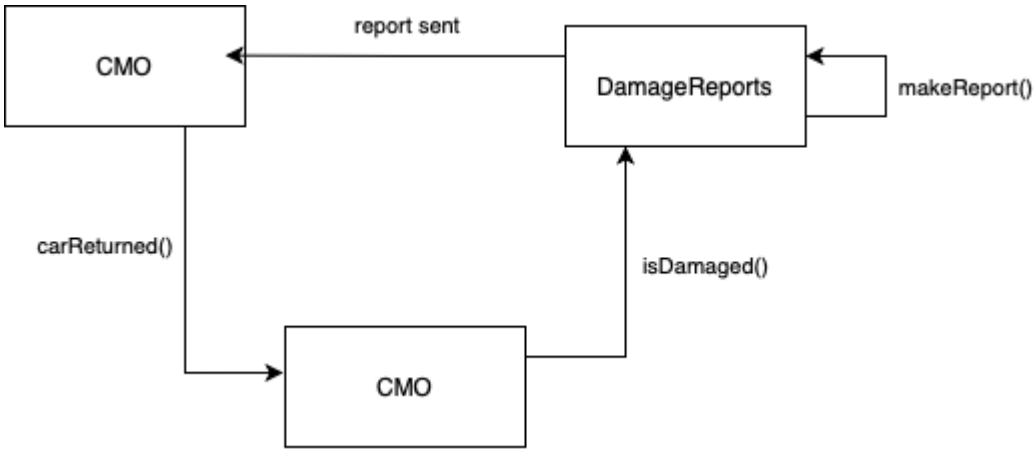
Browse and filter cars



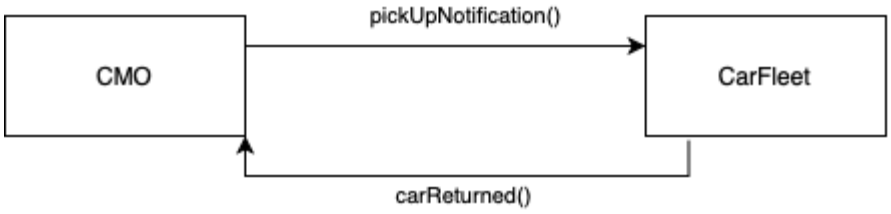
Cancel reservation



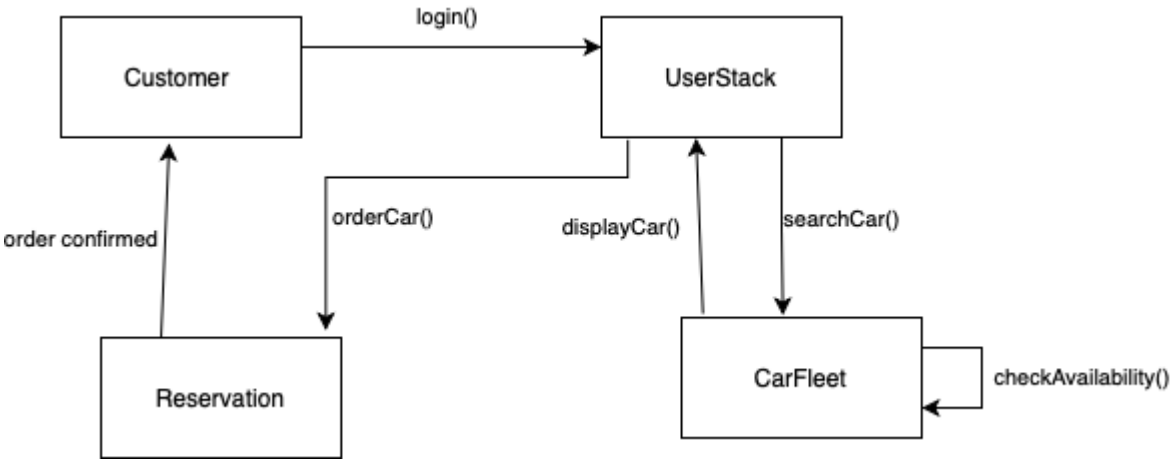
Report damages or fines



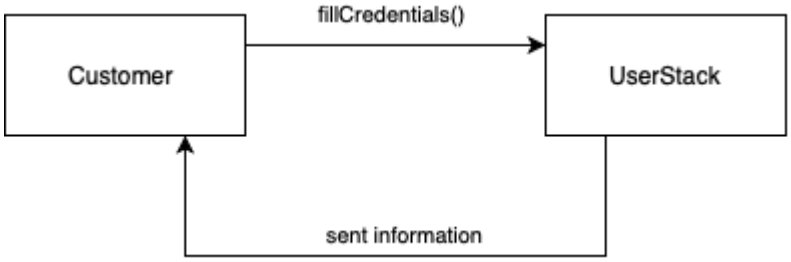
Report when car is returned



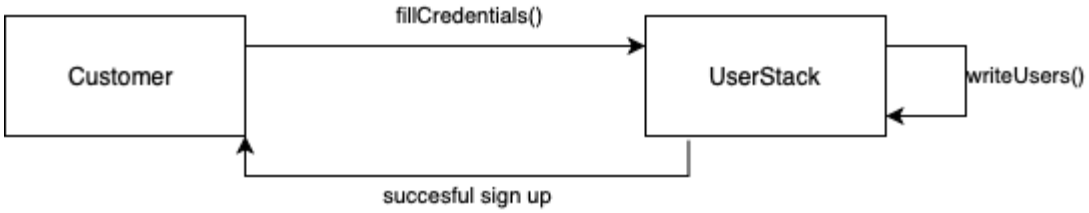
Make reservation



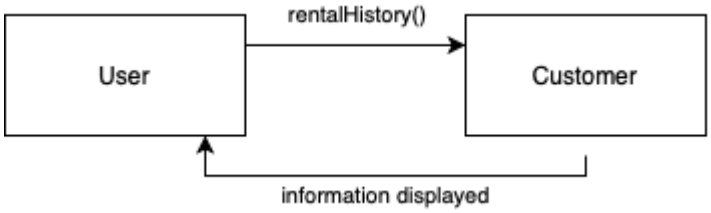
Sign In



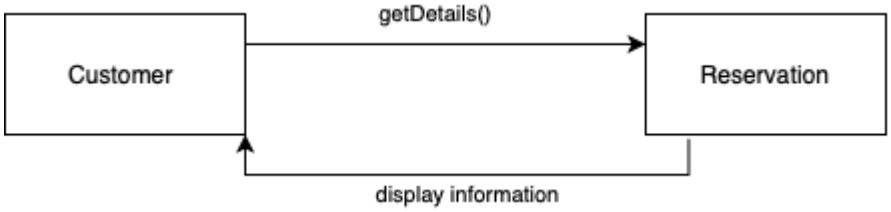
Sign Up



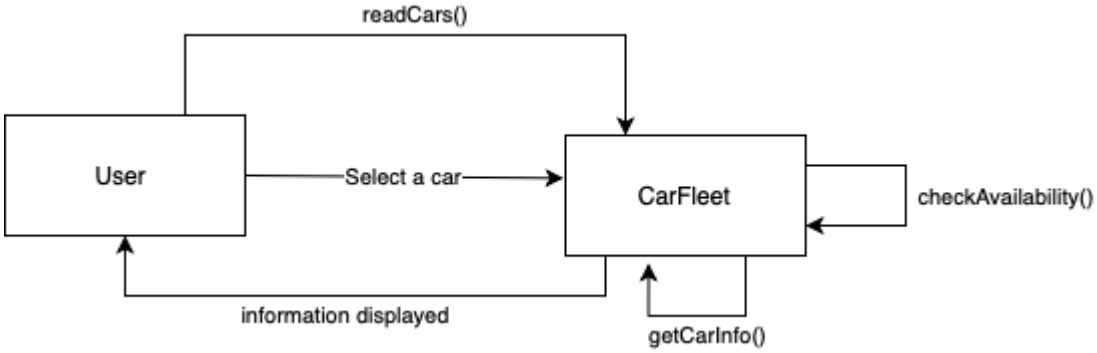
View rental history



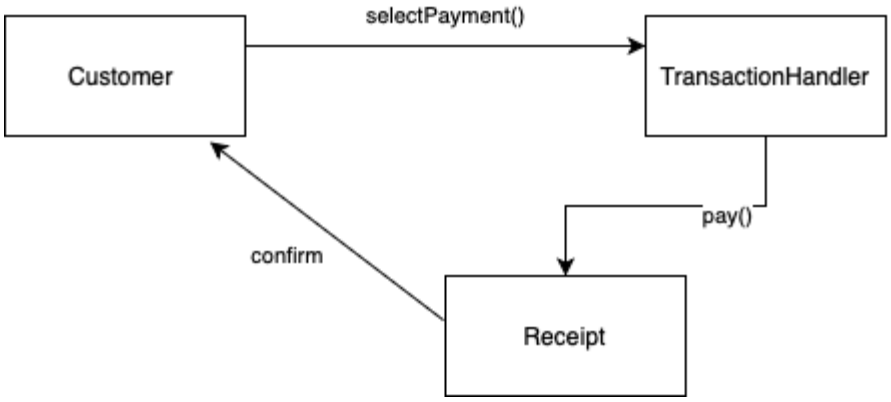
View reservation

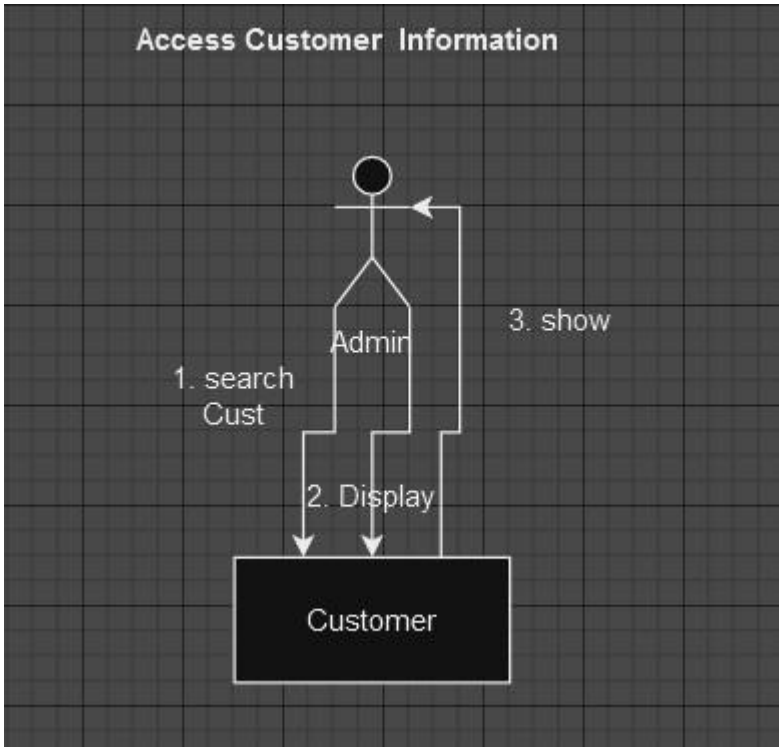


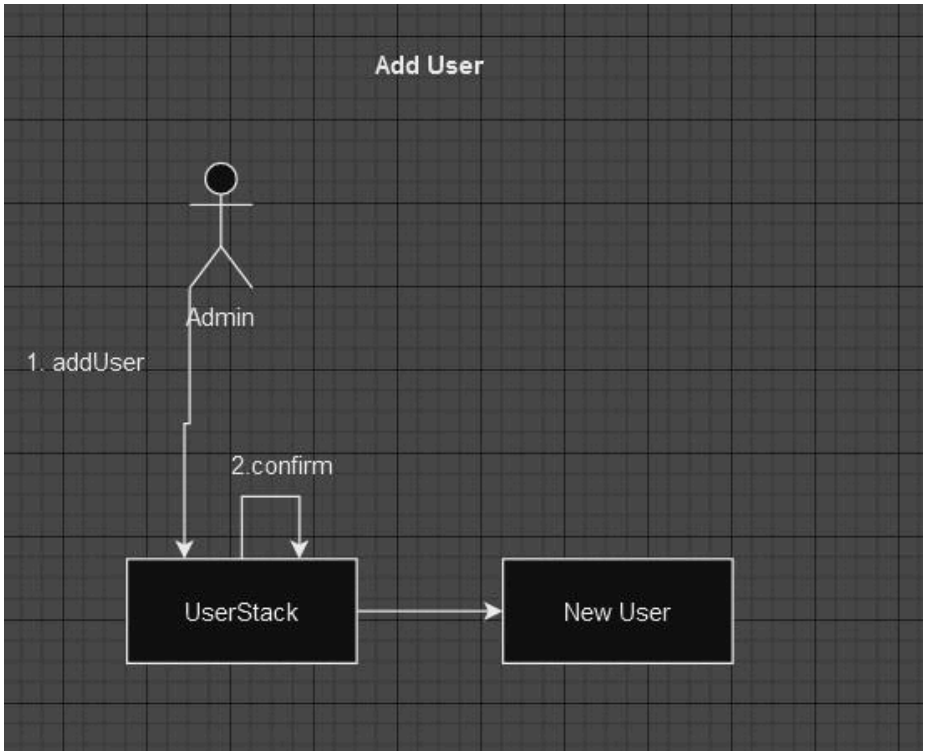
View car specification

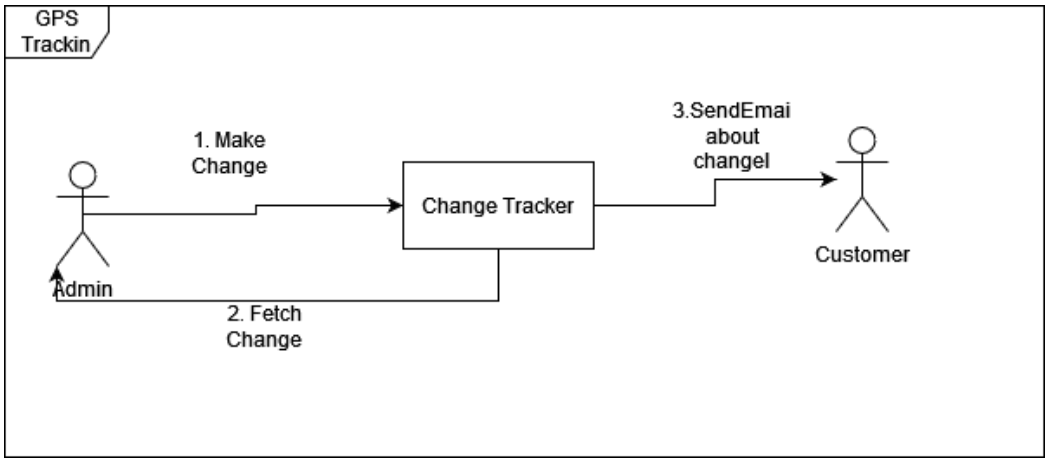


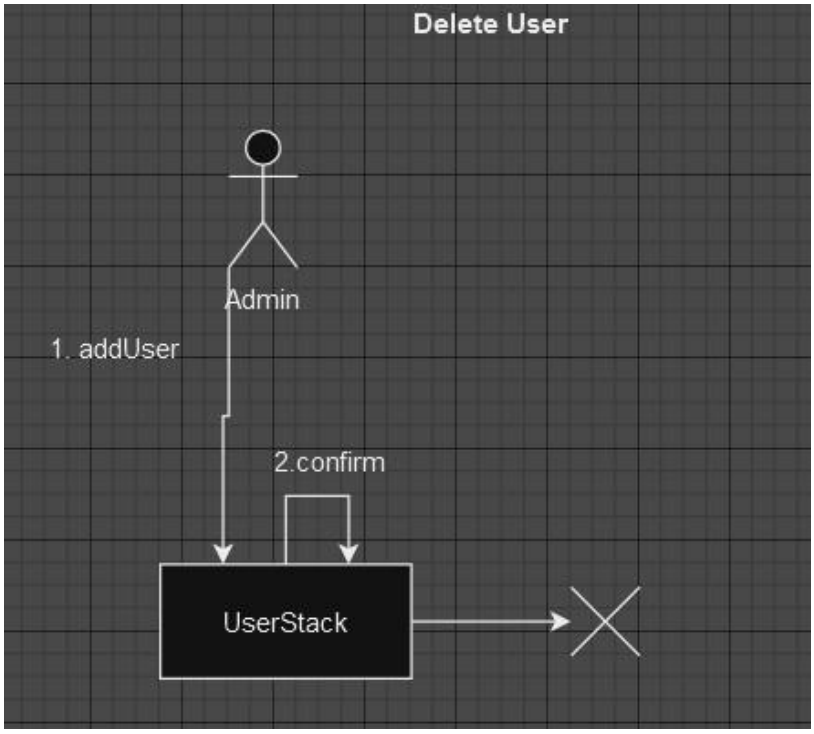
Pay via card

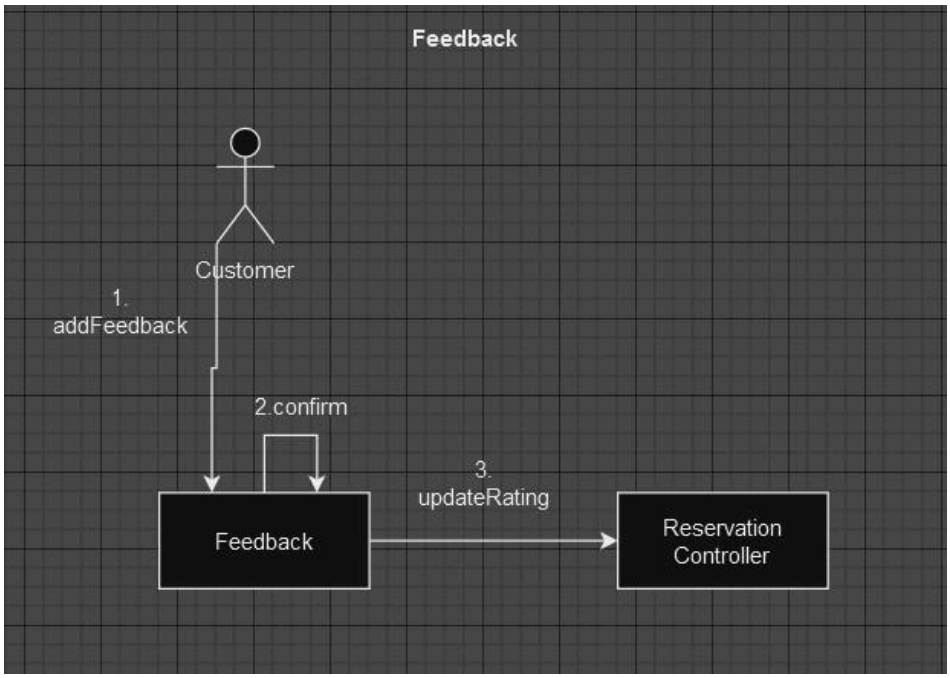

selectPayment()
Customer
TransactionHandler
pay()
confirm
Receipt

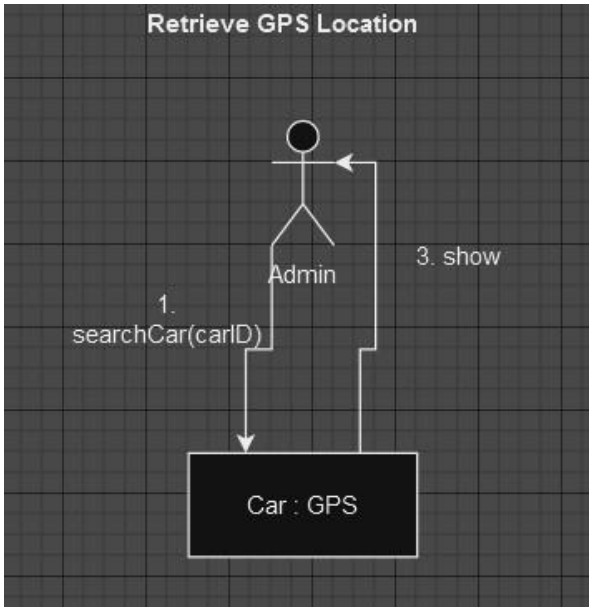

Access Customer Information
Admin
1. search
Cust
2. Display
3. show
Customer

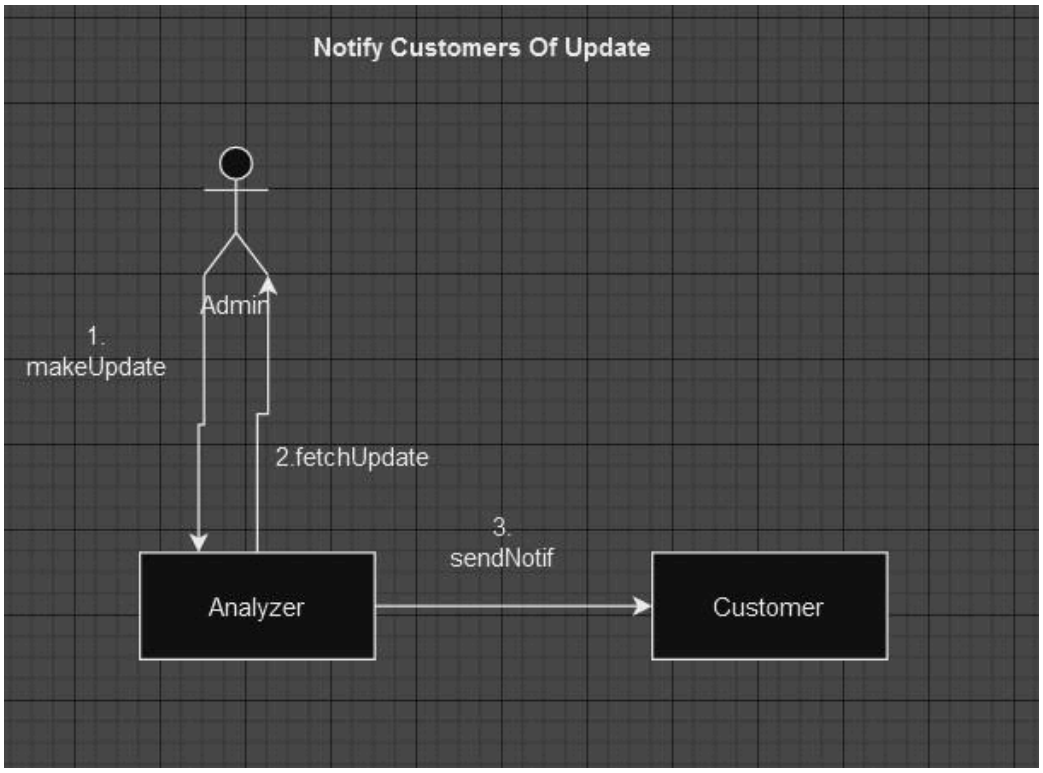

Add User
Admin
1. addUser
2.confirm
UserStack
New User

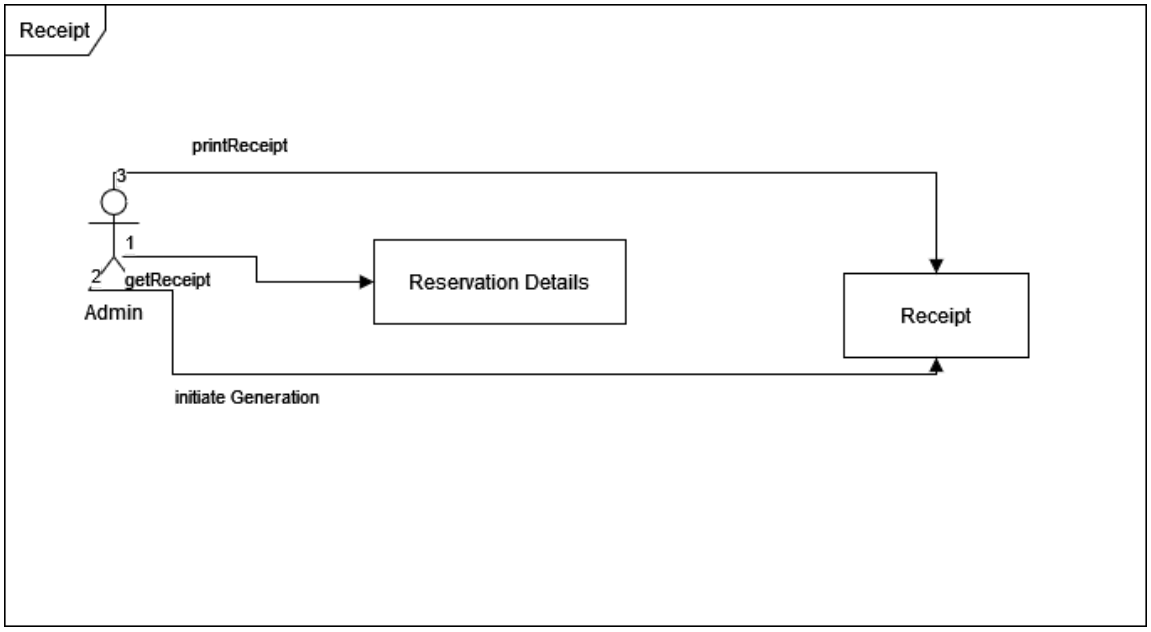

GPS
Trackin
1. Make
Change
Change Tracker
3.SendEmai
about
changel
Customer
Admin
2. Fetch
Change

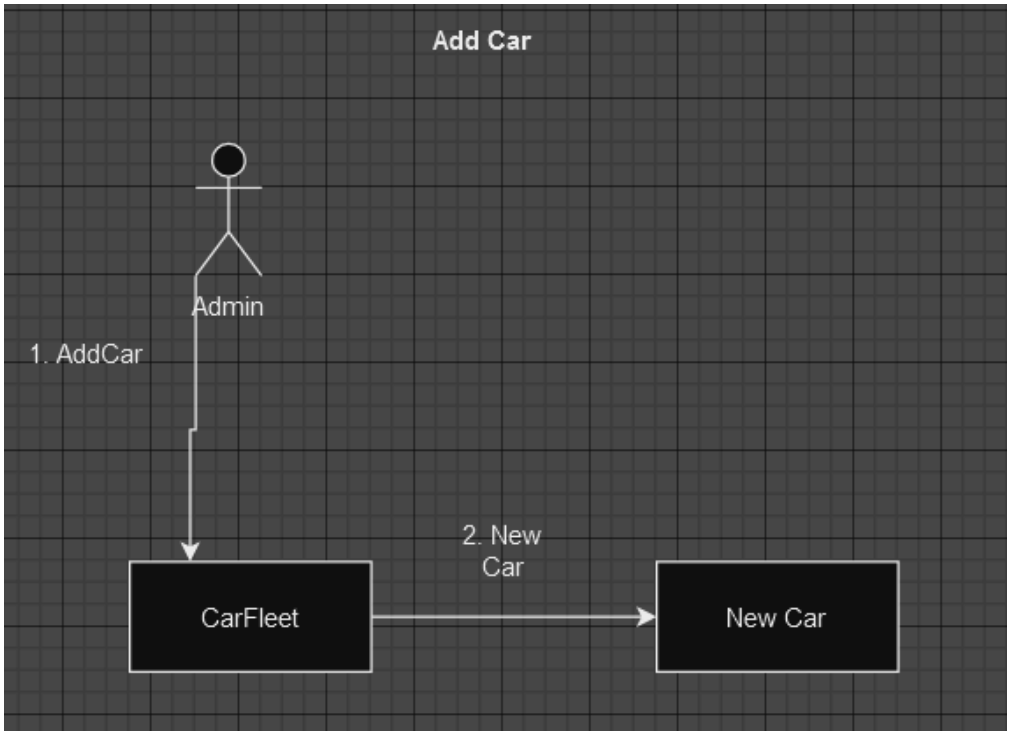

Delete User
Admin
1. addUser
2.confirm
UserStack

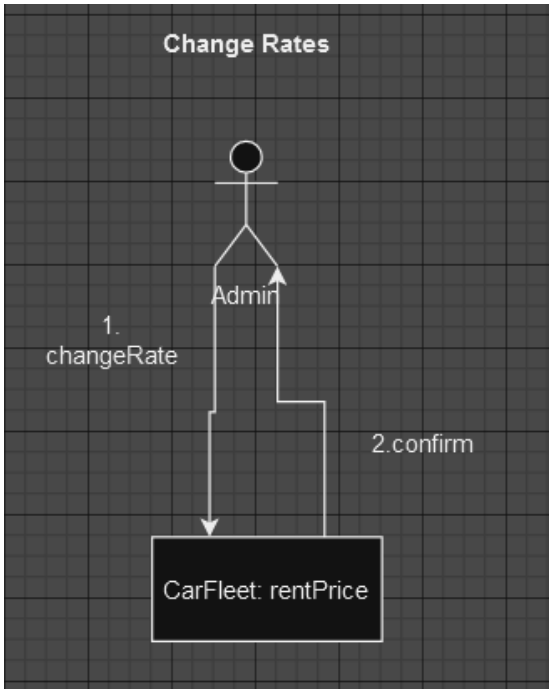

Feedback
Customer
1.
addFeedback
2.confirm
3.
updateRating
Feedback
Reservation
Controller

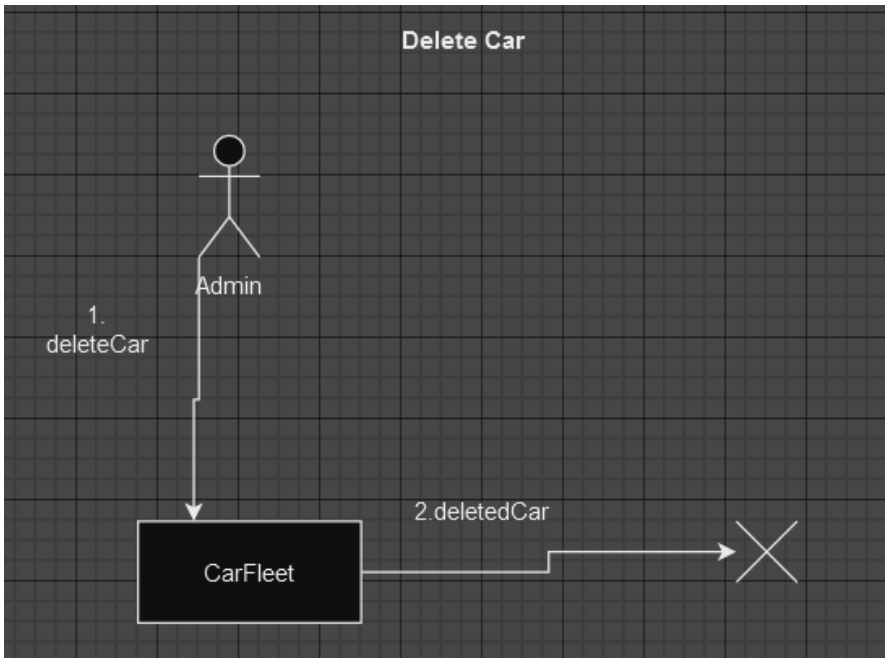

Retrieve GPS Location
Admin
1.
searchCar(carID)
3. show
Car : GPS

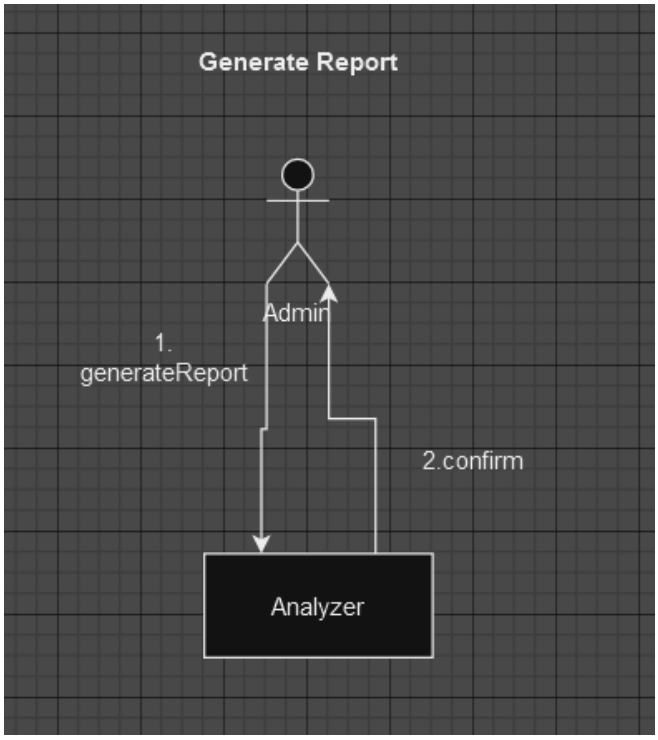

Notify Customers Of Update
Admin
1.
makeUpdate
2.fetchUpdate
3.
sendNotif
Analyzer
Customer


Receipt
printReceipt
3
1
2
getReceipt
Admin
Reservation Details
Receipt
initiate Generation

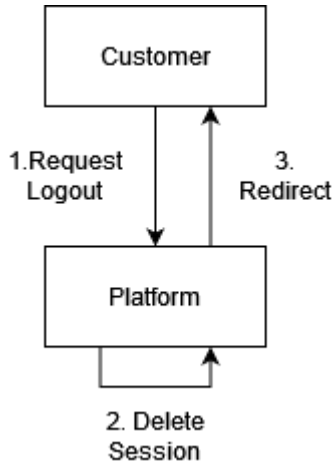

Add Car
Admin
1. AddCar
2. New Car
CarFleet
New Car

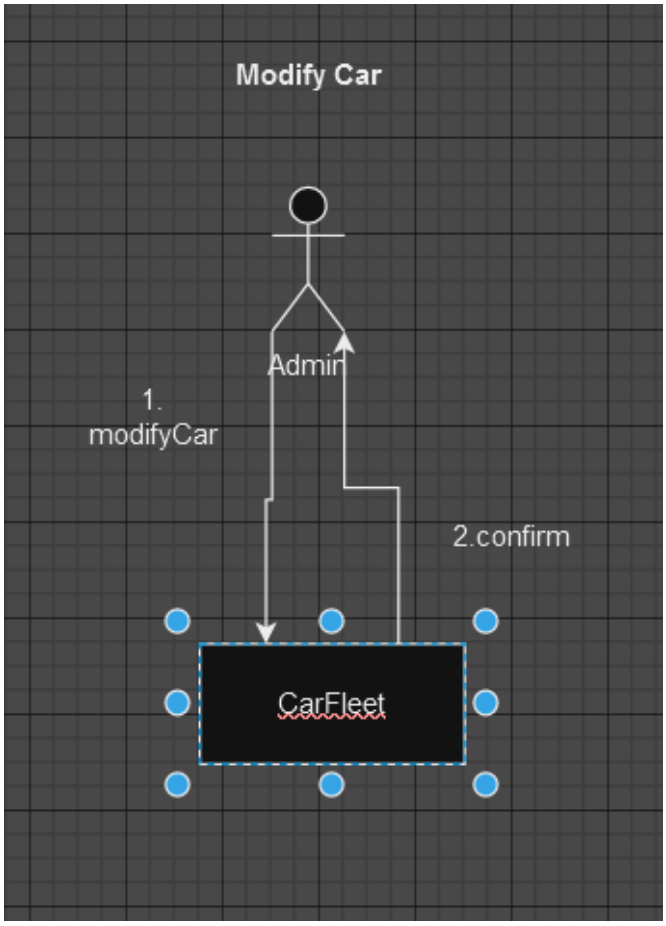

Change Rates
Admin
1.
changeRate
2.confirm
CarFleet: rentPrice

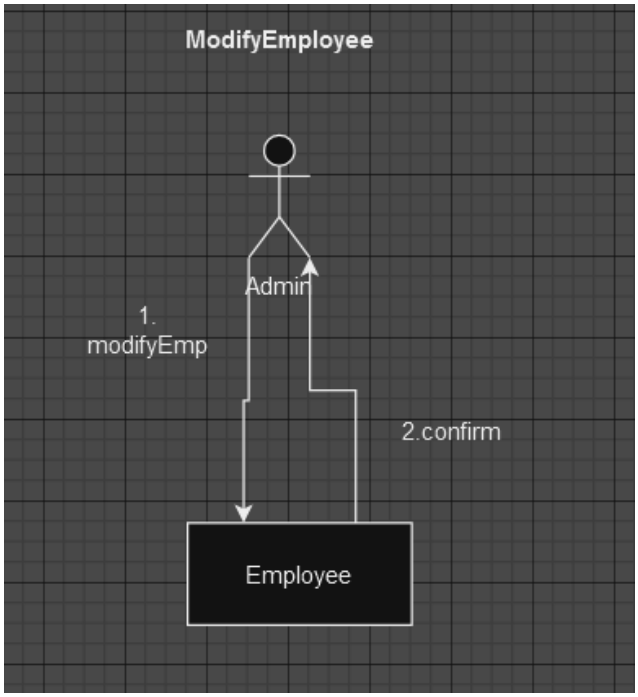

Delete Car
Admin
1.
deleteCar
2.deletedCar
CarFleet

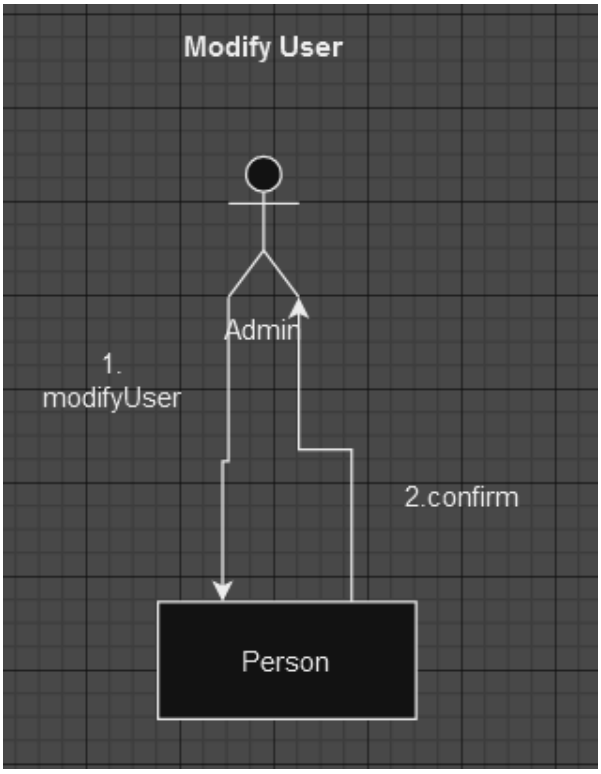

Generate Report
Admin
1.
generateReport
2.confirm
Analyzer

Logout

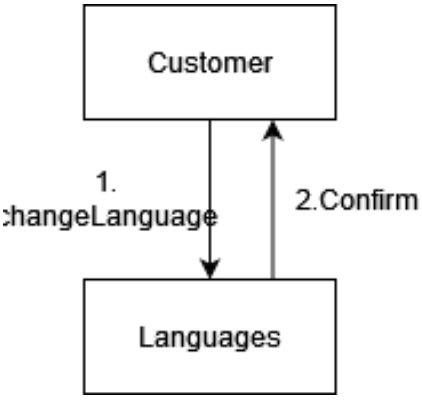

Customer
1.Request
Logout
3.
Redirect
Platform
2. Delete
Session

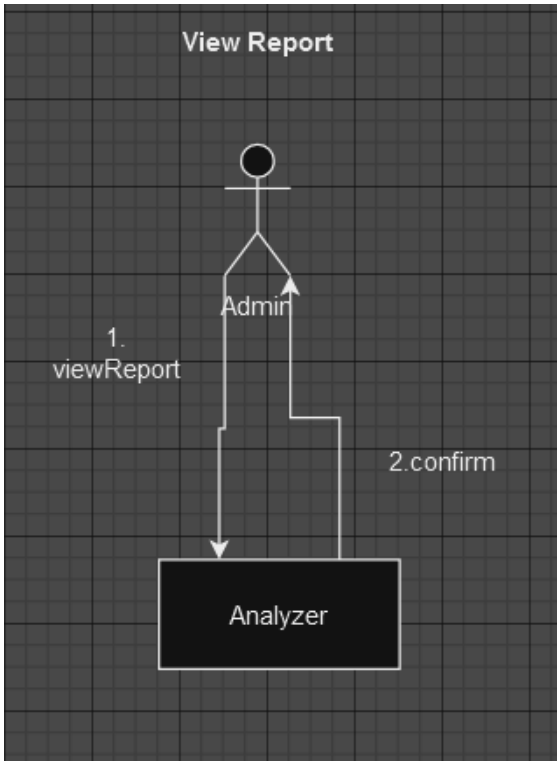

Modify Car
Admin
1.
modifyCar
2.confirm
CarFleet


ModifyEmployee
Admin
1.
modifyEmp
2.confirm
Employee

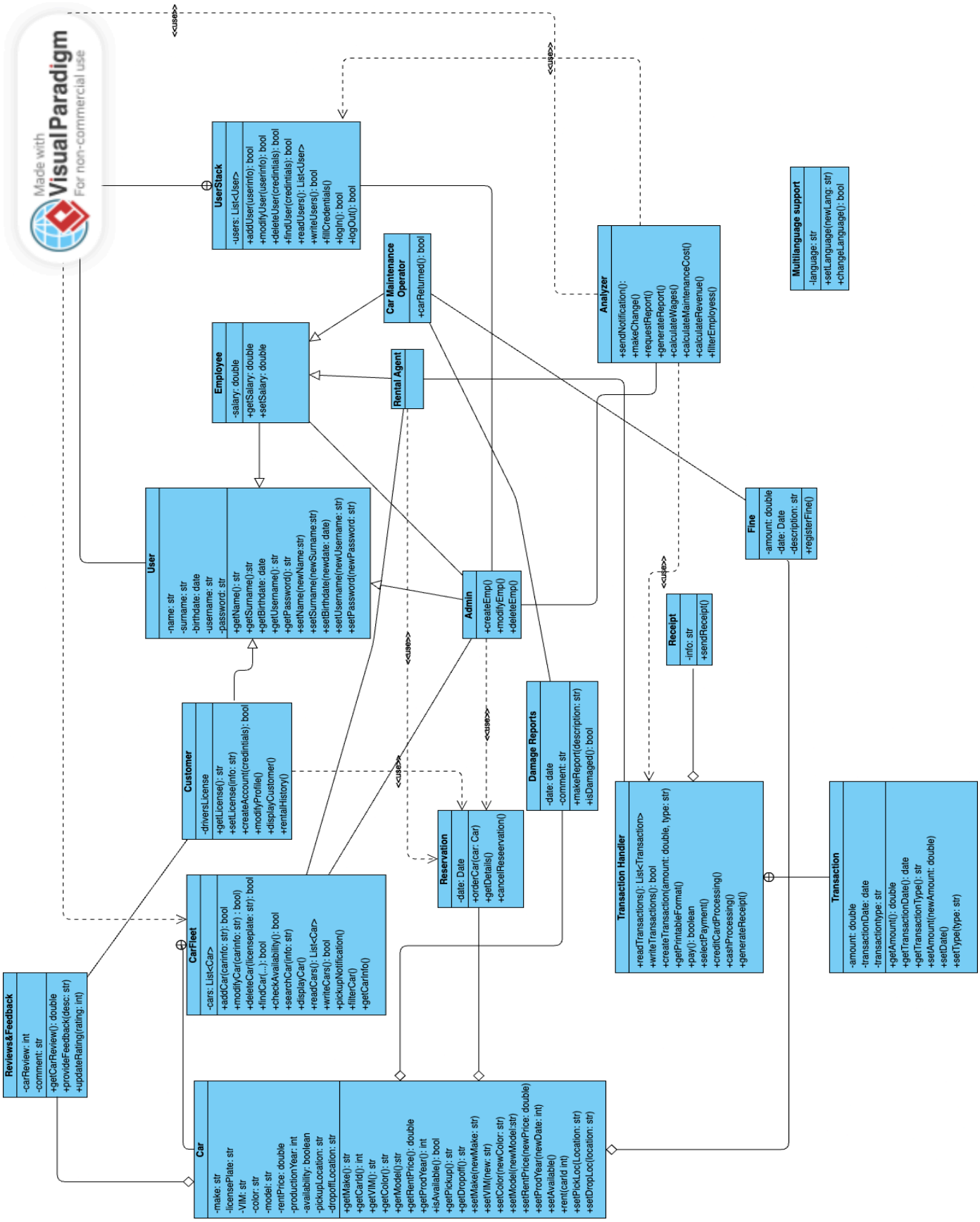


Multilanguage support




View Report
Admin
1.
viewReport
2.confirm
Analyzer

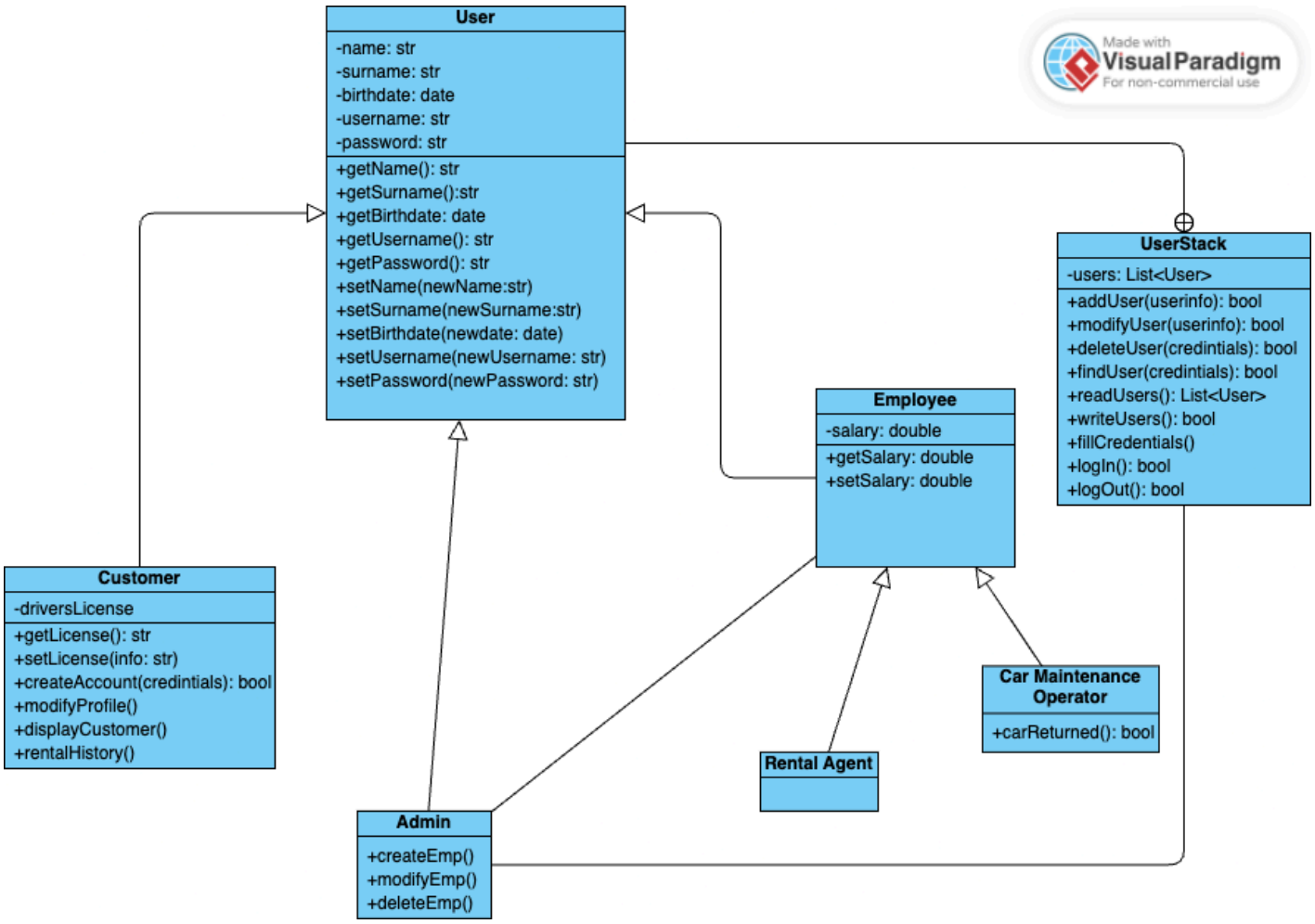
4.10 Class Diagrams



5. Design Patterns

Bridge

We have used the bridge pattern in the way we have designed the user levels. Instead of having a field in the class person which is a Boolean or string that represents the user's level we have created the classes Admin and Employer. The latter also is derived into Agent and Car Maintenance operator.



Singleton

There is always the same instance of the class analyzer that interacts with the other classes. The classes that are retrieving or are giving information to the analyzer are not aware that this is the same instance all the time the program is running.