HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

KHOA CÔNG NGHỆ THÔNG TIN 1

MÔN: NHẬP MÔN CÔNG NGHỆ PHẦN MỀM

Lớp **:** E22CQCN02-B
Nhóm bài tập lớn: 02
Đề tài bài tập lớn: Garage car service management
Danh sách nhóm : 1. Trần Xuân Kiên - B22DCVT269
2. Trịnh Diệu Linh - B22DCVT310
3. Vũ Ngọc Linh - B22DCDT179
4. Trần Nhật Minh - B22DCVT349
Module **:** Receiving car
Yêu cầu **:** Testing phase

Hà Nội – 2025

A. Requirement

I. Concept exploration (glossary list only)

No	English concept	Vietnamese	Meaning
Concept related to human			
1	Garage/Service Manager	Quản lý Gara/Trung tâm dịch vụ	Oversees overall operations, staff, and customer service.
2	Workshop Supervisor	Giám sát Xưởng sửa chữa	Manages daily workflow and ensures timely repairs.
3	Operations Manager	Quản lý Vận hành	Handles business growth, efficiency, and performance tracking.
4	Customer Service Manager	Quản lý Dịch vụ Khách hàng	Ensures customer satisfaction and service quality.
5	Parts & Inventory Manager	Quản lý Kho & Phụ tùng	Manages stock, orders parts, and tracks usage.
6	Fleet Maintenance Manager	Quản lý Bảo dưỡng Xe	Oversees servicing for commercial vehicle fleets.
7	Warranty Administrator	Nhân viên Xử lý Bảo hành	Processes warranty claims and liaises with manufacturers.
8	Service Advisor	Cố vấn Dịch vụ	Acts as the link between customers and technicians, explaining repairs and costs.
9	Billing & Accounts Specialist	Nhân viên Kế toán & Thanh toán	Handles invoicing, payments, and financial records.
10	Health & Safety Officer	Nhân viên An toàn Lao động	Ensures workplace safety compliance and risk management.
11	Lead Technician/Shop Foreman	Tổ trưởng Kỹ thuật/Quản đốc Xưởng	Supervises mechanics and ensures repair quality.
12	Automotive Diagnostic Specialist	Chuyên viên Chẩn đoán Ô tô	Uses technology to diagnose vehicle issues.
13	Master Mechanic	Kỹ thuật viên Bậc cao	Highly skilled technician specializing in complex repairs.

14	General Auto Technician	Thợ sửa chữa Ô tô	Performs standard maintenance and repair tasks.
15	Electrical Specialist	Chuyên viên Điện Ô tô	Focuses on vehicle electrical systems and repairs.
16	Body Shop Manager	Quản lý Xưởng Đồng sơn	Oversees vehicle body repairs and painting.
17	Alignment & Suspension Technician	Kỹ thuật viên Cân chỉnh Góc lái & Hệ thống treo	Specializes in wheel alignment and suspension systems.
18	Tire & Brake Specialist	Chuyên viên Lốp & Phanh	Handles tire replacements, rotations, and brake system repairs.
19	Air Conditioning & Heating Technician	Kỹ thuật viên Điều hòa & Sưởi Ô tô	Repairs and maintains vehicle HVAC systems.
20	Detailing & Cleaning Specialist	Chuyên viên Rửa xe & Chăm sóc Ô tô	Prepares vehicles for customers with cleaning and finishing services.
	Concept related to activities		
21	Scheduling & Appointments	Lên lịch và đặt hẹn	Make appointments with customer for car services
22	Service Estimation	Báo giá dịch vụ	An estimation of the cost and time required for a particular vehicle service or repair. It includes labor costs, parts costs, and any additional fees.
23	Customer Vehicle History Tracking	Theo dõi lịch sử sửa xe của khách	Tracking all previous maintenance and repairs performed on a customer's vehicle. It helps in diagnosing issues, planning future maintenance, and ensuring service consistency.
24	Customer Satisfaction Survey	Khảo sát sự hài lòng của khách hàng	Collect customer opinions about their service experience. It helps the garage improve service quality and address any customer concerns.

25	Parts Ordering	Đặt hàng phụ tùng	The process of purchasing spare parts from suppliers when inventory levels are low. This ensures that necessary components are available for vehicle repairs.
26	Inventory Control	Kiểm soát kho	Managing stock levels of spare parts and tools to ensure availability without overstocking. It includes tracking parts usage, replenishing supplies, and preventing stock shortages.
27	Vehicle Check	Kiểm tra tình trạng xe	A general inspection of the vehicle to assess its overall condition and detect potential issues before they become major problems.
28	Engine Performance Testing	Kiểm tra hiệu suất động cơ	A diagnostic test to check engine efficiency, fuel consumption, and emission levels. It helps detect performance issues and optimize the engine.
29	Brake System Inspection	Kiểm tra hệ thống phanh	A detailed check of brake components, including brake pads, rotors, calipers, and brake fluid, to ensure safe braking performance.
30	Fluid Level Check	Kiểm tra hệ thống dầu nhớt	Inspecting and refilling essential vehicle fluids, such as engine oil, coolant, brake fluid, transmission fluid, and power steering fluid, to maintain optimal vehicle function.
31	Exhaust System Inspection	Kiểm tra hệ thống xả	Checking the exhaust system for leaks, rust, and blockages to ensure proper emission control and prevent harmful gas leaks.
32	Headlights & Signal Lights Inspection	Kiểm tra đèn pha, đèn tín hiệu	Testing the functionality of headlights, brake lights, turn

			signals, and other exterior lights to ensure visibility and compliance with safety regulations.
33	Airbag System Inspection	Kiểm tra hệ thống túi khí	Checking the airbag system for any malfunctions or warning lights to ensure passenger safety in case of an accident.
34	Electrical System Check	Kiểm tra hệ thống điện	Diagnosing the vehicle’s electrical system, including the battery, alternator, starter motor, wiring, and fuses, to ensure proper operation.
35	Scheduled Maintenance	Bảo dưỡng định kì	Routine maintenance based on the manufacturer’s recommendations to keep the vehicle in good condition. It includes oil changes, tire rotations, brake inspections, and more
36	Oil & Filter Change	Thay dầu nhớt, lọc dầu	Replacing engine oil and oil filters to ensure smooth engine performance and prevent engine wear.
37	Coolant Flush & Refill	Xả và thay nước làm mát	Draining old coolant, cleaning the cooling system, and refilling it with new coolant to prevent overheating and engine damage.
38	Transmission Fluid Change	Thay dầu hộp số	Replacing the transmission fluid to ensure smooth gear shifting and prevent transmission wear.
39	Fuel System Cleaning	Vệ sinh hệ thống nhiên liệu	Removing carbon deposits and contaminants from the fuel injectors and intake system to improve fuel efficiency and engine performance.

40	Air Filter & Cabin Filter Replacement	Thay lọc gió động cơ, lọc gió điều hòa	Replacing the engine air filter to maintain air intake quality and the cabin air filter to ensure clean air circulation inside the vehicle.
41	Battery Charging & Replacement	Sạc và thay bình ắc quy	Testing, recharging, or replacing the car battery to ensure reliable starting power.
42	Spark Plug Replacement	Thay bugi đánh lửa	Replacing spark plugs to maintain efficient engine ignition and improve fuel efficiency.
43	Brake Pad & Rotor Replacement	Thay má phanh và đĩa phanh	Replacing worn-out brake pads and rotors to ensure effective braking and prevent damage to the braking system.
44	Engine Repair	Sửa động cơ	Fixing engine issues such as overheating, misfiring, or oil leaks to restore proper functionality.
45	Water Pump Replacement	Thay bơm nước làm mát	Replacing the water pump, which circulates coolant through the engine to prevent overheating.
46	Exhaust Leak Repair	Sửa chữa rò rỉ ống xả	Fixing leaks in the exhaust system that may cause increased emissions, noise, or decreased engine efficiency.
47	Tire Replacement	Thay lốp	Replacing worn-out or damaged tires to maintain traction, stability, and safety.
48	Flat Tire Repair	Vá lốp	Fixing punctured tires to restore air pressure and usability.
49	Glass & Windshield Replacement	Thay kính chắn gió	Replacing damaged windshields or car windows to maintain driver visibility and safety.
50	Dent Repair	Sửa móp xe	Fixing dents in the car’s body to restore its appearance.

51	Scratch & Paint Touch-Up	Sửa vết trầy và sơn dặm	Repairing minor scratches and applying touch-up paint to maintain the car's exterior aesthetics.
52	Full Body Repainting	Sơn lại toàn bộ xe	Repairing minor scratches and applying touch-up paint to maintain the car's exterior aesthetics.
53	Jump Start Service	Kích bình ắc quy	Providing assistance when a vehicle’s battery is dead by using jumper cables or a portable jump starter.
54	Towing Service	Dịch vụ kéo xe	Transporting a disabled or broken-down vehicle to a repair facility using a tow truck.
55	Waste Disposal	Xử lý chất thải	Properly disposing of used oil, fluids, tires, and other waste materials generated during vehicle servicing in an environmentally friendly manner.
56	Customer Profile Management	Quản lý thông tin khách hàng	Maintaining detailed records of each customer, including contact details, vehicle information, service history, and preferences.
57	Promotion program	Chương trình quảng cáo	Offering special discounts, loyalty rewards, and promotional deals to attract and retain customers.
58	Assigning Tasks to Technicians	Giao việc cho kỹ thuật viên	Distributing repair and maintenance tasks to available technicians based on their expertise and workload.
59	Workload & Performance Tracking	Thống kê hiệu suất làm việc	Monitoring the efficiency and productivity of technicians by tracking the number of completed jobs and time spent on each task.

60	Shift & Attendance Management	Quản lý ca, chấm công	Managing employee work schedules, tracking attendance, and ensuring proper shift coverage.
61	Financial Analysis	Báo cáo doanh thu	Evaluating garage revenue, costs, and profitability to make informed business decisions.
62	Customer Retention Analysis	Phân tích tỉ lệ khách hàng quay lại	Analyzing customer return rates, service preferences, and feedback to improve customer loyalty and retention strategies.
	Concept related to policy law/ rules		
63	Wear Proper Clothes	Mặc trang phục phù hợp	To prevent employees from wearing unacceptable attire
64	Use PPE (personal protective equipment)	Sử dụng đồ bảo hộ cá nhân	To ensure basic shop safety, personal protective equipment (PPE) is essential
65	Maintaining Fire Extinguishers	Duy hệ thống cảnh báo cháy	Make sure fire extinguishers are easily accessible and appropriate for all potential fire types
66	Be Careful with Batteries	Cẩn thận với pin	The battery should always be disconnected when employees work on electrical systems and near/around electrical wiring
67	Be Careful with Engines	Cẩn thận với động cơ	No one should place hands, tools, or other objects near the engine while it is running.
68	Be Careful Underneath Vehicles	Cẩn thận ở dưới gầm xe	Employees should not work underneath a vehicle unless it has been properly supported
69	Remove the Vehicle Key	Nhớ rút chìa khóa xe	The key should never be left in the ignition switch, as the key can draw an electrical charge from the battery.

70	Watch the Temperature	Chú ý nhiệt độ	Proper vehicle safety procedures include being aware of the vehicle’s temperature before beginning any work
71	Avoid Casual Eating	Ăn uống đúng nơi đúng chỗ	Contaminating food with chemicals used in the workshop or eating while working on cars or car parts are two significant risks of choosing to eat in a work area
72	Establish Smoke-Free Zones	Có khu vực hút thuốc riêng	Never allow smoke in or near repair bays or garages.
73	Avoid Clutter	Tránh sự lộn xộn	To maintain a safe auto garage, keep work areas clean and organized
74	Duty of care	Trách nhiệm chăm sóc	legal obligation of an organization or a person to avoid acts or omissions that are likely to cause harm to others
75	Risk mitigation	Giảm thiểu rủi ro	proactive efforts towards maintenance scheduling, record keeping and driver training
76	Driver safety		Creating a written vehicle maintenance policy helps to promote a culture of safety for your drivers
77	Third-Party Liability	Trách nhiệm của bên thứ ba	The insurer will pay on behalf of the insured all sums for which the insured is legally obligated
78	Accident Benefits	Quyền lợi khi gặp tai nạn	This coverage includes the same accident benefits as the OAP1 (Ontario Automobile Policy) and also adds income replacement, caregiver, non-earner benefit, medical benefit, rehabilitation,

			attendant care, death & funeral and indexation.
79	Direct Compensation – Property Damage (DCPD)	Bồi thường trực tiếp – Thiệt hại tài sản (DCPD)	Customers are compensated directly by their own insurer, and the Garage Auto policy becomes the second payee.
80	Collision or Upset	Va chạm hoặc lật đổ	Coverage for damage to a customer's vehicle when it is involved in a collision or tips over.
81	Specified Perils	Rủi ro cụ thể	Covers loss or damage caused by specific perils (events that can cause damage) such as fire, theft, vandalism, lightning, windstorm, etc., and provides legal defense coverage for the Garage Auto policyholder.
82	Comprehensive	Toàn diện	Comprehensive coverage broadens protection for customers' vehicles beyond specified perils to include careless acts of the Named Insured and employees.
	Concepts related to Finance & Billing		
83	Invoice	Hóa đơn	Customers need an invoice after repairs for payment.
84	Payment	Thanh toán	The process of customers paying for the service.
85	Revenue	Doanh thu	The total amount the garage earns from repair services.
86	Price	Giá	The listed price of each service and spare part.
87	Cost	Chi phí	The expenses incurred by the garage for parts and salaries.

88	Receipt	Phiếu thu	A receipt acknowledging the payment received from customers.
89	Expense Voucher	Phiếu chi	A form documenting expenses, such as purchasing parts.
90	Profit	Lợi nhuận	Revenue minus costs, indicating business efficiency.
91	Discount	Chiết khấu	Some customers or services may receive a discount.
92	Tax	Thuế	Taxes related to revenue and transactions.
93	Debt	Khoản nợ	When customers don’t pay immediately, their debt is recorded.
94	Salary	Lương	The garage must pay salaries, especially for technicians.
95	Bonus	Thưởng	Employees may receive additional bonuses based on performance.
96	Service Fee	Phí dịch vụ	Some garages may charge a fixed service fee.
97	Pricing Table	Bảng giá	A list of prices for services and spare parts.
98	Financial Report	Báo cáo tài chính	Monthly statistics on revenue, profit, and costs.
99	Cashbook	Sổ quỹ	A record of the garage's cash inflows and outflows.
100	Balance	Số dư	The remaining amount in the garage’s financial account.
101	Contract	Hợp đồng	Long-term agreements with customers or suppliers.
102	Transaction	Giao dịch	The history of payments, purchases, and sales in the system.
	Concepts related to Documentation & Records		
103	Customer	Khách hàng	The main entity using the garage's services.

104	Vehicle	Xe	Each customer may have multiple vehicles requiring repairs.
105	License Plate	Biển số	Each vehicle is uniquely identified by its license plate.
106	Car Model	Mẫu xe	Information about the car model helps technicians repair accurately.
107	Brand	Nhãn hiệu	The brand of the vehicle (Toyota, Ford, Honda, etc.).
108	Repair History	Lịch sử sửa chữa	Records all previous repair visits of the vehicle.
109	Service Log	Nhật ký dịch vụ	A detailed list of performed services.
110	Spare Part	Phụ tùng	Replacement components used during repairs.
111	Inventory	Kho hàng	The number of available spare parts in the garage.
112	Repair Ticket	Phiếu sửa chữa	A form documenting information about a vehicle needing repair.
113	Service Catalog	Danh mục dịch vụ	A list of service types offered by the garage.
114	Employee	Nhân viên	The list of staff working in the garage.
115	Technician	Kỹ thuật viên	A group of employees specialized in vehicle repairs.
116	Supplier	Nhà cung cấp	Companies that provide spare parts and components.
117	Statistics Report	Báo cáo thống kê	A summary of data on revenue and completed services.
118	Maintenance Contract	Hợp đồng bảo dưỡng	A periodic maintenance agreement between the customer and the garage.
119	Stock Receipt	Phiếu nhập kho	A record acknowledging the receipt of spare parts into inventory.

120	Warranty Book	Sổ bảo hành	Stores warranty information for vehicles and spare parts.
121	Parts List	Danh mục linh kiện	A list of replaceable spare parts.
122	Tracking Book	Sổ theo dõi	Records repair and maintenance activities.
	Concepts related to Technology & Software		
123	Customer Database	Cơ sở dữ liệu khách hàng	A system that stores customer information, including contact details and service history.
124	Vehicle Database	Cơ sở dữ liệu xe	Stores vehicle details such as make, model, year, VIN (Vehicle Identification Number), and service history.
125	Search & Filtering System	Hệ thống tìm kiếm	Enables searching customers, vehicles, services, and parts using keywords.
126	New Customer Registration	Đăng ký khách hàng mới	A feature allowing the addition of new customers if they do not exist in the database.
127	New Vehicle Registration	Đăng ký xe mới	A feature allowing the addition of a vehicle to an existing customer profile if not previously recorded.
128	Service Selection System	Hệ thống chọn dịch vụ	Allows selection of requested repair and maintenance services from a predefined list.
129	Spare Parts Selection System	Hệ thống chọn phụ tùng	Enables searching and selecting spare parts based on name or part number.
130	Temporary Invoice Generation	Tạo hóa đơn tạm thời	Creates a draft invoice that includes selected services and parts before finalizing the payment.
131	Price Calculation Module	Mô-đun tính giá	Automatically calculates the cost based on unit price and quantity of services/parts.

132	Tax & Discount System	Hệ thống thuế & chiết khấu	Applies taxes, service charges, and discounts to invoices as per company policy.
133	Final Invoice Generation	Tạo hóa đơn cuối cùng	Generates a finalized invoice for the customer after service completion
134	Invoice Printing & Digital Copy Storage	In hóa đơn & Lưu trữ bản sao kỹ thuật số	Supports printing invoices and storing them in digital format for future reference.
135	Payment Processing System	Hệ thống xử lý thanh toán	Handles different payment methods (cash, card, digital payment, etc.).
136	Revenue Tracking System	Hệ thống theo dõi doanh thu	Monitors income generated from services and spare parts
137	Service & Component Usage Statistics	Thống kê sử dụng dịch vụ & linh kiện	Tracks which services and spare parts are most frequently used.
138	Authentication System	Hệ thống xác thực	Requires users to log in with usernames and passwords.
Concepts related to Vehicles			
139	Convertible	Xe mui trần	a specially designed car with a hood that can be opened or closed at will.
140	Sedan	Xe sedan	Low-chassis car, elegant design, fuel-efficient, suitable for urban areas.
141	Pickup Truck	Xe bán tải	a type of car with a front passenger cabin and a rear cargo bed, designed to carry goods or to be transported over various terrains.
142	Crossover	Xe crossover	A hybrid vehicle between an SUV and a sedan/hatchback, providing the driving comfort of a sedan but with the high ground clearance and maneuverability of an SUV.

143	Hybrid	Xe lai điện	A vehicle that combines a gasoline engine and an electric motor, helping to save fuel and reduce emissions compared to traditional gasoline/diesel vehicles.
144	A-segment	Xe mini	The smallest car in the popular car segment, mainly small hatchbacks, suitable for urban areas because of their compact size, easy handling and affordable price.
145	B-segment	Xe ô tô gia đình cỡ nhỏ	A car in the small segment, larger than a class A car but smaller than a class C car. This is a popular car because of its reasonable price, fuel economy and suitability for urban conditions.
146	Coupe	Xe coupe	A sporty car, usually with only 2 doors (some models have 4 doors but still have a coupe design). Coupes have a sloping roofline that gives them a luxurious look and powerful driving performance.
147	Sport Cars	Xe thể thao	The vehicle is designed to deliver high performance, fast speed and flexible handling.
148	Wagon	Xe wagon	The car has a long tail design, creating more spacious space than a sedan, but is still lower than an SUV or crossover.
149	Electric Vehicle	Xe điện	A vehicle that uses an electric motor instead of an internal combustion engine (petrol/diesel). Energy is stored in a lithium-ion battery, which can be recharged

			using mains power or a fast charging station.
150	Luxury	Xe hạng sang	luxury cars, offering a driving experience and interior that surpasses that of mainstream cars. They feature sophisticated design, advanced technology, premium materials and powerful performance.
151	Hatchback	Xe hatchback	The car has a 5-door design, in which the rear door (hatch) opens upwards, integrating the rear windshield, providing easy access to the luggage compartment.
152	SUV	Xe thể thao đa dụng	SUV, high chassis, strong design, spacious interior, suitable for many types of terrain, from urban to off-road.
153	Minivan	Xe minivan	This car line has a spacious design, seating from 7 to 8 seats, large luggage compartment, often using sliding doors on both sides, suitable for families or passenger transport services.
154	Off-road vehicle	Xe chuyên dụng cho địa hình gồ ghề, bùn lầy, cát, sỏi.	A vehicle designed to travel over rough, uneven terrain such as dirt, sand, mud, or snow. These vehicles typically have a high chassis, four-wheel drive (4WD/AWD), and a sturdy suspension system.
155	MUV	Xe MUV	Practical car, can carry many people & goods, durable, fuel efficient.
	Concepts related to Inventory		

156	Engine Components	Phụ tùng động cơ	Parts that support engine function, including spark plugs, filters, belts, and cooling system components.
157	Transmission & Drivetrain	Hệ thống truyền động	Components that transfer power from the engine to the wheels, such as the clutch, gearbox, and drive shafts.
158	Suspension & Steering	Hệ thống treo & lái	Parts that provide stability and control, including shock absorbers, struts, and power steering pumps.
159	Braking System	Hệ thống phanh	Components that slow or stop the vehicle, such as brake pads, rotors, calipers, and brake fluid.
160	Electrical Components	Hệ thống điện	Parts responsible for the vehicle's electrical system, including the battery, alternator, fuses, and lighting.
161	Exhaust System	Hệ thống xả	Components that control emissions and reduce engine noise, such as catalytic converters and mufflers.
162	Body & Interior Parts	Phụ tùng thân vỏ & nội thất	Exterior and interior components like side mirrors, door handles, seat belts, and dashboard elements.
163	Tires & Wheels	Hệ thống lốp & bánh xe	Parts that ensure road contact and stability, including tires, alloy wheels, and wheel bearings.
164	Cooling & Heating System	Hệ thống làm mát & sưởi	Systems that regulate engine temperature and cabin comfort, such as radiators and AC compressors.
165	Fluids & Lubricants	Dầu & chất bôi trơn	Essential liquids for operation and maintenance, including engine oil, brake fluid, coolant, and transmission fluid.

166	Lifting & Handling Equipment	Thiết bị nâng hạ	Devices used to raise vehicles off the ground for repairs, such as hydraulic lifts, jacks, and jack stands
167	Diagnostic Tools	Thiết bị chẩn đoán	Instruments that help identify vehicle issues, including OBD-II scanners, multimeters, and compression testers
168	Hand Tools	Dụng cụ cầm tay	Basic tools like wrenches, screwdrivers, and pliers used for mechanical work.
169	Power Tools	Dụng cụ điện	Electrically or pneumatically powered tools such as impact wrenches and drills, used to speed up repairs.
170	Engine & Transmission Tools	Thiết bị sửa chữa động cơ & hộp số	Specialized tools for working on engines and transmissions, such as torque wrenches and engine hoists.
171	Brake & Suspension Tools	Thiết bị phanh & hệ thống treo	Equipment used to service braking and suspension systems, including brake bleeders and spring compressors
172	Tire & Wheel Equipment	Thiết bị lốp & bánh xe	Tools for changing and balancing tires, like tire changers and wheel balancers.
173	Welding & Fabrication Equipment	Thiết bị hàn & gia công	Machines used for metalwork, such as welders and grinders.
174	Fluid Maintenance Equipment	Thiết bị bảo dưỡng chất lỏng	Devices for handling vehicle fluids, like oil drain pans and coolant flush machines.
175	Air & Electrical Equipment	Thiết bị khí nén & điện	Tools like air compressors and battery chargers, essential for various repair tasks.

176	Safety & Storage Equipment	Thiết bị an toàn & lưu trữ	Items such as gloves, fire extinguishers, and tool chests for protection and organization.

II. Business model

1. By natural language

1.1. Objective of Scope

- This is a desktop-based application which will be used internally in a garage for car service management
- Type of application: winform (desktop-based application)
- Scope of provide: 1 garage car (it could execute on several desktops of the garage, but use the same database)
- User: only the garage car's employees could use, include: Garage's manager, system administrator, technical manager, cashier
- Provided function:
    + Receiving customer's car
    + Payment and returning cars to the customers
    + View statistical reports: reports on the revenue by services/ components, sales by month

1.2. User and the function that each user could use

- Manager:
    + View statistical reports: reports on the revenue by services/ components, sales by month
- System administrator:
    + Manage user account by their demand: add, edit, delete
- Technical manager:
    + Receiving the car bring by customer
- Cashier:
    + Do the payment process with the requirement of customer

1.3. The detail of how the functions work

a) Receiving a car

- Customer brings the car to the garage → The technical manager examines the car. Based on the car's condition, he/she identifies the required services and possible component replacements then opens the system and selects the receiving menu →An interface to find the customer by name appears with the editable text field: name, phone number, address, note and 2 button: search and

add → The technical manager ask the customer's name and insert to the text field then click the search button to find

→ If there is no result or no result with right information about the customer → The technical manager click the add new customer button → A new line including customer's information appears in the result list → technical manager selects that result → next

→ If there is result about the customer with all the correct information → the technical manager selects that result → next

- An interface to search the customer's car's information by plate number appears with editable text fields to insert plate number, name, brand, type and 2 button, one to search and other is to add → the technical manager asks the customer about the car's information → the customer provides the car's information → technical manager inserts these information into the text fields and click search button

  → If there is result which means the car had been brought to the garage before, technical manager select that line → next

  → If there is no result which means the car hasn't been brought to the garage before→ The technical manager click the adding new customer's car button → the line including that customer's car's information appears in the result list → the technical manager selects that result → next
- The interface to add services/ components to customer's order appears including one editable text field to enter the service/component name, a search button, a list of added service/component (which is blank at the start) that can modify the quantity → The technical tells the customer about the services or components that need to be done and its quantity and asks the customer to confirm

  → If the customer refuses to do that services to components repair → repeat the asking step

  → If the customer agrees → technical manager enter the name of service/ component to the text field and click search button→ A list of services/ components with the name containing the key word entered appears, each line contains detail information about that service/component: ID, Name, Unit price→ Technical manager clicks to select the correct service/ component → a

new line appears in the list of added services/components, including information about that service/component: No, ID, Name, Unit price, Quantity, Total with the quantity is 1 at first → Technician inserts the correct the quantity of that service/component, repeats the asking process until all the required services/ components are added to the customer's order and click next button → next

- The interface to select slot to repair that car appear with a list including these columns: No of slot, number of cars in slot's queue, arranged in ascending order
  → If all the slots have the number of cars in queue > 0, the technical manager informs the customer that his/her car will have to wait, then select the first slot with the smallest number of cars in the queue → next
  → If the number of car in queue of the first slot is 0, which means that slot if free → technical manager selects the first slot in the list → next
- The interface to assign service to technician appears with a dropdown menu to select which added service, a dropd own menu to select the free time slot, a button the search free technicians, a list of technicians which are free in the selected time slot and a button next → The technical manager chooses the added service, choose the free time slot then click search technician → The list of free technicians at that chosen time slot appears → Technical manager selects technicians to do that service, repeats the process until assign all the services to technicians then click next → next
- The system shows the temporary invoice including:
  + Invoice code
  + Customer information: Name, phone number, note
  + Car information: Name, number plate, type
  + Creator: Technical manager's name
  + Slot: slot number
  + Queue:
  + List of services/parts used: Each service or part is listed in a separate line with the following details:
    - No
    - ID: Unique identifier for the service or part.
    - Name: Name of the service or part.
    - Unit price: Price per unit of the service or part.
    - Quantity: Number of units used.

- Item total: Total cost for that service or part (unit price * quantity).
- Technician: ID of all the technicians who did that

+ Total amount: The sum of all services/parts used cost, displayed at the bottom of the invoice.

→ technical manager tells the total bill to the client and asks to confirm

→ If the customer wants to cancel → technical manager click the cancel button → the process ends exceptionally

→ If the customer confirms → technical manager clicks the confirm button → the system saves and prints out a temporary invoice to attach the car

b) Payments and Returning Cars to Customers

- Cashier selects the payment menu → The system displays the invoice search page, where the cashier can search for invoices using the invoice code →Invoice search page appears → The cashier is prompted to enter the invoice code from the temporary invoice provided by the customer.

  - If the invoice code is found → The system displays a detailed invoice interface with the following information:
    - Invoice code.
    - Customer information: Name, phone number, address, and any notes.
    - Car information: Name, number plate, model.
    - List of services/parts used/replaced: Each service or part is listed in a separate line with the following details:
      - ID: Unique identifier for the service or part.
      - Name: Name of the service or part.
      - Unit price: Price per unit of the service or part.
      - Quantity: Number of units used.
      - Total money: Total cost for that service or part (unit price * quantity).
    - Total amount: The sum of all services/parts used, displayed at the bottom of the invoice.

  → Cashier asks the customer to confirm the information → Next

  - If the invoice code is not found →The system displays an error message: "Invoice code not found." →The cashier asks the customer to double-check the invoice code on the temporary invoice.

- If the code is still incorrect:
→ The cashier contacts the technical manager
→ End exceptionally
- If the code is correct
→ Repeat the process

- Customer reviews the invoice:
- If the customer notices any discrepancies or wants to make changes → The customer informs the cashier about the required changes (e.g., adding, removing, or modifying services/parts) → The cashier makes the necessary adjustments in the system:
- To add a service/part:
→ The cashier clicks the "Add Service/Part" button.
→ The system displays a search interface for services/parts.
→ The cashier enters the name of the service/part and selects it from the list.
→ The cashier enters the quantity and confirms.
→ The system updates the invoice with the new service/part and recalculates the total amount.
- To remove a service/part:
→ The cashier selects the service/part to be removed from the list.
→ The cashier clicks the "Remove" button.
→ The system removes the service/part and recalculates the total amount.
- To modify a service/part:
→ The cashier selects the service/part to be modified.
→ The cashier changes the quantity or other details.
→ The system updates the invoice and recalculates the total amount.
→ The updated invoice is displayed again for the customer to review.

→ Cashier ask the customer to confirm

- If confirm

→ Next

- If not confirm

→ Repeat the asking process

- If the customer is satisfied with the invoice →Next.

- The cashier requests the customer to choose a payment method and then selects the chosen method in the system.

- If the customer chooses to pay by installment, the cashier selects this payment method in the system.

→ The system records the remaining balance for future payment.

→ The cashier informs the customer of the due amount and sets a payment deadline.

→ Next

- If the customer makes a full payment, the cashier selects this payment method in the system.

→ The cashier collects the payment and selects the "Paid" button in the system.

→ The system updates the invoice status to paid and records the transaction in the database.

→ Next

- System saves and prints the invoice:

→ The system automatically generates a final invoice.

→ The cashier prints two copies of the invoice:

- One copy for the customer.
- One copy for the garage's records.

→ The customer and the cashier sign both copies of the invoice.

→ The customer receives their copy of the signed invoice

→ Next

- Customer receives the car:

→ The cashier informs the customer that the payment process is complete.

→ The cashier hands over the car keys to the customer.

→ The customer leaves the garage with their repaired car.

c) Statistics of services/ components by revenue

- The manager logs into the system using their username and password → The manager's UI appears, it has the following options: manage the garage's information, manage garage's inventory, manage customer's membership, view statistics reports → The manager chooses the view statistic menu → The interface of the statistical reports appear with two sections:
    + The first section is the object of the report, including: service/component, revenue.
    + The second section is the criteria of the report including: by revenue, by month (if the object is revenue), by quantity

  → The manager selects the object is service/component, the criteria is by revenue → The report UI appears with two input dates: the start and the end of the duration to count the report → The manager enters these dates and clicks on the view button → The result pages show a list of services/ components, each line corresponding to: code, service/component name, total quantity provided, total revenue → The manager can click the filter button to view service/component which has the highest revenue to the lowest one and vice versa → The manager can click a line of service/component to see the detail history of time that service/component is used. → An interface appears including the information and the used history of that chosen service/component, each row including detail information about one transaction from the closest to the furthest one: date, customer name, vehicle name, translation name service/equipment, unit price, quantity, payment → the manager can click the close button to return to the interface before.

d) Monthly sales statics

- The manager logs into the system using their username and password → The manager's UI appears, it has the following options: manage the garage's information, manage garage's inventory, manage customer's membership, view statistics reports → The manager chooses the view statistic menu → The interface of the statistical reports appear with two sections:

+ The first section is the object of the report, including: service/component, revenues, client.
+ The second section is the criteria of the report including: by revenue, by month (if the object is revenue), by quantity

→ The manager selects the object is revenue and the criteria is by month → The results page appears including a list of the last 12 months (in chronological order time), each row corresponding to a month with: month name, total revenue, number of bills → The manager can click on 1 line to see list of bills of that month, each line corresponds to: date, customer name, vehicle name, total number of services/components repair that had been done, total amount (order from the newest bill to the oldest one) → Manager can click on 1 line to see the detailed bill → an invoice interface appears as when paying for the customer: date, customer's name, customer's telephone, each used service/component includes its code, name, unit price, quantity, total price, invoice code → the manager can click the close button to return to the interface before.

1.4. Information about related objects

- Service: code, name, price, description about required technicians
- Component: code, name, price, remaining quantity, description
- Client: name, phone number, address
- Customer car: name, plate number, brand, type
- Technician:code, name, phone number
- Slot: no, number of car in queue
- Employee/user account: full name, username, password, role
- Suppliers: code, name, address, email, phone, description
- Invoice:
    + Invoice code
    + Customer information: Name, phone number
    + Car information: Name, number plate
    + Creator: Technical manager's name
    + Technician: Technician's name
    + Slot: slot number
    + Queue:
    + List of services/parts used/replaced: Each service or part is listed in a separate line with the following details:
        - No

        - ID: Unique identifier for the service or part.
        - Name: Name of the service or part.
        - Unit price: Price per unit of the service or part.
        - Quantity: Number of units used.
        - Item total: Total cost for that service or part (unit price * quantity).
    + Total amount: The sum of all services/parts used cost, displayed at the bottom of the invoice.
- Components Import Invoice: invoice code, supplier's name, imported time, the components on a table, each line corresponds to a component: code, name, unit price, quantity, and amount, payment time, total amount
- Salary Invoice: code, employee's information, month, basic salary, revenue share, total salary
- Statistics of service/component by revenue: code, service/component name, total quantity provided, total revenue
- Statistics of revenue by month: month name, total revenue, number of bills

1.5. Relationship among objects

- A customer can own more than one car. A car belongs to one customer.
- A car can have many invoices for many visits. An invoice can includes more than one car
- A technician can do more than one service for a car at a time. One service for a car visit can require more than one technician to do
- A spare part can be imported many times from different suppliers. One supplier can provide more than one spare part at a time
- A temporary invoice is created by one technical manager, one technical manager can create many invoices.
- A car visit can require more than one service/component repair. One service or component can be done for many cars
- An invoice is associated with one cashier. A cashier could do payment process for many clients
- A customer can have more than one invoice. An invoice is for one customer
- At a time, the maximum number of cars being repaired is equal to the number of technicians or the number of slots of the garage (whichever is smaller, that counts).
- A manager can view statistics for services, parts, and revenue.
- Statistics are based on revenue, times and client visits.

2. Business model by UML

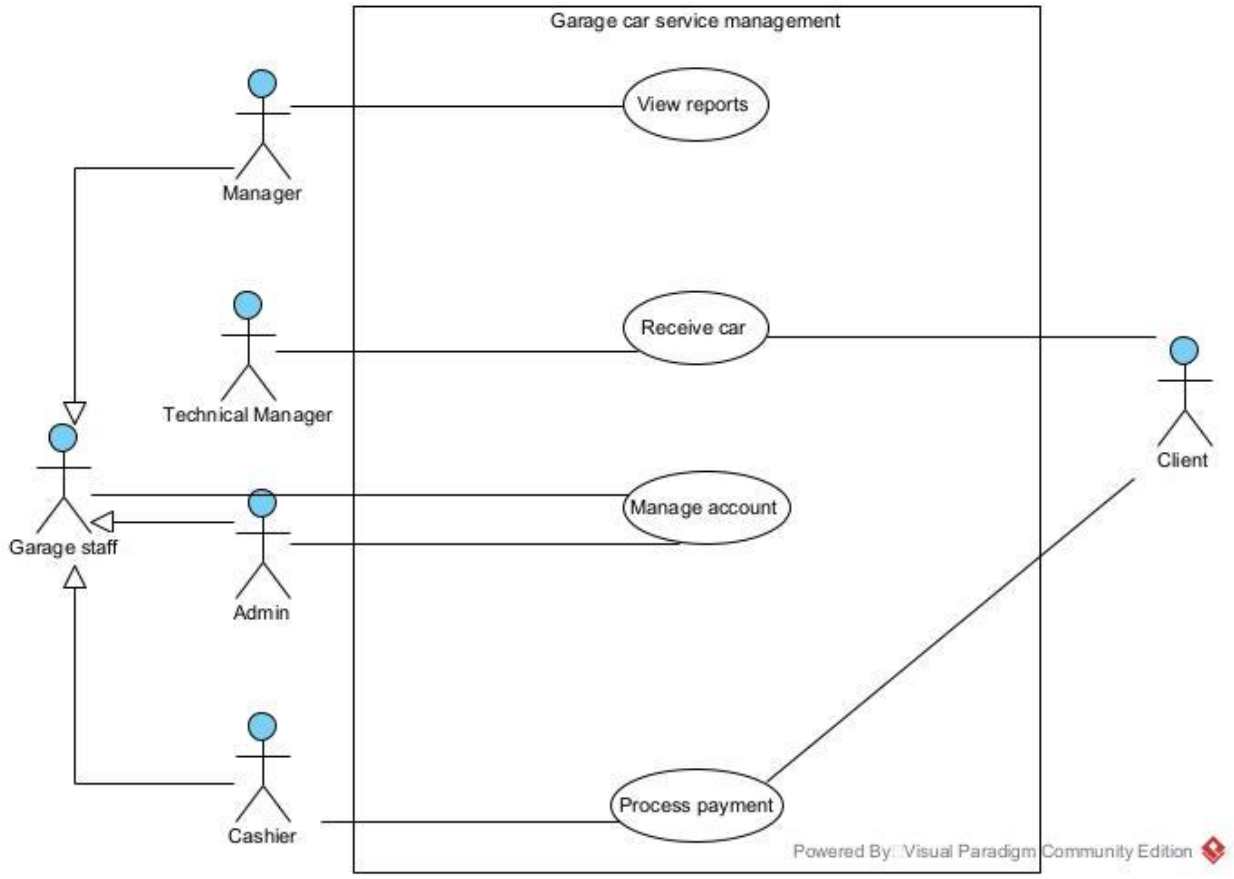
2.1. General UC

Step 1: actors

- Direct actors: manager, admin, technical manager, cashier. They are the same type as employees of the garage (Employee - abstract actor).
- Indirect actor: customer. It could kick off some functions: receiving cars, payments and returning cars.

Step 2 and 3: use case

- Manager could use the following functions:
  + View statistical reports: reports on the revenue by services/ components, sales by month → Viewing reports.
- Admin could use the following functions:
  + Manage user account by their demand → manage account
- Technical manager could use the following functions:
  + Receiving the car bring by customer → Receive car
- Cashier could use the following functions:
  + Do the payment process with the requirement of client → Process payment



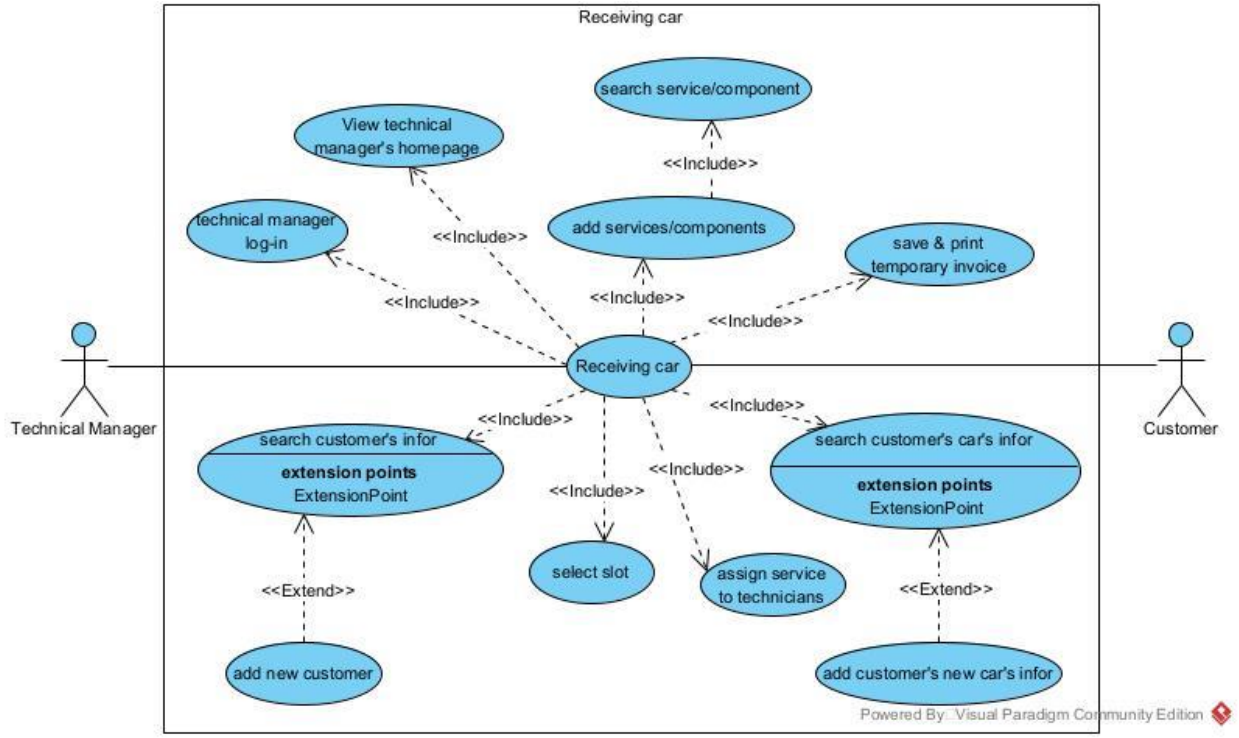
The use cases are described as below:

- View reports: It enables the manager to view statistical report about services/components by revenue and monthly sale
- Receive car: This enables the technical manager to create temporary invoice for customer when they bring car to the garage
- Manage account: this use case enables the administrator to manage the user account by the demand of related users.
- Process payment: This use case enables the cashier to process payments based on client requirements.

2.2. Use case Receiving car

Step 1: actors

- Direct actor: Technical manager
- Indirect actor: customer



Use case description:

- Technical manager log-in: for technical manager
- View technical manager homepage: this enables the technical manager to view the homepage after log into the system
- Add service/component: this enables the user to add new services/components to temporary invoice
- Search service/component: this enables the user to search service/component by its name
- Process customer information:

    + Search customer's information by name
    + Add new customer
- Process the customer car's information:
    + Search customer car's information by plate number
    + Add customer new car's information
- Assign service to technicians: this enables the user to select technicians to do each service
- Select slot: this enables the user to select the slot to put the car in
- Save and print temporary invoice: this enables user to save all the invoice information to the system and print out to attach the car

B. Analysis

I. Scenario

<table>
<tr><td>Scenario</td><td>Receiving car</td></tr>
<tr><td>Actor(s)</td><td>Technical manager, client</td></tr>
<tr><td>Pre-condition</td><td>Technical manager has an account to access the system</td></tr>
<tr><td>Post-condition</td><td>The car has been received successfully</td></tr>
<tr><td>Main events</td><td>1. A technical manager A login with username = A,
password = 123 into the system to receive a car bring by customer
2. The system displays the technical manager main UI which includes an option to select receiving car function
3. The technical manager selects the receiving car function
4. The system shows search bar to search customer information, there are:
- Input: full name
- Button: search, add new, next
5. The technical manager asks the customer's name
6. The customer answers his name is B
7. The technical manager enters customer's name = B to the search bar and click search button
8. A list of all clients whose name contains the keyword "B" is listed as follow:
<table><tr><td>No</td><td>Name</td><td>Phone number</td><td>Address</td></tr><tr><td>1</td><td>Nguyen Van A</td><td>0123456789</td><td>Ha Dong</td></tr></table></td></tr>
</table>

2	Nguyen Van Duc A	0112233445	Ha Dong
3	Nguyen Van A	1223344556	Hoai Duc

9. Technical manager asks customer if the first row information is correct
10. Customer confirms
11. Technical manager selects the first row and clicks next
12. The interface of searching customer's car's information appears with:
    - Input: Plate number
    - Button: search, add, next
13. The technical manager asks customer about plate number of the car he/she has brought
14. The customer answer the plate number of the car is 30G-636.11
15. The technical manager inserts the plate number 30G-636.11 into the text field and click search
16. The result with a line including information about the car with plate number = 30G-636.11 brought by customer before appears as follow:

No	Plate number	Name	Brand	Type
1	30G-636.11	Mazda 6	Mazda	7 seats

17. The technical manager selects the result and clicks next
18. The interface to add services/components to customer's order appears including
    - Input: Name of service/component, quantity
    - Button: search, add, next
    - List of added services/components, each line include: ID, Name, Unit Price, Quantity, Item total
19. The technical manager tell the customer the component that the car needs to change 2 tires which takes the service tire replacement and 2 new tires and ask the customer confirm
20. The customer agrees
21. The technical manager inserts the key word "tire" into the text field and click the search button
22. The list of service/component which name includes the key word "tire" appears as follow

No	ID	Name	Unit price
1	S456	Flat tire repair	100.000đ
2	S467	Tire replacement	100.000đ
3	C555	Tire	200.000đ

23. The technical manager selects the second row
24. A new line of tire replacement service appears in the list of added services/components as follow

No	ID	Name	Unit Price	Quantity	Item total
1	S467	Tire replacement	100.000đ	1	200.000đ

25. Technical manager modifies the quantity to 2, repeats the asking process after adding all the required services/components then clicks next
26. The interface to assign added services to technicians appears as below:

Service: [Select ▼]
[S467 - Tire replacement]
Time slot: [Select ▼]
[13:30-15:00]

No	ID	Name	Select
1	TC005	Nguyen Van A	☐
2	TC066	Pham Thi C	☐

- Button: next
-

27. Technical manager clicks the dropdown Select
28. A list of technicians that are not repairing any car appears
29. Technical manager selects 3 technicians T006, T011, T012 for the first service, repeats the process and select 2 technicians T006, T011 for the second one then clicks next
30. The interface to select slot to repair the car appears with a list of all the slots and its number of car in queue arranged in ascending order

Slots:

<table>
<tr>
<td></td>
<td>
<table>
<tr><td>No</td><td>Slot</td><td>Number of car in queue</td></tr>
<tr><td>1</td><td>S01</td><td>0</td></tr>
<tr><td>2</td><td>S07</td><td>1</td></tr>
</table>
- Button: Next
31. Technical manager selects the slot S01 and clicks next
32. The system shows the temporary invoice including:
- Invoice code: 789
- TimeIn: 13:15 PM 11/01/2022
- Customer name: B
- Customer phone number: 0123456789
- Customer address: Ha Dong
- Car ‘s name: Mazda 6
- Car’s plate number: 30G-636.11
- Receiving staff: A
- Technician: T006, T011, T012
- Slot: S01
- Queue: 1
<table>
<tr><td>No</td><td>ID</td><td>Name</td><td>Unit price</td><td>Quantity</td><td>Item Total</td></tr>
<tr><td>1</td><td>SC467</td><td>Tire replacement</td><td>100.000đ</td><td>2</td><td>200.000đ</td></tr>
<tr><td>2</td><td>C555</td><td>Tire</td><td>150.000đ</td><td>2</td><td>300.000đ</td></tr>
<tr><td>3</td><td>SC123</td><td>Washing</td><td>100.000đ</td><td>1</td><td>100.000đ</td></tr>
</table>
- Total: 600.000đ
And 2 button: Cancel, Confirm
33. The technical manager tells the total bill to the client and ask to confirm
34. The customer confirms the invoice
35. The technical manager click the Save & Print button
36. The system displays a successful message and print out the temporary invoice
</td>
</tr>
<tr>
<td>Exception</td>
<td>2. The system alerts that username or password is incorrect
2.1. The technical manager click the OK button</td>
</tr>
</table>

2.2. The message closes and return to login UI with username = A, password = 123, button log in
2.3. The technical manager modifies the username and/or password and click the login button
2.4. The system displays the technical manager main UI which includes an option to select to receive car function
8. There is no customer found or the client B is not in the system yet
8.1. The technical manager clicks the Add button to add new customer to the system
8.2. The add new customer interface appears with editable text fields: Name, Phone number, Address
8.3. The technical manager asks customer's name, phone number, address
8.4. Customer provides information
8.5. Technical manager inserts information to text fields and click add
16. There is no result which means that customer has never brought the car with plate number 30G-636.11 to the garage before
16.1. The technical manager clicks the Add button to add new customer's car to the system
16.2. The add new customer's car interface appears with editable text fields: plate number, name, brand, type
16.3. The technical manager asks customer's cas's plate number, name, brand, type
16.4. Customer provides information
16.3. Technical manager inserts information to text fields and clicks add
28. The interface to assign service to technicians appears as below:
Services for car: 30G-636.11
Service: [Select ▼]
[S467 - Tire replacement]
Time slot: [Select ▼]
[13:30-15:00]

No	ID	Name	Select
1	TC005	Nguyen Van A	☐

2	TC066	Pham Thi C	

- Button: next

28.1 Technical manager clicks the dropdown menu
28.2 No technician is shown which means no technicians can repair the car right now
28.3 Technical manager informs customer that his/her car will have to wait because there are no free technicians
28.4. Customer agrees to wait
28.5. Technical manager clicks next
30. The interface to select slot to repair the car appears with a list of all the slots and its number of car in queue arranged in ascending order
Slots:

No	Slot	Number of car in queue
1	S01	2
2	S07	5

- Button: Next

30.1. Technical manager informs customer that there are no free slot right now and the car will have to wait for at least 2 turns
30.2. Customer agrees to wait
30.3. Technical manager selects the first row which has only 2 cars in queue - the smallest
30.4. Number of car in queue of slot S01 increases to 3
30.5. Technical manager clicks next
34. Customer changes his mind and doesn't want to repair the car anymore
34.1. Technical manager clicks the cancel button

II. Entity class extraction

Step 1: Describe the Receiving car module

The system assists the technical manager in receiving cars brought in by customers and creating a temporary invoice linked to each car. The technical manager can search for a customer by name or add a new customer if they are not already in the system. Next, the technical manager can search for the received car by its license plate number. If the car has previously been brought, its information will be available; otherwise, the technical manager can add a new car for the customer. The technical manager can then add the required services or components by searching for their names and selecting the correct items. After adding each item, they can specify the quantity before proceeding to

the next step. Once all required services and components have been added, the technical manager can click Next to assign services to technicians. The system will display a list of added services and description of required number of technicians, a dropdown menu to select technicians according to required number. After assigning all the services, the technical manager clicks next to select the free slot. An interface including a list of all slots and its number of cars in the queue arranged in ascending order appears. Technical manager selects the first row and clicks next. The interface display detail temporary invoice appears. Technical manager can click confirm to save the invoice or cancel the unsave.

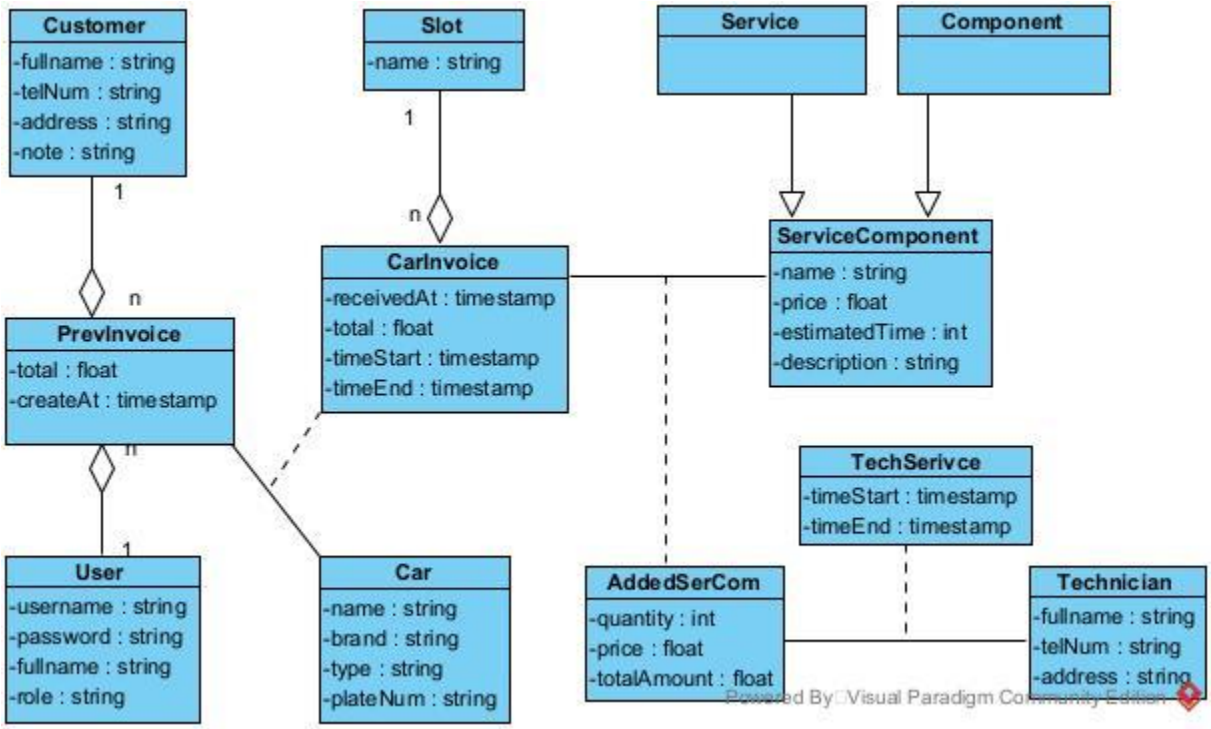
Step 2+3: Noun extraction and evaluate

- System: an abstract noun → reject
- Technical manager → class User: username, password, fullname, role
- Car → class Car: plate number, name, brand, type
- Customer → class Customer: name, phone number, address
- Customer's name: attribute → reject
- License plate number: attribute → reject
- Car's information: attribute → reject
- Service → class Service: ID, name, price, description, estimatedTime
- Component → class Component: ID, name, price, remain quantity, description
- Temporary invoice → class PrevInvoice: ID, createdAt, total
- Technician: fullname, telNum, address
- Slot: no

Step 4+5: Quantity and relationship among classes

- A user can create many temporary invoices, a temporary invoice is created by one user, so the relationship between User - PrevInvoice is 1-n
- A car can have many temporary invoices for many visits, an invoice can be for more than one car so Car - PrevInvoice is n-n. We create an entity CarInvoice between them
- Service and Component have many attributes in common and each time searching will search for both service and component so we create an entity ServiceComponent. Service and Component will inherit from this entity
- A car at one visit can require many service or repairing parts, a service/component can be done for many cars so CarInvoice - ServiceComponent is n-n. We create a class AddedSerCom between these two entities

- An added service can require many technicians to do, a technician can do many services so Technician-AddedSerCom is n-n. We create a class TechService between these two entities
- A car is put in one slot, a slot can be used for many cars at different times so CarInvoice -Slot is n-1



III. Class diagram

Analysis receiving car module:

- Enter the system → Log-in interface appears → Need a class LoginView
  - Input for username → inUsername
  - Input for password → inPassword
  - A submit to login → subLogin
- Enter the username + password → the system must check if the login is correct → need a method check login:
  - Name: checkLogin()
  - Input: username, password (class User)
  - Output: boolean
  - Owner class: User
- Once the login is successful → the technical manager's homepage appears → need a class TechMngHomeview which has at least:
  - An option to choose Receiving car → subReceiveCar

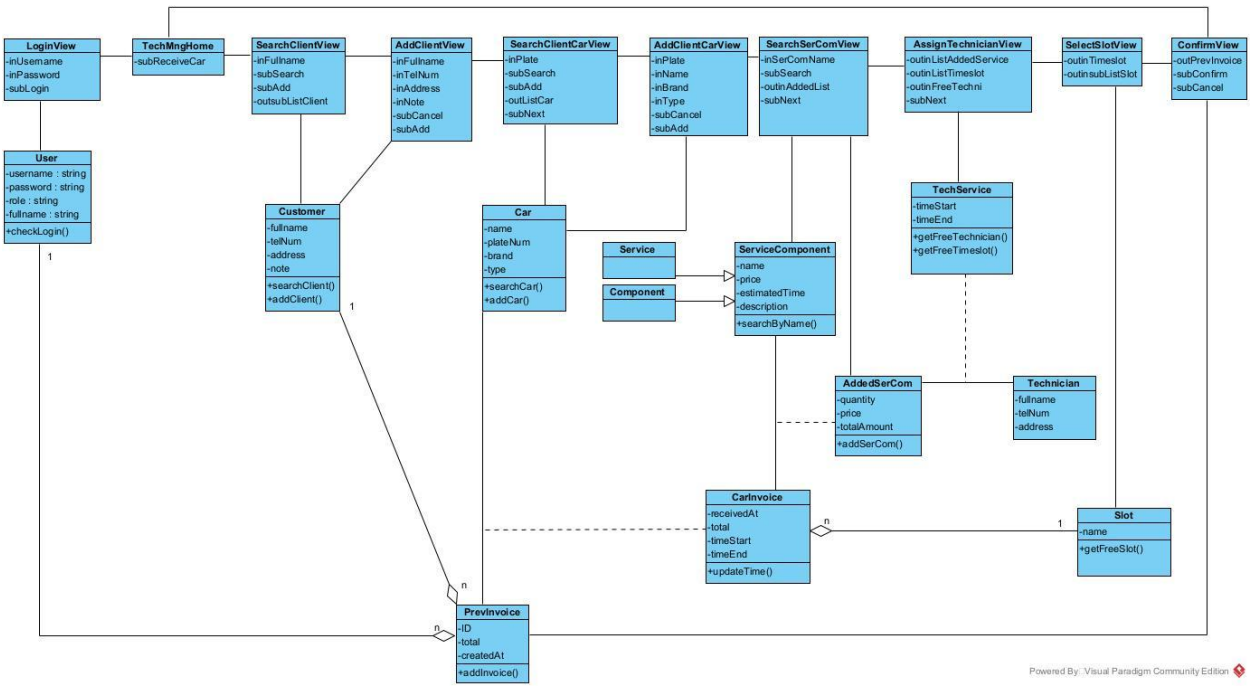
- Choose the option to receive car → The search client's information UI appears → need a class SearchClientView
    + Input for fullname → inFullname
    + A submit to search → subSearch
    + A submit to add → subAdd
    + A list of customers whose name includes the inserted name, can click to choose the correct one → outsubListClient
- Enter customer's information to search → The system must search all customers whose name contains the entered fullname → need a method to search client by name
    + Name: searchClient()
    + Input: key words
    + Output: a list of customers
    + Owner class: Customer
- The results are returned to (and displayed on) the SearchClientView.
- The technical manager chooses the correct customer, if the customer is new, click button add, the AddClientView interface appears including :
    + inFullname
    + inTelNum
    + inAddress
    + subAdd
    + subCancel

  And a method to add new customer:
    + Name: addClient()
    + Input: an object of customer
    + Output: a new customer
    + Owner class: Customer
- The search customer's car's information appears → need a class SearchClientCar
    + inPlate
    + subSearch
    + subAdd
    + A list of car whose plate includes the inserted string, can click to choose the correct one → outsubListCar

- Enter customer's car's plate number to search → The system must search all car whose plate contains the entered string → need a method to search car by plate number
    + Name: searchCar()
    + Input: a string of plate number
    + Output: a list of car
    + Owner class: Car
- The results are returned to (and displayed on) the SearchClientCar.
- The technical manager chooses the correct car, if the car is new, click button add and AddNewCar interface appears, including:
    + outinPlate
    + outinName
    + outinBrand
    + outinType
    + subAdd
    + subCancel

  And a method to add new car:
    + Name: addCar()
    + Input: an object of car
    + Output: a new car
    + Owner class: Car
- The search service/component appears → need a class SearchSerCom
    + inSerComName
    + subSearch
    + A list of services/components whose name includes the inserted string, can click to choose the correct one → outListSerCom
    + A list of added services/components that can modify the quantity → outinAddedList
    + A button to click next after adding all service/component → subNext
- Enter service/component's name to search → The system has to search all service/component whose name contains the entered words → need a method to search service/component by name
    + Name: searchByName()
    + Input: key words
    + Output: a list of service/component

    + Owner class: ServiceComponent
- The results are returned to (and displayed on) the SearchSerCom.
- After adding the service/component, technical manager modifies the quantity of that service/component → need a method to update the quantity:
    + Name: updateQuantity()
    + Input: an added service/component and a number
    + Output: a added service/component with updated quantity
    + Owner class: AddedSerCom
- The interface to assign services to technicians appears → need a class AssignTechnician including:
    + A list of added services with a column to select technicians → outin
- After adding all service/component and click next, the confirm interface is appeared -> need a class ConfirmView
    + display all information about the temporary invoice → outPrevInvoice
    + a confirm button → subConfirm.
    + a cancel button → subCancel
- The technical manager chooses to confirm after having the aggregation from the customer → The system has to save the invoice into the database → need a method:
    + name: addInvoice()
    + input: an object of PrevInvoice
    + output: none or boolean
    + Owner class: PrevInvoice
- After saving to the DB, the system returns to the TechMngHome.



IV. Sequence diagram

1. Scenario version 2

1) The customer brings car to fix and technical receives
2) The technical manager logins from the LoginView
3) The class LoginView calls the class User to process.
4) The class User executes the function checkLogin()
5) The class User returns result to the LoginView
6) The class LoginView calls the class TechMngHome.
7) The TechMngHome shows to the technical manager
8) The technical manager selects the receiving car function from the homepage
9) The TechMngHome calls SearchClientView
10) The SearchClientView shows to the technical manager
11) The technical manager asks the customer's name
12) Customer answers his/her name
13) Technical Manager enter customer's name and click search button of SearchClientView
14) The SearchClientView calls Customer class
15) The Customer class executes function searchClient()
16) The Customer class returns results to SearchClientView
17) SearchClientView shows results to technical manager
18) The technical manager selects the right customer on SearchClientView

19) SearchClientView calls SearchClientCar
20) The SearchClientCar shows to the technical manager
21) The technical manager asks the customer's car's plate number
22) Customer answers his/her car's plate number
23) Technical Manager enter customer's car's plate number and click search button of SearchClientCar
24) The SearchClientCar calls CustomerCar class
25) The CustomerCar class executes function searchCar()
26) The CustomerCar class returns results to SearchClientCar
27) SearchClientCar shows results to technical manager
28) The technical manager selects the right car
29) The SearchClientCar calls ReceivedCar class
30) The ReceivedCar class executes function addReceivedCar()
31) The ReceivedCar class returns to SearchClientCar
32) SearchClientCar calls SearchSerCom
33) The SearchSerCom shows to the technical manager
34) The technical manager tells the customer about the required service/component
35) Customer agrees
36) Technical Manager enter service/component's key word and click search button of SearchSerCom
37) The SearchSerCom calls ServiceComponent class
38) The ServiceComponent class executes function searchByName()
39) The ServiceComponent class returns results to SearchSerCom
40) SearchSerCom shows results to technical manager
41) The technical manager clicks on the right service/component, enters quantity and click add
42) The SearchSerCom calls AddedSerCom class
43) The AddedSerCom class executes function updateQuantity()
44) The AddedSerCom class returns to SearchSerCom
45) SearchSerCom calls ConfirmView
46) The class ConfirmView displays all temporary invoice's information to the technical manager
47) The technical manager repeats these information to the client and requires the client to confirm.
48) The client confirms them.
49) The technical manager clicks the confirm button.
50) The class ConfirmView call the class PrevInvoice to process

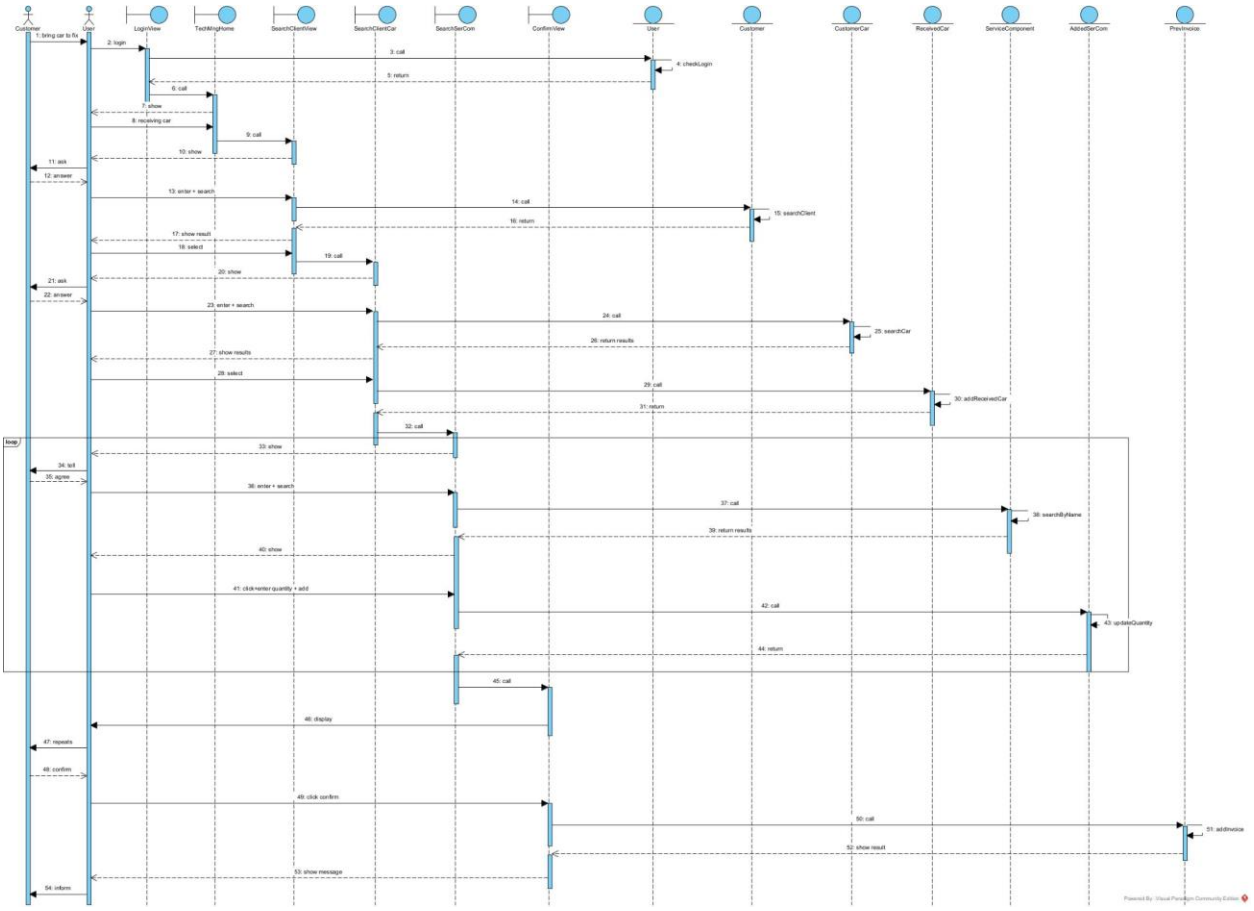
51)The class PrevInvoice calls the method addInvoice().

52) The PrevInvoice shows result to ConfirmView

53) The ConfirmView shows successful message to technical manager

54) The technical manager inform the success to customer

2. Sequence diagram



C. Design

I. Entity class design

Step 1: Add ID attribute for classes: Customer, Car, User, PrevInvoice, CarInvoice, Slot, SerCom, AddedSerCom, TechService, Technician

Step 2: Add type for attribute for all classes

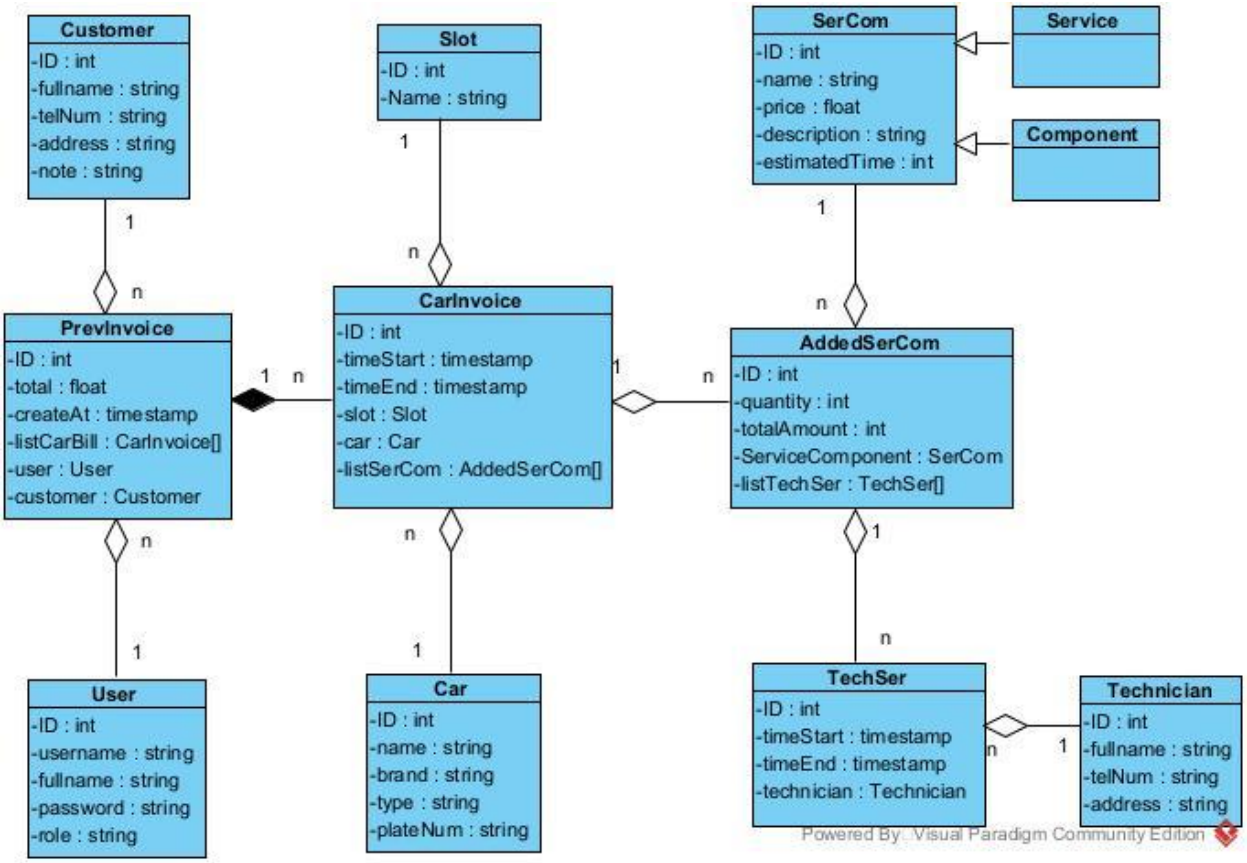
Step 3: Convert association relationships to correspond aggregation/composition relationships:

- Car + PrevInvoice → CarInvoice: Car is a component of CarInvoice, CarInvoice is a component of PrevInvoice

- CarInvoice + ServiceComponent → AddedSerCom: AddedSerCom is a component of CarInvoice, ServiceComponent is a component of AddedSerCom
- AddedSerCom + Technician → TechService: Technician is a component of TechService, TechService is a component of AddedSerCom

Step 4: Add object attributes:

- User is a component of PrevInvoice, of type 1-n → PrevInvoice has a User
- CarInvoice is a component of PrevInvoice, of type n-1 → PrevInvoice has a list of CarInvoice
- Car is a component of CarInvoice, of type 1-n → CarInvoice has a Car
- Customer is a component of PrevInvoice, of type 1-n → PrevInvoice has a Customer
- Slot is a component of CarInvoice, of type 1-n → CarInvoice has a Slot
- AddedSerCom is a component of CarInvoice, of type n-1 → CarInvoice has a list of AddedSerCom
- ServiceComponent is a component of AddedSerCom, of type 1-n → AddedSerCom has a ServiceComponent
- Technician is a component of TechService, of type 1-n → TechService has a Technician
- TechService is a component of AddedSerCom, of type n-1 → AddedSerCom has a list of TechService



II. Database design

Step 1: Create a table for each entity

- Customer → tblCustomer
- PrevInvoice → tbl PrevInvoice
- User → tblUser
- Slot → tblSlot
- CarInvoice → tblCarInvoice
- Car → tblCar
- ServiceComponent → tblSerCom
- AddedSerCom → tblAddedSerCom
- TechService → tblTechService
- Technician → tblTechnician
- Service → tblService
- Component → tblComponent

Step 2: For each attribute of entities which is not an object, convert to attribute of correspond table

- tblCustomer: ID, fullname, telNum, address, note
- tblPrevInvoice: ID, createdAt
- tblUser: ID, username, password, fullname, role
- tblSlot: ID, name
- tblCarInvoice: ID, timeStart, timeEnd
- tblCar: ID, name, brand, type, plateNumber
- tblSerCom: ID, name, price, description, estimatedTime
- tblAddedSerCom: ID, quantity
- tblTechService: ID, timeStart, timeEnd
- tblTechnician: ID, fullname, telNum, address
- tblService: ID
- tblComponent: ID

Step 3: Convert the cardinality relationships between entity classes into cardinality relationships between database tables

- 1 tblCustomer – n tblPrevInvoice
- 1 tblUser – n tblPrevInvoice
- 1 tblSlot – n tblCarInvoice
- 1 tblPrevInvoice – n tblCarInvoice
- 1 tblCar – n tblCarInvoice
- 1 tblSerCom – n tblAddedSerCom
- 1 tblAddedSerCom – n tblTechService
- 1 tblTechnician – n tblTechService
- 1 tblSerCom – 1 tblService
- 1 tblSerCom – 1 tblComponent

Step 4: Config the key columns for tables

- For each table has ID attribute → set as primary key
- Foreign key:
  + 1 tblCustomer – n tblPrevInvoice → tblPrevInvoice has foreign key customerID
  + 1 tblUser – n tblPrevInvoice → tblPrevInvoice has foreign key userID
  + 1 tblSlot – n tblCarInvoice → tblCarInvoice has foreign key slotID
  + 1 tblPrevInvoice – n tblCarInvoice → tblCarInvoice has foreign key invoiceID
  + 1 tblCar – n tblCarInvoice → tblCarInvoice has foreign key carID
  + 1 tblSerCom – n tblAddedSerCom → tblAddedSerCom has foreign key sercomID

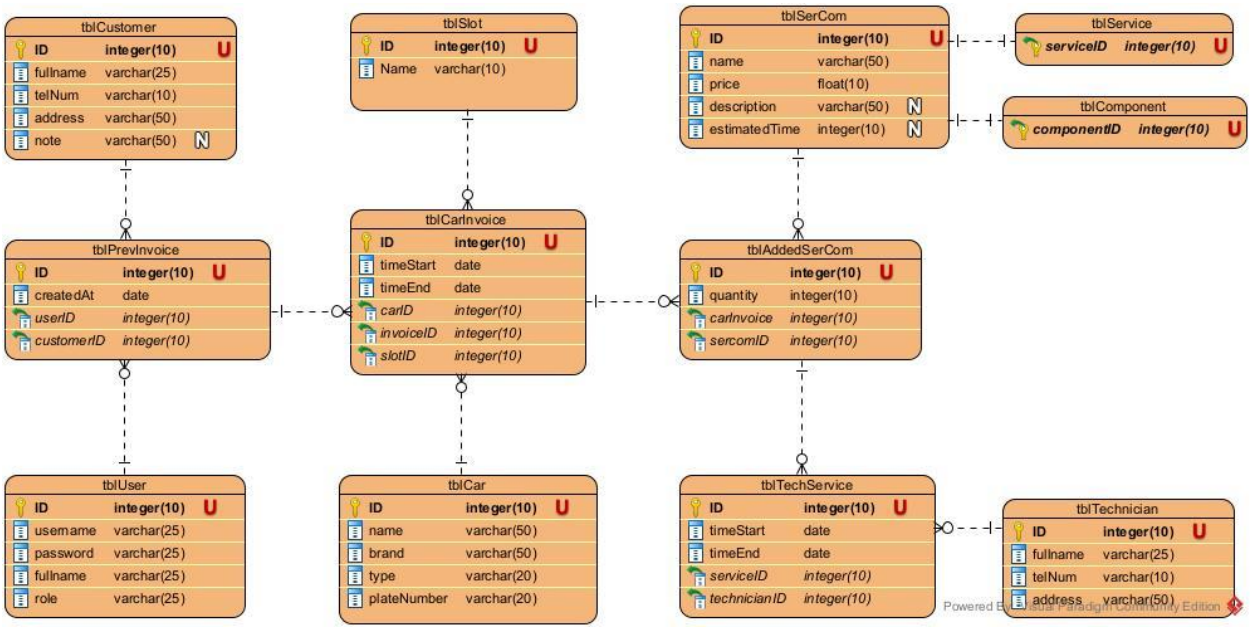
+ 1 tblAddedSerCom – n tblTechService → tblTechService has foreign key serviceID

+ 1 tblTechnician – n tblTechService → tblTechService has foreign key technicianID

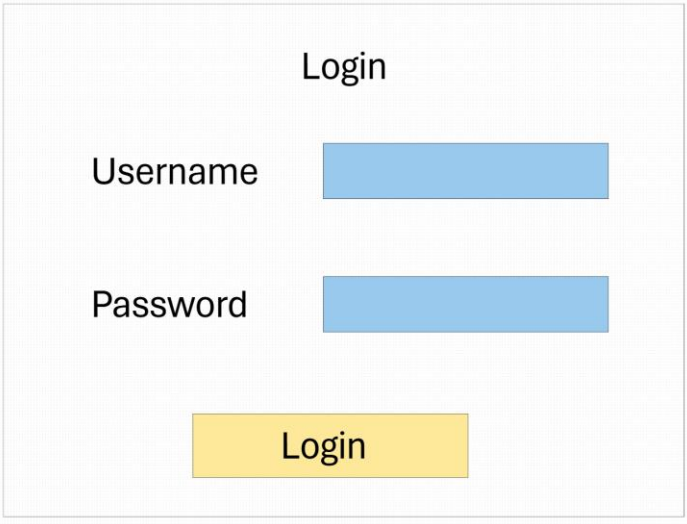
+ 1 tblSerCom – 1 tblService → tblService has foreign key serviceID

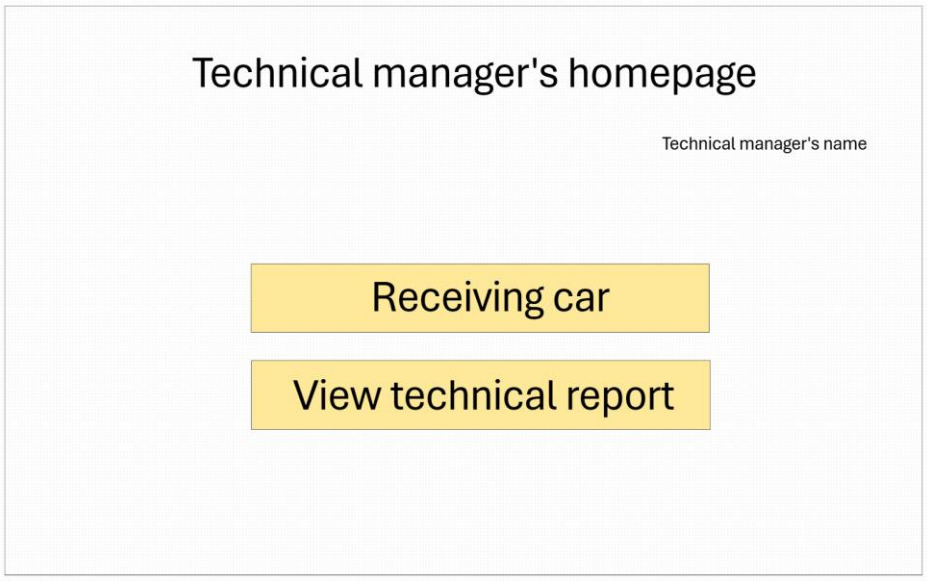
+ 1 tblSerCom – 1 tblComponent → tblComponent has foreign key componentID

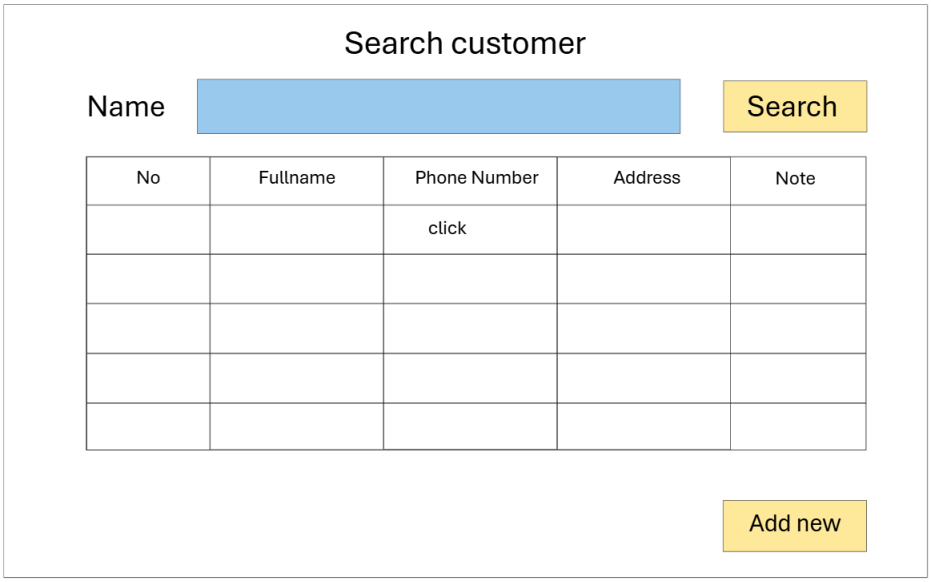
Step 5: Eliminate duplicate or inherit attributes

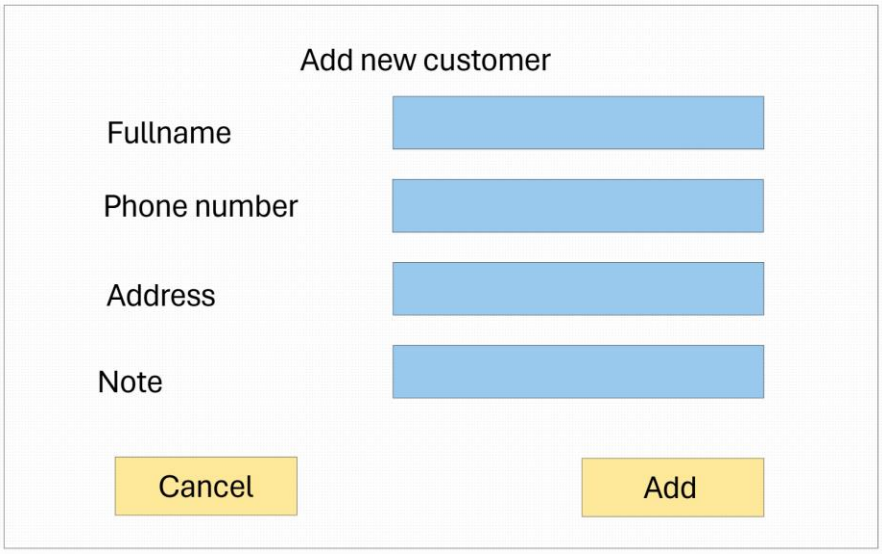


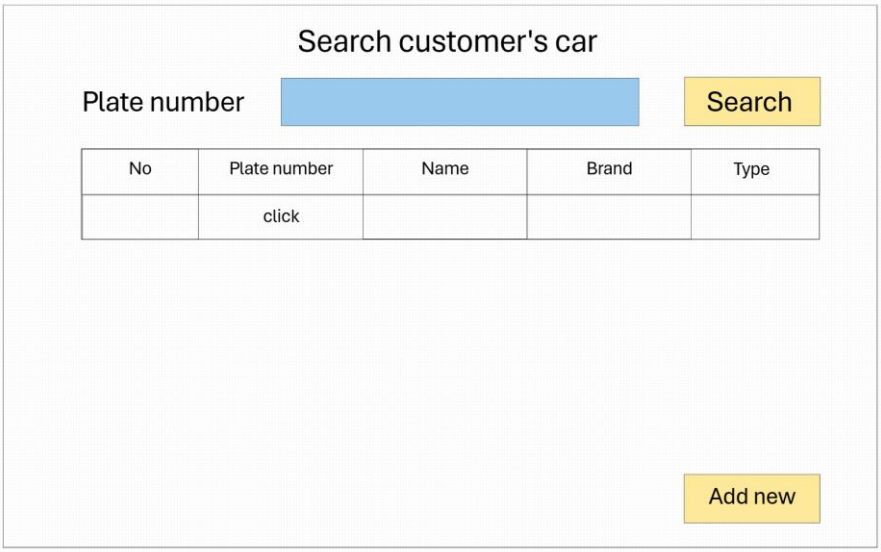
III. Design

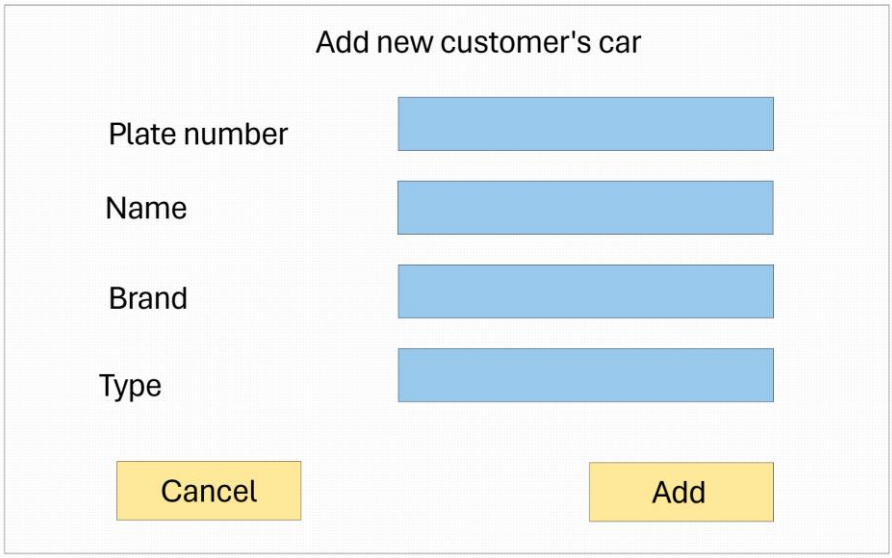


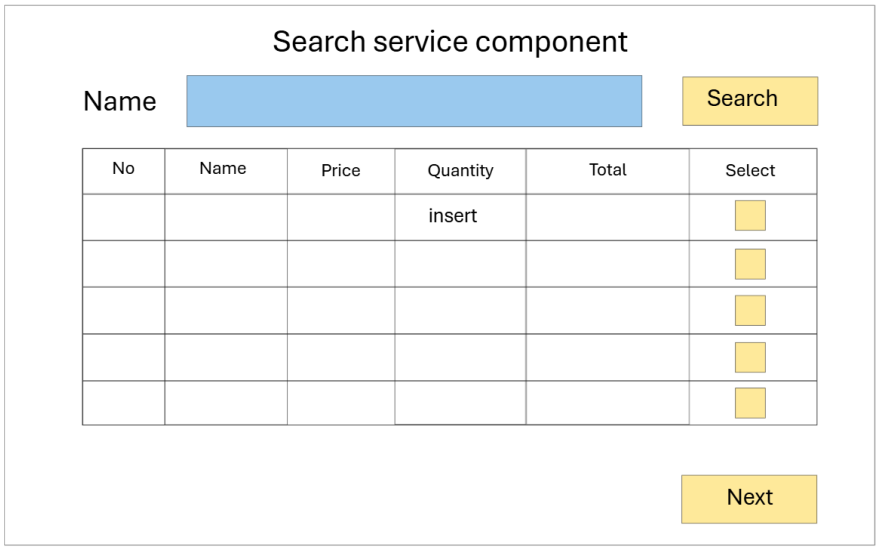

Technical manager's homepage
Technical manager's name
Receiving car
View technical report

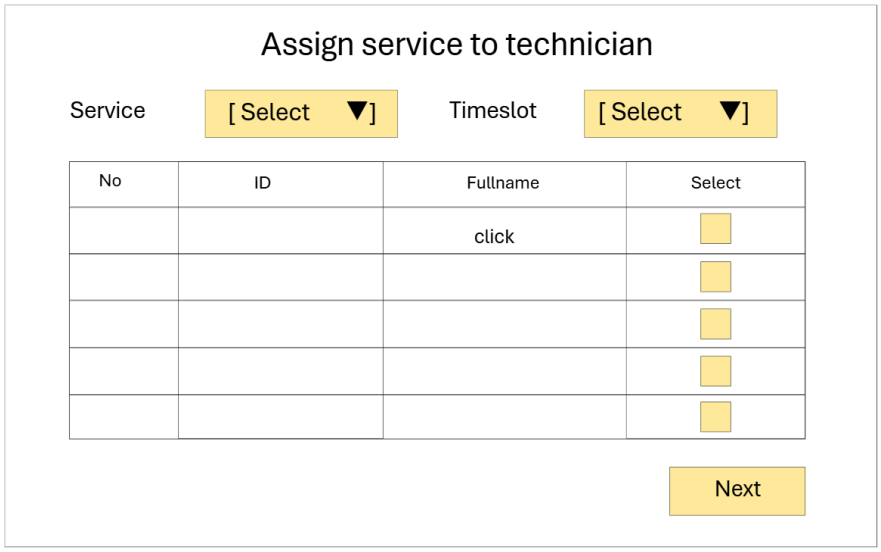

Search customer
Name
Search
No
Fullname
Phone Number
Address
Note
click
Add new

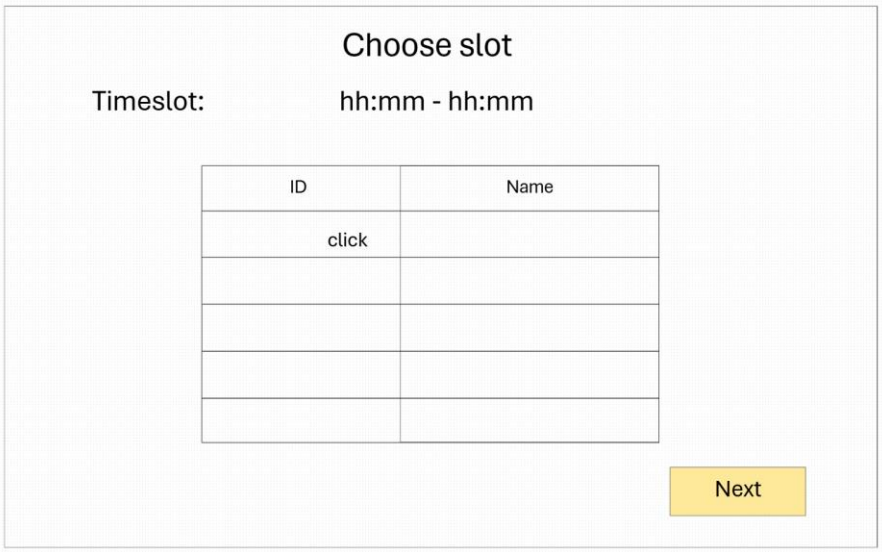

Add new customer
Fullname
Phone number
Address
Note
Cancel
Add

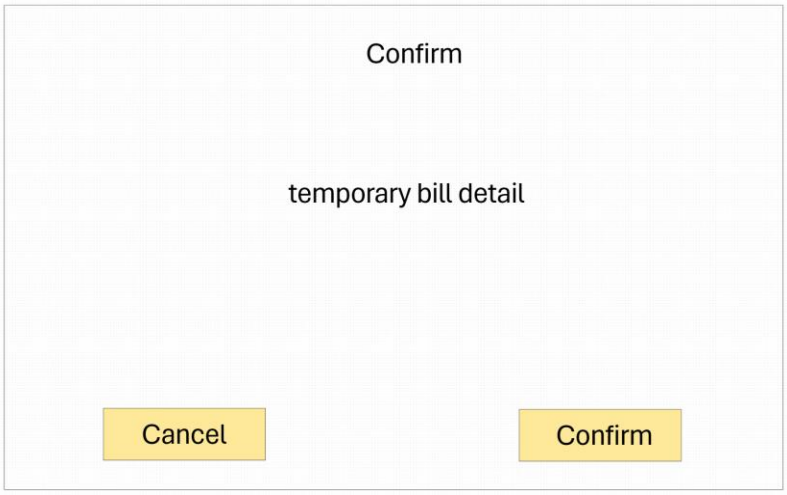

Search customer's car
Plate number
Search
No
Plate number
Name
Brand
Type
click
Add new


Add new customer's car
Plate number
Name
Brand
Type
Cancel
Add


Search service component
Name
Search
No
Name
Price
Quantity
Total
Select
insert
Next


Assign service to technician
Service
[Select ▼]
Timeslot
[Select ▼]
No
ID
Fullname
Select
click
Next


Choose slot
Timeslot:
hh:mm - hh:mm
ID
Name
click
Next



IV. Class diagram

View Classes:

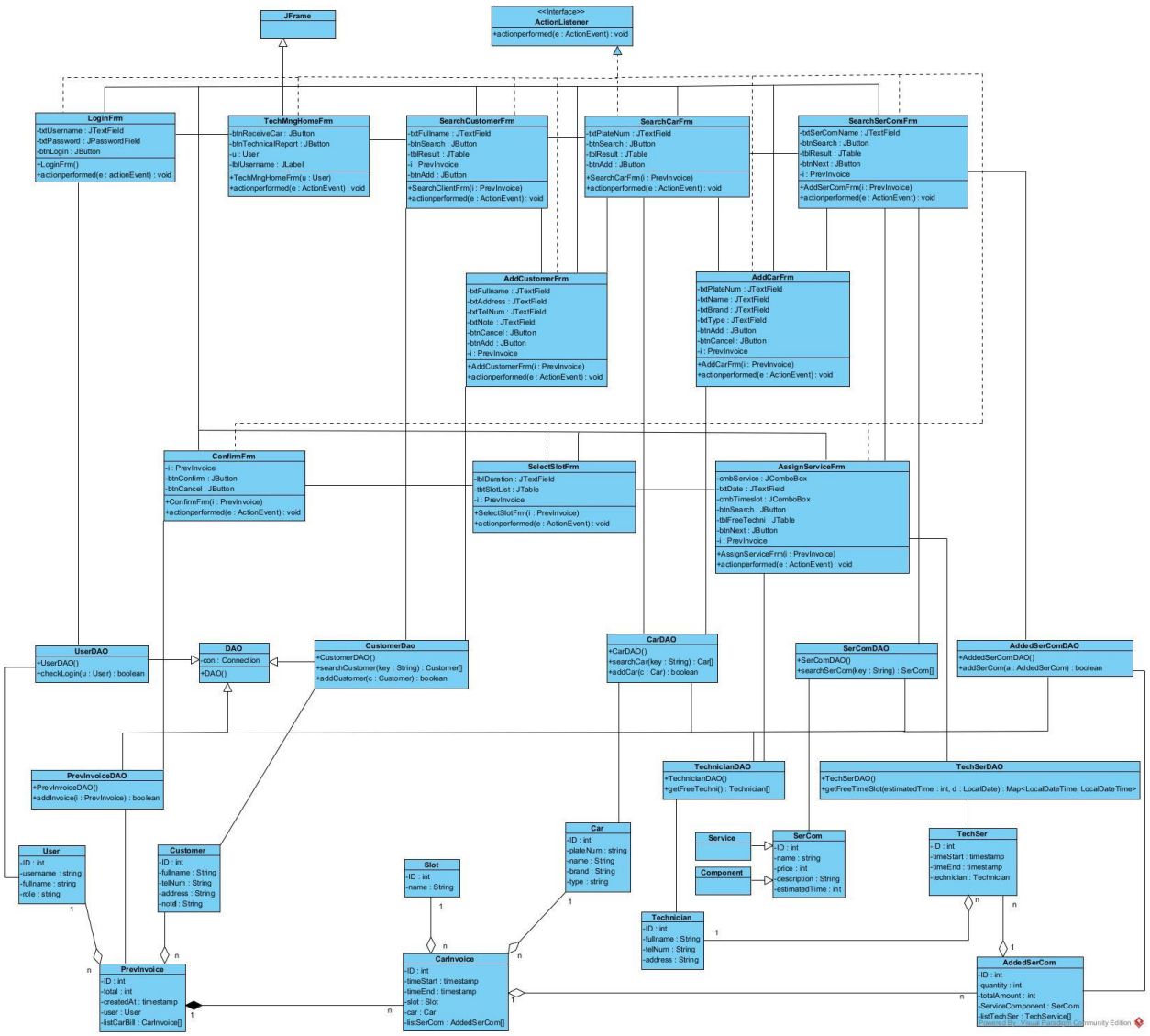
- LoginFrm is the interface to login. It needs a text field to enter the username, a text field to enter password, and a button to login.
- TechMngHomeFrm is the home interface for the technical manager. It needs at least a button to go to the receive car function.
- SearchCustomerFrm is the interface to search and select the customer. It needs a text field to enter the keyword to search client by name, a button to search, a table to show the list of founded customers, and a button to add new client if the customer does not exist in the database.
- AddCustomerFrm is the interface to add new customer. It needs the input text fields to enter the customer's information: name, address, phone number, note and a button to save, another button to cancel
- SearchCarFrm is the interface to search and select the car brought by customer. It needs a text field to enter the keyword to search car by plate number, a button to search, a table to show the list of founded cars, and a button to add new car if the car does not exist in the database
- AddCarFrm is the interface to add new car. It needs the input text fields to enter the car's information: plate number, name, brand, type and a button to save, another button to cancel
- SearchSerComFrm is the interface to search and add new service or component to repair the car. It needs a text field to search service/component by keyword and a button to search, a table to show the list of founded services and components which includes a column quantity allowing technical manager to insert the quantity and a column to select the right service/component

- AssignServiceFrm is the interface to assign technicians to services. It has a dropdown menu to select the service, a dropdown menu to select free timeslot, a table of list free technicians and a button to click next
- SelectSlotFrm is the interface to select slot to put the car in. It has a table to show list of slots that are free
- ConfirmFrm is the interface to confirm invoice information

Control (DAO) classes:

- DAO is a general class of DAO. It has only the construction to connect to the database and provides the common connection for all inherited DAO classes in the system.
- UserDAO is the class for manipulating with database related to the User object. In this module, it needs a method to verify whether the login information is correct or not, it is checkLogin() method.
- CustomerDAO has two methods:
    - searchCustomer(): to search customer whose name contains the entered key
    - addCustomer(): to add new customer into the database
- CarDAO has two methods:
    - searchCar(): to search car which has plate number contains the entered key
    - addCar(): to add new car into the database
- SerComDAO has a method to search service/component which name contains the entered key searchSerCom()
- AddedSerComDAO has a method to add new required service/component addSerCom()
- TechSerDAO has two methods:
    - getFreeTimeSlot(): to get a list of free time slot that has the duration bigger or equal to estimatedTime of the chosen service
    - addTechSer(): to add new TechSer into the database
- TechnicianDAO has a method to find list of technicians that is free in the chosen time slot getFreeTechni()
- SlotDAO has a method to find list of slots that are free between the time start of the earliest service and time end of the latest service getFreeSlot()
- PrevInvoiceDAO has a method to add new invoice addInvoice()

Entity classes: Customer, User, PrevInvoice, Car, CarInvoice, Slot, SerCom, Technician, AddedSerCom, TechSer



V. Sequence diagram

1. Scenario version 3

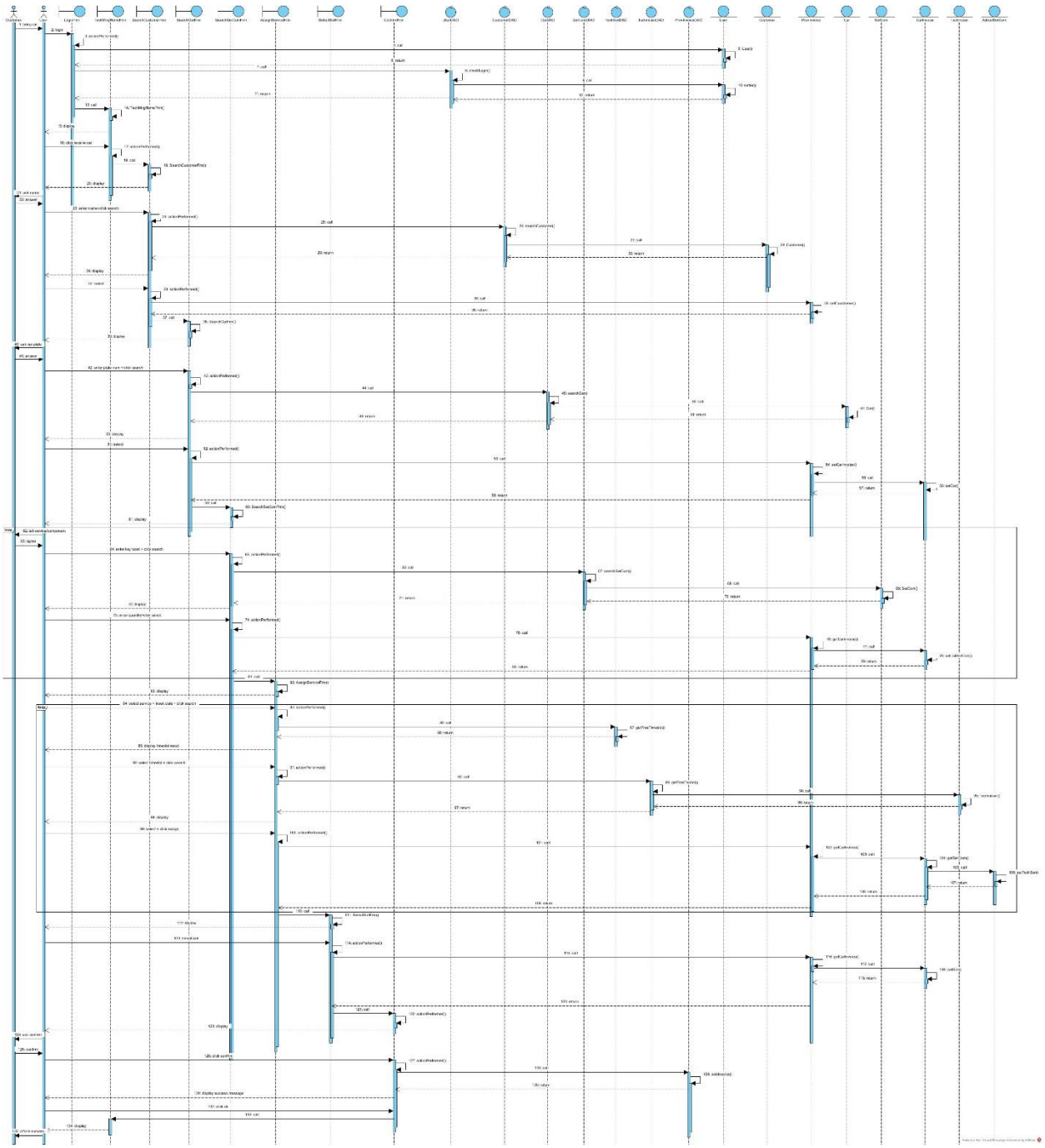
1. The customer brings car to fix
2. The technical manager enters username, password and clicks on the login button on LoginFrm
3. The method actionPerformed() of LoginFrm is called
4. The method actionPerformed() calls User to create an User object
5. The class User packs the information into an User object
6. The class User returns User object to the method actionPerformed()
7. The method actionPerformed() calls method checkLogin() of the class UserDAO
8. The method checkLogin() checks the login information
9. The method checkLogin() calls the class User set more two attributes name, role
10. The class User calls its method setName(), setRole()

11. The class User returns the User object to the method checkLogin()
12. The method checkLogin() returns the results to the actionPerfomed()
13. The method actionPerformed() calls the class TechMngHomeFrm
14. The constructor TechMngHomeFrm() is called
15. The TechMngHomeFrm is shown to the technical manager
16. The technical manager clicks on the receive car button
17. The method actionPerformed() is called
18. The method actionPerformed() calls the SearchCustomerFrm
19. The constructor SearchCustomerFrm() is called
20. The interface SearchCustomerFrm is shown to technical manager
21. The Technical manager asks the customer name
22. The customer answers the technical manager
23. The technical manager enters the customer's name and clicks search
24. The method actionPerformed() of class SearchCustomerFrm is called
25. The method actionPerformed() calls the method searchCustomer() of the class CustomerDAO
26. The method searchCustomer() executes
27. The method searchCustomer() calls the class Customer the pack the results
28. The class Customer packs Customer objects
29. The class Customer returns the packed objects to the method searchCustomer()
30. The method searchCustomer() returns the results to the method actionPerformed()
31. The method actionPerformed() displays the results on the SearchCustomerFrm to the technical manager
32. The technical manager clicks on the row corresponding to the right customer
33. The method actionPerformed() of the class SearchCustomerFrm is called
34. The method actionPerformed() calls the class PrevInvoice to add customer information to it
35. The PrevInvoice calls the method setCustomer()
36. The class PrevInvoice returns the packed object to the method actionPerformed()
37. The method actionPerformed() calls the class SearchCarFrm
38. The constructor SearchCarFrm() is called
39. The interface SearchCarFrm if shown to the technical manager
40. The technical manager asks the customer about the car's plate number
41. The customer answers the technical manager
42. The technical manager enters the car's plate number and click search
43. The method actionPerformed() of class SearchCarFrm is called
44. The method actionPerformed() calls the method searchCar() of the class CarDAO

45. The method searchCar() executes
46. The method searchCar() calls the class Car to pack the results
47. The class Car packs Car objects
48. The class Car returns the packed object to the method searchCar()
49. The method searchCar() return results to the method actionPerformed()
50. The method actionPerformed() displays results on the SearchCarFrm to the technical manager
51. The technical manager clicks on the row corresponding to the right car
52. The method actionPerformed() of class SearchCarFrm is called
53. The method actionPerformed() calls the class PrevInvoice
54. The class PrevInvoice calls the method setCarInvoice()
55. The method setCarInvoice() calls the class CarInvoice to add new car's information to it
56. The class CarInvoice calls the method setCar()
57. The class CarInvoice returns the packed object to the method setCarInvoice() of class PrevInvoice
58. The class PrevInvoice returns the packed object to the method actionPerformed()
59. The method actionPerformed() calls the class SearchSerComFrm
60. The constructor SearchSerComFrm() is called
61. The SearchSerComFrm is shown to the technical manager
62. The technical manager tells customer about the required service/component and its quantity
63. The customer agrees
64. The technical manager enters the key word and click search
65. The method actionPerformed() of class SearchSerComFrm is called
66. The method actionPerformed() calls the method searchSerCom() of the class SerComDAO
67. The method searchSerCom() executes
68. The method searchSerCom() calls the class SerCom to pack the results
69. The class SerCom packs the SerCom objects
70. The class SerCom returns the packed object to the method searchSerCom()
71. The method searchSerCom() returns results to the method actionPerformed()
72. The method actionPerformed() displays results on the SearchSerComFrm to the technical manager
73. The technical manager inserts the quantity for the correct service/component and clicks select
74. The method actionPerformed() of class SearchSerComFrm is called

75. The method actionPerformed() calls the class PrevInvoice
76. The class PrevInvoice calls the method getCarInvoice()
77. The method getCarInvoice() calls class CarInvoice
78. The CarInvoice calls the methods setListSerCom()
79. The class CarInvoice returns the packed object to the method getCarInvoice()
80. The method getCarInvoice returns to actionPerformed()
81. The method actionPerformed() calls the class AssignServiceFrm
82. The constructor AssignServiceFrm() is called
83. The interface AssignServiceFrm is shown to the technical manager
84. The technical manager selects the added service in the select service and insert date, the default is current day and click search Timeslot
85. The method actionPerformed() of class AssignServiceFrm is called
86. The method actionPerformed() calls the method getFreeTimeslot() of the class TechSerDAO
87. The method getFreeTimeslot() executes
88. The class TechSerDAO packs the results and returns to actionPerformed() of AssignServiceFrm
89. The AssignServiceFrm displays results to technical manager
90. Technical manager selects one timeslot and click search free technicians
91. The method actionPerformed() is called
92. The method actionPerformed() calls the method getFreeTechni() of class TechnicianDAO
93. The method getFreeTechni() executes
94. The method getFreeTechni() calls the class Technician to pack the results
95. The class Technician packs the Technician objects
96. The class Technician returns result to method getFreeTechni()
97. The method getFreeTechni() returns results to method actionPerformed() of AssignServiceFrm
98. The AssignServiceFrm displays results to technical manager to select
99. The technical manager selects the technicians then click assign
100. The method actionPerformed() of AssignServiceFrm is called
101. The method actionPerformed() calls class PrevInvoice
102. The class PrevInvoice executes methos getCarInvoice()
103. The method getCarInvoice() calls the method getSerCom() class CarInvoice
104. The method getSerCom() executes
105. The method getSerCom() calls the method setTechSer() of class AddedSerCom
106. The method setTechSer() executes

107. The class AddedSerCom returns result to getSerCom()
108. The class CarInvoice returns result to getCarInvoice()
109. The method getCarInvoice returns to actionPerformed()
110. The method actionPerformed() calls the class SelectFlotFrm
111. The constructor SelectSlotFrm() is called
112. The interface SelectSlotFrm is shown to technical manager to select slot
113. The technical manager selects a slot
114. The method actionPerformed() of SelectSlotFrm is called
115. The method actionPerformed() calls the method getCarInvoice() of class PrevInvoice
116. The method getCarInvoice() executes
117. The method getCarInvoice() calls the class CarInvoice to add a new slot to it
118. The class CarInvoice calls method setSlot()
119. The class CarInvoice returns the packed object to getCarInvoice() of class PrevInvoice
120. The class PrevInvoice returns to method actionPerformed()
121. The method actionPerformed() calls the class ConfirmFrm
122. The cconstructor ConfirmFrm() is called
123. The interface ConfirmFrm is shown to the technical manager
124. The technical manager repeats the invoice information to the client and asks him to confirm.
125. The customer confirms it.
126. The technical manager clicks on the confirm button.
127. The method actionPerformed() of the class ConfirmFrm is called.
128. The method actionPerformed() calls the method addInvoice() of the class PrevInvoiceDAO.
129. The method addInvoice () executes.
130. The method addInvoice () returns the turn to the method actionPerformed()
131. The method actionPerformed() displays a success message
132. The technical manager clicks on the OK button of the message.
133. The method actionPerformed() recalls the interface TechMngHomeFrm.
134. The interface TechMngHomeFrm is shown to the technical manager.
135. The technical manager confirms the success to the customer.



D. Testing

I. Black box test plan

No.	Module	Test case
1	Receiving car	Customer and car already exist
	Receiving car	Customer exists but car doesn't exist

	Receiving car	Customer doesn’t exist
	Receiving car	Log in with valid username and password
	Receiving car	Log in with invalid credentials (wrong username/password)
	Receiving car	Insert quantity and select service/component
	Receiving car	Select service/component but not insert quantity
	Receiving car	Add 2 times a service/component for a car invoice
	Receiving car	Select service and insert date to search for free timeslot
	Receiving car	Select service and insert date but in the past
	Receiving car	Select technicians and click assign
	Receiving car	Search for a service/component
	Receiving car	Search for a service/component but does not exist
	Receiving car	Select a slot for the car

II. Standard black box test case

1. Database before testing

- tblUser

ID	username	password	fullname	role
1	a	a	A	Technical Manager
2	m	m	M	Manager

- tblCustomer

ID	fullname	telNum	address	note
1	B	123	Hanoi	
2	Binh	124	Quoc oai	
3	An	125	Thanh oai	

- tblCar

ID	plateNumber	name	brand	type
1	30G-123	Mazda G6	Mazda	4 seats
2	30G-1234	Toyota X5	Toyota	7 seats
3	27G-123	Mazda G6	Mazda	4 seats

- tblSlot

ID	name
1	A2
2	A5
3	B3
4	B4

- tblTechnician

ID	fullname	telNum	address
1	Nguyen van a	098	Nam dinh
2	Dinh thi b	089	Hanoi
3	Pham thi c	890	Yen bai

- tblSerCom

ID	Name	Price	EstimatedTime	Description
1	Airbag	500.000đ		
2	Tire replacement	200.000đ	90	1-2 technicians
3	Flat tire repair	100.000đ	90	1-2 technicians

- tblService

ID
2
3

- tblComponent

ID
1

- tblTechService

ID	timeStart	timeEnd	serviceID	technicianID
1	2025-05-13 09:00	2025-05-13 10:30	3	3
2	2025-05-13 10:30	2025-05-13 12:30	2	2
3	2025-05-13 12:00	2025-05-13 13:30	3	1
4	2025-05-13 14:00	2025-05-13 15:30	2	2

-tblCarInvoice

ID	timeStart	timeEnd	carID	invoiceID	slotID
1	2025-05-13 09:00	2025-05-13 12:30	3	1	3
2	2025-05-13 12:00	2025-05-13 15:30	2	2	4

- tblPrevInvoice

ID	createdAt	userID	customerID
1	2025-05-13 09:00	1	3
2	2025-05-13 12:00	1	2

- tblAddedSerCom

ID	quantity	carInvoiceID	sercomID
1	1	1	3
2	2	1	2
3	1	1	1
4	1	2	3
5	1	2	2

2. Testing scenario and expected results

Scenario	Expected results

<table>
<tr><td>1. Login with username = a, password = a</td><td>Technical manager homepage UI appears:

- Receive car
- View technical report</td></tr>
<tr><td>2. Click receive car</td><td>Search customer UI appears: a textfield to enter name, button search, button add</td></tr>
<tr><td>3. Enter name=b, click search</td><td>Search customer UI name = b

<table><tr><td>ID</td><td>fullname</td><td>telNum</td><td>address</td><td>note</td></tr><tr><td>1</td><td>b</td><td>123</td><td>ha noi</td><td></td></tr><tr><td>2</td><td>binh</td><td>124</td><td>Quoc oai</td><td></td></tr></table></td></tr>
<tr><td>4. Select row with fullname=b</td><td>Search car UI appear: a textfield to enter plate number, button search, button add</td></tr>
<tr><td>5. Enter plate number = 30G-123</td><td>Search car UI plate number = 30G-123

<table><tr><td>ID</td><td>plateNumber</td><td>name</td><td>brand</td><td>type</td></tr><tr><td>1</td><td>30G-123</td><td>Mazda G6</td><td>Mazda</td><td>4 seats</td></tr><tr><td>2</td><td>30G-1234</td><td>Toyota X5</td><td>Toyota</td><td>7 seats</td></tr></table></td></tr>
<tr><td>6. Select row with plate number = 30G-123</td><td>Search service/component UI appear: a textfield to enter name, button search, button next</td></tr>
<tr><td>7. Enter keyword = tire to search for service Tire replacement</td><td>Search service/component UI for keyword=tire

<table><tr><td>ID</td><td>Name</td><td>Unit Price</td><td>Quantity</td><td>Item total</td><td>Select</td></tr><tr><td>2</td><td>Tire replacement</td><td>200.000đ</td><td></td><td></td><td>☐</td></tr><tr><td>3</td><td>Flat tire repair</td><td>100.000đ</td><td></td><td></td><td>☐</td></tr></table>
- Button: next</td></tr>
<tr><td>8. Update quantity to 2</td><td>Search service/component UI for keyword=tire</td></tr>
</table>

<table>
<tr>
<td>and select the first service</td>
<td>
<table>
<tr><th>ID</th><th>Name</th><th>Unit Price</th><th>Quantity</th><th>Item total</th><th>Select</th></tr>
<tr><td>2</td><td>Tire replacement</td><td>200.000đ</td><td>2</td><td>400.000đ</td><td>X</td></tr>
<tr><td>3</td><td>Flat tire repair</td><td>100.000đ</td><td>1</td><td>100.000đ</td><td></td></tr>
</table>
- Button: next
</td>
</tr>
<tr>
<td>9. Click next</td>
<td>Assign service to technician UI appears:

Service: [Select ▼]

Date: 13/05/2025

Button: Search timeslot

Time slot: [Select ▼] Button: search technician

<table>
<tr><th>No</th><th>ID</th><th>Name</th><th>Select</th></tr>
</table>

Button: next</td>
</tr>
<tr>
<td>10. Select the service S467 and keep the current date then click search free time</td>
<td>- The time slots which are equals to estimated time of service will appears in the time slot JComboBox

Service: [S467 - Tire replacement]

Date: 13/05/2025

Button: Search timeslot

Time slot: [Select ▼] Button:search technician

[13:30 – 15:00]

[14:30 – 16:00]

<table>
<tr><th>No</th><th>ID</th><th>Name</th><th>Select</th></tr>
</table>
</td>
</tr>
</table>

<table>
<tr>
<td>12. Select the timeslot 13:30 – 15:00 and click search technician</td>
<td>The list of free technicians in the chosen time slot will appear in the list

Service: [S467 - Tire replacement]

Time slot: [13:30 – 15:00]

<table>
<tr><th>ID</th><th>Name</th><th>Select</th></tr>
<tr><td>1</td><td>Nguyen Van A</td><td>☐</td></tr>
<tr><td>3</td><td>Pham Thi C</td><td>☐</td></tr>
</table>
</td>
</tr>
<tr>
<td>11. Select technician 1 and 3 then click next</td>
<td>Select slot UI appears with list of slots that are free in 13:30-15:00 in 13/05/2025:

Time slot: 13:30-15:00

<table>
<tr><th>ID</th><th>Name</th></tr>
<tr><td>1</td><td>A2</td></tr>
<tr><td>2</td><td>A5</td></tr>
</table>
</td>
</tr>
<tr>
<td>12. Select slot A2</td>
<td>Confirm invoice UI appears:

Customer’s name: b Phone number: 123

Car: 30G-123 Car’s name: Mazda G6

Created at: 13:00 10/05/2024

Slot: A2

<table>
<tr><th>No</th><th>Name</th><th>Unit price</th><th>Quantity</th><th>Total</th><th>TechnicianID</th></tr>
<tr><td>1</td><td>Tire replacement</td><td>200.000đ</td><td>2</td><td>400.000đ</td><td>1, 3</td></tr>
</table>
Total: 400.00đ

Button: cancel, confirm</td>
</tr>
</table>

13. Click confirm button	A message appears: Receive car successfully!
14. Click OK	Return to technical manager homepage

3. Database after testing

- tblTechService

ID	timeStart	timeEnd	serviceID	technicianID
1	2025-05-13 09:00	2025-05-13 10:30	3	3
2	2025-05-13 09:30	2025-05-13 11:00	2	2
3	2025-05-13 12:00	2025-05-13 13:30	3	1
4	2025-05-13 14:00	2025-05-13 15:30	2	2
5	2025-05-13 13:30	2025-05-13 15:00	2	1
6	2025-05-13 13:30	2025-05-13 15:00	2	3

- tblCarInvoice

ID	timeStart	timeEnd	carID	invoiceID	slotID
1	2025-05-13 09:00	2025-05-13 15:30	3	1	3
2	2025-05-13 12:00	2025-05-13 15:30	2	2	4
3	2025-05-13 13:30	2025-05-13 15:00	1	2	1

- tblPrevInvoice

ID	createdAt	userID	customerID
1	2025-05-13 09:00	1	3
2	2025-05-13 12:00	1	2
3	2025-05-13 13:00	1	1

- tblAddedSerCom

ID	quantity	carInvoiceID	sercomID
1	1	1	3
2	2	1	2
3	1	1	1

4	1	2	3
5	1	2	2
6	1	3	2
7	1	3	2

- Others tables do not change