Rabbeshly Station

Software Requirements Specification

V0.9.8

1/Mars/2024

Sanad AlArousi
Lead Software Engineer

Prepared for
ITSE311—Software Requirements Analysis
Lecturer: Fatima Ben Lashihar
Fall 2023

Rabbeshly Station SRS Content

This comprehensive system acts as both an Enterprise Resource Planning (ERP) platform and a marketplace, catering to various sectors of the Libyan car industry and its users. It offers a diverse range of services that benefit various stakeholders, including everyday drivers, mechanics, shop owners, scrapyards, and even the government

Products/Scope:

- **1.1 Car Station:** A platform for users to manage their car information and share their passion for cars. This includes:
  - **1.1.1 Car Profile:** Store car details and receive personalized maintenance reminders.
  - **1.1.2 Suggestions and tips:** for the <owners> car to help preserve the cars value condition, such as: ‘Regular Maintenance reminders’**.**
  - **1.1.3 Car-Station Feed:** Share and discuss car-related news and experiences
  - **1.2 Shop Station:** A point-of-sale (POS) system for shops to manage their business operations, including:
  - **1.2.1 Shop Profile:** Increase online presence and sales through an online store.
  - **1.2.2 A.I-driven Assistant:** Optimize sales, purchases, and inventory through data-driven insights.
  - **1.2.3 Online/On-site Management tools:** Manage sales data, employee accounts, inventory, sales performance, and more.
- **1.3 Workshop Station:** An ERP and POS system to manage workshop workflow, including:
  - **1.3.1 Workshop Profile:** Showcase workshop expertise and work history to attract customers.
  - **1.3.2 A.I-driven Assistant:** Improve decision-making for parts per-ordering, avoiding unsuitable repairs, and increasing profitability.
  - **1.3.3 Online/On-site Management tools:** Streamline workflow by managing repairs, providing customer estimates, tracking employee performance, and more.
- **1.4 Marketplace:** A customization feed showcasing cars, workshops, scrapyards, and advertisements, allowing users to buy, sell, and promote products and services.
- 1.5 **Service Station:** The central hub connecting all services, allowing:
  - 1.5.1 **Car owners:** Find nearby workshops based on specialization, previous work, or similar repairs.
  - 1.5.2 **Businesses:** Connect workshops with nearby shops and scrapyards for parts procurement.
  - 1.5.3 **Web scrapping:** Gather data on similar items from external platforms like Facebook and eBay.

- 1.6 **Data Station:** A central data repository for managing:
    - **1.**6.1 **Basic data:** Manage business, car, and customer data.
    - **1.6.2 AI data:** Gain insights through AI analysis.
    - **1.6.3 Statistical data:** Analyze market trends and competitor behavior.
    - **1.6.4 Web scrapped data:** Gather data on similar items from external platforms like Facebook and eBay.
    - **1.6.5 Government access:** Provide authorized entities with data access for regulatory purposes.

1.2. Audience segments:

1.2.1. **Owners:** Car, shop, workshop, and scrapyard owners.

- 1.2.2. **Workers:** Mechanics and other workers in shops, workshops, and scrapyards.
- 1.2.3. **Businesses:** Shops, workshops, and government entities.
- 1.2.4. **Others:** Individuals passionate about cars.
- 1.2.5. **B2B Solution seekers:** Businesses looking for solutions to optimize their operations.
- **1.3 Not Station-like:**
- **1.3.1. Not station-like functionalities:**
- 1.3.1.1 Transfer of ownership and vehicle registration
- 1.3.1.2 The system won’t handle car-related loans
- 1.3.1.3 The system won’t provide car insurance
- 1.3.1.4 Social media features beyond the car-centric feed: While the Car-Station feed
- allows sharing car-related content, features like friend requests, or non-
- ar-related content wouldn't be part of the system.

- 1.3.1.5 Car delivery or repair services: The system would connect users with workshops, but wouldn't offer car delivery, towing, or on-site repair services.

1.3.2. Not station-like entities:

- 1.3.2.1 **Non-car-related businesses:** This system focuses on the car industry. Entities like restaurants, grocery stores, or other non-car-related businesses wouldn't be included in the marketplace or other functionalities.
- 1.3.2.2 **Private individuals selling personal items:** While individuals can sell car-related products in the marketplace, selling non-car-related personal items wouldn't be supported.
- 1.3.2.3 **Non-Libyan entities:** The system is currently targeted toward users and businesses in Libya. International users or businesses wouldn't be included in the initial scope, but would be added in future expansion.

1.3.3. Not station-like audience:

- 1.3.3.1 **Individuals under 18:** The system requires users to be legal adults to register and utilize its services.
- 1.3.3.2 **Individuals without an internet connection or compatible device:** The system relies on internet access and mobile devices, so individuals without these wouldn't be part of the target audience.
- 1.3.3.3 **Individuals not interested in cars:** The system focuses on the car industry, so individuals with no interest in cars wouldn't be a primary target audience.
- **1.4 Overview**
- 1.4.1 **Car Station:** A user-centric platform for managing car information, connecting with other car enthusiasts, and receiving personalized maintenance reminders.
- 1.4.2 **Shop Station:** A point-of-sale (POS) system for shops to manage sales, inventory, and online presence, boosting their reach and efficiency.
- 1.4.3 **Workshop Station:** An integrated ERP and POS system for workshops, enabling efficient workflow management, improved customer service, and optimized parts procurement.
- 1.4.4 **Marketplace:** A customizable feed showcasing cars, workshops, scrapyards, and advertisements, facilitating buying and selling of car-related products and services.
- 1.4.5 **Service Station:** The central hub connecting all stations, empowering car owners to find suitable workshops based on specialization, previous work, or similar repairs, and enabling workshops to source parts efficiently.
- 1.4.6 **Data Station:** A central repository for storing and managing all system data, providing insights through AI analysis, statistical data, and web scraping to empower informed decision-making.

2.0 General Description

- **2.0.1 Market Conditions:**
    - 2.0.1.1 **Internet Penetration:** Increasing internet access (reaching 70% of the population in Libya) fosters the adoption of software solutions to meet evolving needs.
    - 2.0.1.2 **Business Landscape:** While many businesses use social media (e.g., Facebook) and location services (e.g., Google Maps), they often lack the tools to fully leverage the potential of the internet.
- **2.0.2 Technology Gap:**
    - 2.0.2.1 **Uninformed Customers:** Limited knowledge about technology makes finding suitable software solutions challenging.
    - 2.0.2.2 **IT Professional Opportunity:** This lack of awareness creates a market gap for readily accessible and affordable software solutions, with the car industry being a prime example.
- 2.0.3 **Market Needs:**
    - 2.0.3.1 **Workshop Management:** Existing solutions lack functionality for managing customer car repairs, hindering efficient workflow and customer service.
    - 2.0.3.2 **Shop Management:** Current practices, such as relying on social media for sales and marketing, are not sustainable for long-term growth.
    - **2.0.4 Manual processes and limited technology usage create several challenges:**
        - 2.0.4.1 Inefficient workshop operations Limited sales and marketing reach for shops have both lead to intangible losses in sales and customer base
        - 2.0.4.2 Difficulty in managing financial records leading to
        - 2.0.4.2 Lack of transparency and traceability in repair processes

Proposed Solution: Rabbeshly Station :)

- **Combines point-of-sale functionality** *with* **supply chain management features**.
- **Streamlined workflows:** Simplifies workshop operations and customer interactions.
- **Enhanced sales and marketing:** Offers online presence and customer engagement tools.
- **Improved data management:** Facilitates efficient record-keeping and financial management.

2.1 Product Perspective

Many established companies offer functionalities similar to different aspects of my proposed system, although none provide a fully encompassing solution like Rabbeshly Station. Here are what we manage to note:

- **2.1.1 Garage Management Systems (GMS):** These systems like **AutoLeap** and **Shop Boss** primarily focus on workshop management, providing features like appointment scheduling, job tracking, inventory control, and invoicing.

- **2.1.2 E-commerce Marketplaces:** Platforms like **eBay Motors, Autotrader** and **Copart** facilitate buying and selling cars, parts, may in the future expand to the Libyan market, However, they lack functionalities like workshop integration or repair history tracking.

- **2.1.3 Social media platforms:** Platforms like **Facebook Marketplace** can be used for buying and selling car-related items, but lack features specific to the car industry.

- **2.1.4 Enterprise Resource Planning (ERP) Systems** Solutions like **SAP** and **Oracle** offer broader functionalities for managing various aspects of a business, but are typically complex and expensive, making them less suitable for smaller businesses in the car industry.

- **2.1.4 Emerging startups:** Given the growing internet penetration, Libyan entrepreneurs might be developing similar solutions focusing on the Libyan market.

- **2.1.6 Custom-developed software:** Some shops or workshops at car dealerships might have opted for custom software solutions tailored to their specific needs made by another company.

2.2 Product Functions

- ### 2.2.1 Car Station:

    - **2.2.1.1 Car Profile:** Users can store car information, access personalized maintenance reminders, and connect with other car enthusiasts.

    - **2.2.1.2 Suggestions & Tips:** The system provides AI-powered suggestions and tips to help users maintain and preserve the value of their cars.

    - **2.2.1.3 Car-Station Feed:** Users can share news, pictures, and discussions about their cars within a car-centric social media platform.

- ### 2.2.2 Shop Station:

    - **2.2.2.1 Shop Profile:** Businesses can create online profiles to showcase their services, reach new customers, and boost their online presence.

    - **2.2.2.2 AI-Driven Assistant:** An AI assistant provides sales, purchasing, and inventory management insights to optimize business operations.

- **2.2.2.3 Online/On-Site POS & Management Tools:** Users can access point-of-sale functionalities, manage sales data, employee accounts, inventory, and track sales performance.

- **2.2.3 Workshop Station:**
  - 2.2.3.1 **Workshop Profile:** Workshops can create online profiles to showcase their expertise, work history, and attract new customers.
  - 2.2.3.2 **AI-Driven Assistant:** An AI assistant supports decision-making regarding parts pre-ordering, identifying suitable repair opportunities, and improving overall profitability.
  - 2.2.3.3 **Online/On-Site POS & Management Tools:** Streamline workshop operations by managing repairs, providing customer estimates, tracking employee performance, and managing other relevant data.

- **2.2.4 Marketplace:**
  - 2.2.4.1 **Marketplace Feed:** Users can browse a customizable feed showcasing cars, workshops, scrapyards, and advertisements, facilitating buying and selling of car-related products and services.
  - 2.2.4.2 **Online Payments:** The system can facilitate secure online payments for transactions within the marketplace (optional, depending on implementation).

- **2.2.5 Service Station:**
  - 2.2.5.1 **Service Station Platform Connectivity:** Connects all system components, allowing car owners to find suitable workshops based on various criteria and facilitating parts procurement for workshops from nearby shops or scrapyards.

- **2.2.6 Data Station:**
  - 2.2.6.1 **Data Station Platform Connectivity:** Provides centralized data storage and management for all system functionalities.

2.3 User Characteristics

2.3.1 **Owners:**

- 2.3.1.1 **Car Owners:** Any entity, person or shop that owns a car.
- 2.3.1.2 **Shop Owners:** Product selling owners, car parts or cars alike.
- 2.3.1.3 **Workshop Owners:** Independent technicians or businesses.
- 2.3.1.4 **Scrapyard Owners:** Person who operates or assists at operating the scrap yard.

2.3.2 **Workers:**

- 2.3.2.1 **Mechanics:** Independent professionals whom are willing to be identified as `portable workshop.
- 2.3.2.2 **Workshop Mechanics:** Professionals who work for a workshop owner and at a designated location.
- 2.3.2.3 **Scrapyards Mechanics/Workers:** Individuals who run or work at scrapyards.

2.3.3 Businesses:

2..3.3.1 **Auto Parts Stores:** Businesses specializing in selling new and used car parts and accessories.

2..3.3.2 **Car Dealerships:** Businesses selling new and used cars.

2..3.3.3 **Government Agencies:** Entities responsible for regulating and overseeing the car industry, such as transportation departments or environmental agencies.

2.3.4 Car Enthusiasts:

2.3.4.1 **Car Enthusiasts:** Individuals with a strong passion for cars, including:

2.3.4.2 **Modders:** Individuals modifying and customizing car appearance and performance.

2.3.4.3 **Painters:** Professionals specializing in car painting and detailing.

2.3.4.4 **Importers:** Businesses importing specialized car parts, modded cars, or unique car models.
2.3.4.5 **Influencers:** Social media personalities sharing car-related content and experiences.

2.3.4.6 **Event Organizers:** Individuals or organizations planning and hosting car shows, races, or other car-related events.

2.3.5 B2B Solution Seekers:

- 2.3.5.1 **Insurance Companies:** Entities seeking data and insights to assess risk and optimize insurance offerings for the car industry.
- 2.3.5.2 **Financial Institutions:** Banks and financial institutions looking for data-driven solutions to improve car loan and financing processes.
- 2.3.5.3 **Data Analytics Companies:** Businesses seeking access to platform data for market research and trend analysis within the car industry.
- 2.3.5.4 **Logistics Companies:** Businesses seeking optimized solutions for car transportation and parts delivery within the ecosystem.

2.4 General Constraints:

- **2.4.1 Hardware Constraints:**
  - **2.4.1.1 Device Requirements:**
    - **2.4.1.1.1 Minimum Phone Cost:** All user roles (Owners, Shops, Workshops, Scrapyards) require a phone costing at least 1200 Libyan dinars (LYD) with an internet connection accessible at least once daily.
    - **2.4.1.1.2 Touchscreen Devices:** Workshops are encouraged to use touchscreen devices for improved user experience with the system functionalities.
    - **2.4.1.1.3 Shops:** Minimum specifications for Shop computers include 32-bit architecture, 6GB of RAM, a Core i3 processor (5th generation or newer), and Windows 8.1 or 10 operating system.
  - **2.4.2 Software Constraints:**
    - **2.4.2.1 Performance and Scalability:**
      - 2.4.2.1.1 The system should operate efficiently on low-end devices with minimal features enabled to ensure accessibility for users with limited resources.
      - 2.4.2.1.2 The platform should be built using **No-SQL databases** to accommodate the expected surge in data volume.
      - 2.4.2.1.3 Programming with languages like Go and Rust is recommended to ensure fast system execution.
  - **2.4.3 Integration and Data Acquisition:**
    - 2.4.3.1 The platform should seamlessly integrate with APIs from external websites and services to import and display relevant data and content.
    - 2.4.3.2 Web scraping capabilities are required to extract data from various online platforms related to services and items within the car industry.
  - **2.4.4 Compatibility:**
    - 2.4.4.1 The platform should be compatible with Windows 8, 10, and Linux operating systems.
    - 2.4.4.2 Support for 32-bit architecture computers is necessary to ensure wider accessibility.
  - **2.4.5 Interface Constraints:**
    - **2.4.5.1 Identity Verification:**
      - The platform will implement two-factor authentication (2FA) using SMS messages via the "EASYSMS" service provided by "devs.ly".
      - Integration with the National Database is essential to verify the identity of sellers with relevant government organizations.

- **2.4.6 Regulatory Constraints:**
  - The system development and operations will adhere to **Libyan Law No.4 of 1990** and any other relevant regulations.
- **2.4.7 Environmental Constraints:**
  - **2.4.7.1 Scrapyards:**
    - Mobile device usage is preferred and encouraged due to the work environment.
    - Computers should be located indoors and not exposed to the elements.
  - **2.4.7.2 Shops and Workshops:**
    - Touchscreens should be protected with appropriate covers to prevent ` damage.
    - Access to the main computer should be restricted to authorized shop owners only.
    - Computers should be equipped with reliable power supplies to minimize disruptions

2.5 Assumptions and Dependencies:

- **2.5.1 User Adoption:**
  - **2.5.1.1 Shop Owners:**
    - **2.5.1.1.1 High Reliance:** Experienced shop owners with existing business challenges are assumed to be more receptive to the system's features and functionalities if presented clearly and offer a user-friendly on-boarding process. Additionally, shop owners who expect a direct return on investment (ROI) or a noticeable increase in sales and inventory management efficiency are more likely to adopt the system.
    - **2.5.1.1.2 Low Reliance:** New, inexperienced, or opportunistic shop owners may be hesitant to invest due to:
    - **2.5.1.1.2.1 Limited capital:** New businesses might prioritize other investments during the initial stages, putting software implementation on hold.
    - **2.5.1.1.3 Lack of understanding:** Inexperienced owners might not fully grasp the system's benefits and may prioritize immediate profits over long-term value and improved customer experience.
    - **2.5.1.1.4 Free alternatives:** Some owners might seek free, potentially ill-fitting solutions instead of investing in a tailored system.
  - **2.5.1.2 Scrapyard Owners:**
    - **2.5.1.2.1 Low Reliance:** Given the established and potentially multi-generational nature of scrapyard management, owners might be less likely to adopt a new system due to:
    - **2.5.1.2.2 Adherence to tradition:** Familiarity with existing practices might lead to reluctance to embrace new technology.
    - **2.5.1.2.3 Limited system applicability:** The platform's functionalities might not directly address all aspects of scrapyard operations, creating hesitation to invest.
  - **2.5.1.3 Workshop Owners:**
    - **2.5.1.3.1 Medium Reliance:** The workshop business model in Libya, often focusing on daily cash flow for families rather than comprehensive accounting practices, might influence system adoption:

- **2.5.1.3.2 Selective adoption:** Workshop owners might primarily value features that enhance customer visibility and engagement, potentially disregarding functionalities requiring additional equipment investment.

- **2.5.2. Market Conditions:**
  - **2.5.2.1 Technology adoption:** The overall rate of technology adoption within the Libyan car industry, especially among smaller businesses, could impact user acceptance of the system.

- **2.5.3. External Dependencies:**
  - **2.5.3.1 API integration:** The functionality of the system relies on seamless integration with external APIs like "devs.ly", National Database and T-lync. Any changes or limitations in these external services could require adjustments to the system.

- **2.5.4. Regulatory Environment:**
  - **2.5.4.1 Evolving regulations:** New, current or changing regulations could necessitate modifications to the system to ensure compliance.

- **2.5.5. User Feedback:**
  - **2.5.5.1 Continuous feedback:** User feedback throughout the development process and after deployment will be crucial to identify unforeseen challenges and potential needs not addressed by the initial requirements

3. Specific Requirements

- ### 3.0.1.1 Car Station Requirements:
  - **3.0.1.1.1 Car Profile:**
    - 3.0.1.1.1.1 Add Car through filling a form #1
    - 3.0.1.1.1.2 Add Car through taking pictures of it#2
    - 3.0.1.1.1.3 Add time of regular maintenance #3
    - 3.0.1.1.1.4 Add time and cost of heavy maintenance #4
    - 3.0.1.1.1.4 Add time and cost of Precautionary maintenance #5
    - 3.0.1.1.1.6 Add time and cost of Crash maintenance #6
    - 3.0.1.1.1.7 Add Car Specs #7

- 3.0.1.1.1.8 Add Car additional pictures #8
- 3.0.1.1.1.9 Add Car Special specs #9
- *3.0.1.1.1.10 Add persistent issues or accidents #10*
- *3.0.1.1.1.11 Add Car chassis/vin number #11*
- *3.0.1.1.1.12 Display Total maintenance costs #12*
- *3.0.1.1.1.13 Display Car efficiency #13*
- *3.0.1.1.1.14 Display Car #14*
- *3.0.1.1.1.15 Share update to the Feed #15*

- 3.0.1.1.2 Suggestion and tips: to be supplied by Data Station and Service Station later #16

- **3.0.1.1.3 Car Station Feed**
  - 3.0.1.1.3.1 Share post to other platforms #17
  - 3.0.1.1.3.2 Comment, Like and re-share on your Car Profile #18
  - 3.0.1.1.3.3 Create new posts, add pictures from your car profile or from device, write caption, add tags #19
  - 3.0.1.1.3.4 Interact with posts from other platforms such as a Facebook Group from the same niche #20

- **3.0.1.2 Shop Station Requirements:**
  - 3.0.1.2.1 Shop Profile**:**
    - 3.0.1.2.1.1 General Requirements of ***Shops***:
      - 3.0.1.2.1.1.1 Add Shop Details through filling a form (1.2.1) #21
      - 3.0.1.2.1.1.2 Add Product/Car٩ sale price, name and cost (1.2.3) #22
      - 3.0.1.2.1.1.3 Add Categories of products (1.2.3) #23
      - 3.0.1.2.1.1.4 Import spreadsheet to add past sale data (1.2.3) #24
      - 3.0.1.2.1.1.5 Export sales and storage reports (1.2.3) #25
      - 3.0.1.2.1.1.6 Respond to customer inquiries of parts and prices (1.2.3) #26
      - 3.0.1.2.1.1.7 Make offers and flash sales (1.2.3) #27
      - 3.0.1.2.1.1.8 Add employees (1.2.1) #28
      - 3.0.1.2.1.1.9 Decide Employees permissions in shop system (1.2.1) #29
      - 3.0.1.2.1.1.10 Track productivity of each employee (1.2.3) #30
      - 3.0.1.2.1.1.11 Add Discount percent/flat amount to the purchase (1.2.3) #31
    - 3.0.1.2.1.2 Auto Parts Shop:
      - 3.0.1.2.1.2.1 Add shelf position of the products (1.2.3) #32
      - 3.0.1.2.1.2.2 Respond to customer inquiries of parts and prices (1.2.3) #33
      - 3.0.1.2.1.2.3 Manage storage and capital of products stored (1.2.3) #34

- 3.0.1.2.1.3 Car Dealership shop:
  - 3.1.2.1.3.1 Add car make, model and location of shipping (1.2.3) #35

- 3.0.1.3 Workshop Station Requirements**:**
  - **3.0.1.3.1 Workshop Profile:**
    - 3.0.1.3.1.1 Add pictures of equipment (1.3.1) #36
    - 3.0.1.3.1.2 Add employees and technicians by name (1.3.3) #37
    - 3.0.1.3.1.3 Control employees/technicians access (1.3.3) #38
    - 3.0.1.3.1.4 Allow technicians to add their past expertise (1.3.3) #38
    - 3.0.1.3.1.5 Tech/employee record keep arrival of cars for repairs (1.3.3) #39
    - 3.0.1.3.1.6 Technician marks the start of diagnoses and repair (1.3.3) #40
    - 3.0.1.3.1.7 Technician/Car-Owner/Employee can order new/used parts through 1.5 #41
    - 3.0.1.3.1.8 Tech/employee marks the end of repair (1.3.3) #42
    - 3.0.1.3.1.9 Display the statistics of the workshop, per car, per make, per model, per repair #42
    - 3.0.1.3.1.10 Add cost of each repair for each make and model (1.3.3) (1.3.1) #43

- 3.0.1.4 Marketplace Requirements**:**
  - 3.0.1.4.1 Share product, offer, or owned car to the marketplace #44
  - 3.0.1.4.2 Add details of the posting #45
  - 3.0.1.4.3 Share posting on other websites #46
  - 3.0.1.4.4 Every user has its own market feed specialized to his owned car or business#47
  - 3.0.1.4.5 Search with filters to look for the desired part or price #48

3.1 External Interface Requirements
3.1.1 User Interfaces
3.1.2 Hardware Interfaces
3.1.3 Software Interfaces
3.1.4 Communications Interfaces

3.2 Functional Requirements

- ## 3.2.1.1 Car Station Requirements: #1

3.2.1.1.1.Add Car through filling a form #1

- 3.2.1.1.1.1 **Introduction:**
  - The system shall provide a user interface form for adding a car profile. The user shall be able to input the following data:
- 3.2.1.1.1.2 **Inputs:**
  - **Car Make (text):** Mandatory field. User enters the car manufacturer (e.g., Toyota, Volkswagen).

- **Car Model (text):** Mandatory field. User enters the specific car model (e.g., Corolla, Passat, Tiguan).
- **Year (number):** Mandatory field. User enters the car's manufacturing year. The system shall implement validation to ensure the entered year is within a reasonable range (e.g., past 15 years).
- **License Plate Number (text):** Mandatory field. User enters the car's license plate number. The system shall perform validation to ensure the format complies with the user's country regulations (e.g., specific number/letter combination for a particular country).
- **License Plate Number (text):** Mandatory field. User enters the car's license plate number. The system shall perform validation to ensure the format complies with the user's country regulations (e.g., specific number/letter combination for a particular country).
- **VIN (Vehicle Identification Number – text):** Mandatory field. User enters the car's unique VIN. The system shall implement validation to check for VIN format (typically 17 characters alphanumeric).

◦ 3.2.1.1.1.3 **Processing:**

1. Upon form submission, the system shall validate all entered data for completeness and format compliance according to the specifications mentioned in section 3.2.1.2 Inputs.
2. If any validation errors occur, the system shall display clear and specific error messages to the user, indicating which fields contain errors and the nature of the error (e.g., "Invalid year format", "License plate number does not meet country standards").
3. If all data is valid, the system shall attempt to retrieve additional car information (e.g., engine specifications, standard maintenance schedules) using an external API. The specific API to be used and its data parameters shall be defined in a separate document.
4. The system shall store the user-provided car details and any retrieved information from the external API (if successful) in the system's database.
5. The system shall generate a unique identifier for the newly created car profile.

◦ 3.2.1.1.1.4 **Outputs:**

- Upon successful car profile creation, the system shall display a confirmation message to the user, informing them that the car has been added.
- The system shall automatically navigate the user to the newly created car profile within the Car Station interface. This profile shall display the entered information and any retrieved data from the external API (if applicable).

◦ 3.2.1.1.1.5 Error Handling:

- In case of validation errors (section 3.2.1.3.2), the user shall be provided with the opportunity to correct the errors and resubmit the form.

- If the external API call for car information fails (section 3.2.1.3.3), the system shall store the user-provided details and provide the user with the following options:
- Proceed with creating the car profile without additional information.
- Manually enter missing information about the car (e.g., engine specifications).

3.2.1.1.1.2 Add Car through taking pictures of it (see later sections for clarity) #2

- 3.2.1.1.1.2.1 **Introduction:**
  - This section details the requirements for adding a car profile using the image recognition feature within the Car Station.
- 3.2.1.1.1.2.2 **Inputs:**
  - The system shall accept multiple images (at least two recommended) of the car captured using the device camera. The system shall specify supported image file formats (e.g., JPEG, PNG).
- 3.2.1.1.1.2.3 **Processing:**
  - The system shall utilize an image recognition service (details on the service to be specified in a separate document) to extract car make, model, and year from the uploaded images.
  - The system shall employ algorithms to assess the quality and clarity of the uploaded images to ensure a higher success rate for car identification.
  - Based on the extracted information, the system may attempt to retrieve additional car details (e.g., engine specifications) using an external API (refer to Section 3.2.1.1.1.3 for details on external API).
- 3.2.1.1.2.4 **Outputs:**
  - If the image recognition is successful:
    - The system shall display the extracted car information (make, model, year) to the user for confirmation.
      - Upon user confirmation, the system shall proceed to store the details in the database (Section 3.2.1.1.2.5.1).
  - If the image recognition fails to identify the car:
    - The system shall display a message to the user informing them that the car could not be identified from the images.
    - The system shall provide the user with the option to:
      - Retry capturing images with improved clarity.
      - Manually enter the car details.
  - 3.2.1.1.2.5 **Error Handling:**
  - **Image Upload Errors:**
    - If the user attempts to upload unsupported file formats or exceeds image size limitations, the system shall display an error message specifying the issue and allowed formats/sizes.
  - **Image Recognition Failure:**
    - If the image recognition service fails to identify the car due to poor image quality or other factors, the system shall handle this as described in Section 3.2.1.1.2.4.2.
  - **External API Failure (Optional):**

1. If the attempt to retrieve additional car details using the external API fails (refer to Section 3.2.1.1.1.3 for details), the system shall store the user-confirmed car information (make, model, year) and proceed with car profile creation. The user may be informed about the missing information and the option to enter it manually.

3.0.1.1.1.3 Add time of regular maintenance #3

- **3.0.1.1.1.3.1 Introduction:**
  - This section details the requirements for adding a user-specified time interval for regular maintenance reminders within the Car Station. This functionality empowers users to proactively schedule car maintenance based on their car's needs and preferences, promoting preventative care and potentially extending vehicle lifespan.
- **3.0.1.1.1.3.2 Inputs:**
  - The system shall provide a user interface element for users to specify the desired time interval between maintenance reminders. Here are two options to consider:
    - **User-friendly Interface:**
      - Pre-defined options: Offer a set of common time intervals displayed prominently (e.g., 3 months, 6 months, 1 year) to simplify selection for users.
      - Custom value input: Allow users to enter a specific time interval (e.g., in months) through a dedicated field for greater flexibility.
- **3.0.1.1.1.3.3 Processing:**
  - Store the user-provided time interval for regular maintenance reminders in the car's profile within the database (Section 3.2.1.1.2.5.1).
  - Utilize this stored interval to trigger automated notifications reminding the user about upcoming maintenance. This notification system can be integrated with the Car Station's notification center or leverage push notifications on mobile devices.
- **3.0.1.1.1.3.4 Outputs:**
  - Upon successful addition of the time interval, the system shall display a confirmation message to the user. This message should clearly inform them that the chosen reminder frequency has been added to their car profile (e.g., "Maintenance reminders set to notify you every 6 months").

- **3.0.1.1.1.3.5 Error Handling:**

  Invalid Input:

  The system shall implement safeguards to handle potential errors during user input:
  - If the user enters a custom value that falls outside the allowed range (e.g., negative values), the system shall display an error message specifying the issue and providing guidance (e.g., "Please enter a positive value for the reminder interval").
  - The system may define a minimum and/or maximum allowed value for custom intervals to ensure reminders remain practical (e.g., minimum 1 month, maximum 2 years).

  This section incorporates the following aspects based on the Rabbeshly Station project:

3.0.1.1.1.4 Add Time and Cost of Heavy Maintenance #4

- **3.0.1.1.1.4.1 Introduction:**
  This section details the requirements for adding user-specified information about heavy maintenance, including the estimated time frame and associated costs. This functionality allows users to track and plan for major car repairs or servicing that require significant time and resources.
- **3.0.1.1.1.4.2 Inputs:**
  The system shall provide user interface elements for users to specify:
  - **Estimated Time:** This could be a text field where users can enter a duration (e.g., days, weeks) or a calendar date picker for scheduled maintenance.
  - **Estimated Cost:** This could be a text field where users can enter a monetary value. The system may offer an optional field for currency selection.

- **3.0.1.1.1.4.3 Processing:**
  1. Store the user-provided estimated time and cost of heavy maintenance in the car's profile within the database (Section 3.2.1.1.2.5.1).
  2. This information can be used for various purposes, such as:
     - Displaying upcoming heavy maintenance reminders alongside regular maintenance reminders.
     - Factoring in estimated costs when calculating total maintenance costs (Section 3.0.1.1.1.12).
- **3.0.1.1.1.4.4 Outputs:**
  Upon successful addition of the heavy maintenance details, the system shall display a confirmation message to the user. This message should clearly state that the estimated time and cost have been added to the car's profile (e.g., "Heavy maintenance information added: Estimated time - 2 weeks, Estimated cost - $1,000").
- **3.0.1.1.1.4.5 Error Handling:**
  - **Invalid Input:**
    - If the user enters a negative value for estimated time or cost, the system shall display an error message prompting them to enter a valid positive value.
    - The system may define a reasonable range for estimated time (e.g., minimum 1 day, maximum 1 year) to avoid unrealistic entries.

3.0.1.1.1.5 Add Time and Cost of Precautionary Maintenance #5

- **3.0.1.1.1.5.1 Introduction:**
  This section details the requirements for adding user-specified information about precautionary maintenance, including the estimated time frame and associated costs.
  This functionality allows users to proactively plan for preventative servicing that helps maintain the car's condition and potentially avoid future issues.
- **3.0.1.1.1.5.2 Inputs:**
  - **Estimated Time:** Text field or calendar date picker for users to enter the anticipated duration.
  - **Estimated Cost:** Text field with optional currency selection for users to enter the expected cost.
- **3.0.1.1.1.5.3 Processing:**
  - Store the user-provided estimated time and cost in the car's profile database (Section 3.2.1.1.2.5.1).

- Utilize this information for:
- Displaying upcoming precautionary maintenance reminders.
- Factoring into total maintenance cost calculations (Section 3.0.1.1.1.12).

- **3.0.1.1.1.5.4 Outputs:**
  - Confirmation message upon successful addition, similar to Section 3.0.1.1.1.4.4.
- **3.0.1.1.1.5.5 Error Handling:**
  - Implement error handling for invalid user input, following the guidelines in Section 3.0.1.1.1.4.5.

3.0.1.1.1.6 Add Time and Cost of Crash Maintenance #6

- **3.0.1.1.1.6.1 Introduction:**

  This section details the requirements for adding user-specified information about crash maintenance, including the estimated time for repairs and associated costs. This functionality helps users track repairs after accidents.

- **3.0.1.1.1.6.2 Inputs:**
  - **Estimated Time:** Text field or calendar date picker for users to enter the repair duration.
  - **Estimated Cost:** Text field with optional currency selection for users to enter the expected cost.
- **3.0.1.1.1.6.3 Processing:**
  - Store the user-provided estimated time and cost of crash maintenance in the car's profile within the database (Section 3.2.1.1.2.5.1).
  - Optionally, the system may allow users to:
  - Upload pictures of the crash damage to document the incident (consider integration with Section 3.0.1.1.1.8 Add Car additional pictures).
  - Link to a police report or insurance claim (if applicable).
- **3.0.1.1.1.6.4 Outputs:**
  - Upon successful addition of the crash maintenance details, the system shall display a confirmation message to the user. This message should clearly state that the estimated time and cost have been added to the car's profile (e.g., "Crash maintenance information added: Estimated time - 1 month, Estimated cost - $5,000").
- **3.0.1.1.1.6.5 Error Handling:**
  - Follow the error handling guidelines from Section 3.0.1.1.1.4.5 for invalid input (estimated time, cost).

3.0.1.1.1.7 Add Car Specs #7

- **3.0.1.1.1.7.1 Introduction:**
  - This section details the requirements for adding car specifications to the car profile within the Car Station. This information can be beneficial for users to track car details and for the system to provide relevant maintenance suggestions or connect users with compatible workshops.
- **3.0.1.1.1.7.2 Inputs:**
  - The system shall provide user interface elements for users to enter various car specifications, potentially including:
  - Engine type (e.g., ga soline, diesel, hybrid)
  - Engine displacement (e.g., liters)
  - Transmission type (e.g., automatic, manual)

- Drive type (e.g., front-wheel drive, rear-wheel drive, all-wheel drive)
- Fuel efficiency (e.g., miles per gallon)
- Seating capacity

- **3.0.1.1.1.7.3 Processing:**
  - Store the user-entered car specifications in the car's profile within the database (Section 3.2.1.1.2.5.1).
  - This data can be used for various purposes, such as:
  - Displaying car specifications on the user's car profile page.
  - Filtering workshop search results based on compatibility with the car's specifications (assuming workshop data includes compatible car types).
  - Providing targeted maintenance suggestions based on car type and specifications (referencing Data Station - Section 3.0.1.1.2).
- **3.0.1.1.1.7.4 Outputs:**
  - Upon successful addition of car specifications, the system shall display a confirmation message to the user, informing them that the details have been saved to their car profile.
- **3.0.1.1.1.7.5 Error Handling:**
  - Ensuring valid data types are entered (e.g., numbers for engine displacement, fuel efficiency).
  - Defining reasonable value ranges for certain specifications (e.g., minimum and maximum engine displacement based on common car models).

3.0.1.1.1.8 Add Car Additional Pictures #8

- **3.0.1.1.1.8.1 Introduction:**
  - This section details the requirements for adding additional pictures of the car to enhance the car profile within the Car Station. This functionality allows users to showcase their car and potentially provide more details for reference.
- **3.0.1.1.1.8.2 Inputs:**
  - The system shall provide a user interface element, such as a file upload button or drag-and-drop zone, to allow users to select and upload multiple pictures of their car.
- **3.0.1.1.1.8.3 Processing:**
  - Validate the uploaded pictures, ensuring they are in supported image formats (e.g., JPEG, PNG) and within a reasonable size limit to optimize storage and performance.
  - Store the uploaded pictures securely within the system, associating them with the specific car profile in the database.
- **3.0.1.1.1.8.4 Outputs:**
  - Upon successful upload, the system shall provide visual confirmation to the user, such as displaying thumbnails or a success message indicating the number of pictures added.
- **3.0.1.1.1.8 Error Handling:**
  - The system shall display informative error messages for upload failures, specifying the issue (e.g., unsupported format, exceeding size limit).
  - The system may offer options to retry uploading with valid pictures or suggest image resizing tools.

3.0.1.1.1.9 Add Car Special Specs #9

- **3.0.1.1.1.9.1 Introduction:**
  This section details the requirements for adding user-specified details about car modifications or special features beyond the standard specifications (Section 3.0.1.1.1.7). This functionality allows users to personalize their car profiles and potentially connect with enthusiasts with similar car interests.
- **3.0.1.1.1.9.2 Inputs:**
  The system shall provide a text field or a dedicated section for users to enter information about car special specs. This could include details like:
  - Aftermarket parts (e.g., performance upgrades, audio systems)
  - Unique features (e.g., sunroofs, custom paint jobs)
  - Any other relevant modifications
- **3.0.1.1.1.9.3 Processing:**
  - The system shall store the user-entered car special specs in the car's profile within the database (Section 3.2.1.1.2.5.1). This information can be used for:
  - Displaying special specs on the car profile page.
  - Potentially filtering search results within the Car Station feed (Section 3.0.1.1.3) based on user interests in specific car modifications.
- **3.0.1.1.1.9.4 Outputs:**
  Upon successful addition of car special specs, the system shall display a confirmation message to the user, informing them that the details have been saved to their car profile.
- **3.0.1.1.1.9.5 Error Handling:**
  The system may not require specific error handling for this section as it deals with free-form text input. However, the system could consider:
  - Setting a character limit to prevent excessively long entries.
  - Offering suggestions for common car modifications based on user input (optional, leveraging Data Station - Section 3.0.1.1.2).

3.0.1.1.1.10 Add Persistent Issues or Accidents #10

- **3.0.1.1.1.10.1 Introduction:**
  This section details the requirements for adding user-reported persistent car issues or past accidents to the car profile. This functionality allows users to track recurring problems and potentially connect with workshops specializing in resolving those issues.
- **3.0.1.1.1.10.2 Inputs:**
  The system shall provide a user interface element for users to report persistent issues or accidents. This could involve:
  - Text field for describing the issue or accident
  - Dropdown menu or selectable options for categorizing the issue (e.g., engine problems, electrical issues, collision damage)
  - Optional upload functionality for pictures related to the issue/accident (consider integration with Section 3.0.1.1.1.8 Add Car additional pictures)
- **3.0.1.1.1.10.3 Processing:**
  - Store the user-reported persistent issue or accident details in the car's profile within the database (Section 3.2.1.1.2.5.1).
  - This information can be used for:
  - Displaying a history of reported issues on the car profile page.

        - Potentially recommending workshops that specialize in repairs relevant to the reported issues (assuming workshop data includes repair expertise).
- **3.0.1.1.1.10.4 Outputs:**
    - Upon successful addition of the persistent issue or accident report, the system shall display a confirmation message to the user, informing them that the details have been saved to their car profile.
- **3.0.1.1.1.10.5 Error Handling:**

    The system may not require extensive error handling, but it could:
    - Ensure a minimum character limit in the text field to encourage users to provide some description.

3.0.1.1.1.11 Add Car Chassis/VIN Number #11

- **3.0.1.1.1.11.1 Introduction:**

  This section details the requirements for adding the car's Chassis or Vehicle Identification Number (VIN) to the car profile. The VIN is a unique identifier for the car and can be valuable for retrieving detailed car information and facilitating maintenance tasks.
- **3.0.1.1.1.11.2 Inputs:**

  The system shall provide a dedicated text field for users to enter the car's Chassis or VIN number. The system may:
    - Implement format validation to ensure the entered VIN adheres to the standard 17-character alphanumeric format.
    - Offer an information icon or tooltip explaining the importance of the VIN and where to locate it on the car (e.g., driver's side door jamb).
- **3.0.1.1.1.11.3 Processing:**
    - Store the user-entered VIN securely within the car's profile in the database (Section 3.2.1.1.2.5.1).
    - The VIN can be used for various purposes, such as:
    - Potentially retrieving detailed car information from an external database with user consent (refer to Section 3.2.1.1.1.3 for data privacy considerations).
    - Facilitating tasks like ordering compatible spare parts or scheduling maintenance services at workshops that utilize VIN lookups for specific car models.
- **3.0.1.1.1.11.4 Outputs:**

  Upon successful addition of the VIN, the system shall display a confirmation message to the user, informing them that the VIN has been saved to their car profile.
- **3.0.1.1.1.11.5 Error Handling:**

  The system shall implement error handling for invalid VIN input:
    - If the user enters an incorrect format (less or more than 17 characters, invalid characters), the system shall display an error message prompting them to enter a valid VIN number.
    - The system may offer suggestions for correcting the format (e.g., highlighting incorrect characters).

3.0.1.1.1.12 Display Total Maintenance Costs #12

- **3.0.1.1.1.12.1 Introduction:**
  This section details the requirements for displaying the total estimated maintenance costs associated with a car profile within the Car Station. This functionality provides users with a clear overview of anticipated expenses for regular, precautionary, and potential crash maintenance (Sections 3.0.1.1.1.4, 3.0.1.1.1.5, 3.0.1.1.1.6).
- **3.0.1.1.1.12.2 Inputs:**
  The system shall not require any user input for this functionality.
- **3.0.1.1.1.12.3 Processing:**
  1. Access the car's profile in the database (Section 3.2.1.1.2.5.1).
  2. Retrieve the estimated costs for regular maintenance intervals (Section 3.0.1.1.1.4), precautionary maintenance (Section 3.0.1.1.1.5), and crash maintenance (Section 3.0.1.1.1.6), if provided by the user.
  3. Calculate the total estimated maintenance cost by summing the retrieved values.
- **3.0.1.1.1.12.4 Outputs:**
  The system shall display the calculated total estimated maintenance cost in a clear and prominent location within the car profile interface. This could be a dedicated section labeled "Total Estimated Maintenance Cost" with a numerical value and optional currency symbol.
- **3.0.1.1.1.12.5 Error Handling:**
  The system may not encounter errors in this process, as it retrieves and sums existing data. However, if any of the cost-related sections (3.0.1.1.1.4, 3.0.1.1.1.5, 3.0.1.1.1.6) have errors due to user input, it would be reflected in the total cost calculation.

3.0.1.1.1.13 Display Car Efficiency #13

- **3.0.1.1.1.13.1 Introduction:**
  This section details the requirements for displaying the car's fuel efficiency within the car profile. This information can be beneficial for users to track their car's performance and potentially make informed decisions about fuel consumption.
- **3.0.1.1.1.13.2 Inputs:**
  The system may have two approaches for obtaining car efficiency data:
  - **User Input:** The system can provide a text field where users can enter their car's fuel efficiency (e.g., miles per gallon or liters per kilometer).
  - **External Data Retrieval (Optional):** The system can attempt to retrieve fuel efficiency data from an external database using the car's VIN number (Section 3.0.1.1.1.11) with user consent (refer to Section 3.2.1.1.1.3 for data privacy considerations). This approach requires integration with a reliable car information database.
  - **3.0.1.1.1.13.3 Processing:**
    1. If user input is available, store the entered fuel efficiency data in the car's profile within the database (Section 3.2.1.1.2.5.1).
    2. If enabled and with user consent, attempt to retrieve fuel efficiency data from an external database using the car's VIN.
    3. The system shall prioritize user-entered data over retrieved data, as user input might reflect real-world driving conditions.
    4. If both user input and retrieved data are available, the system may display them both for user reference, allowing them to compare and potentially adjust the user-entered value based on the retrieved information.

- **3.0.1.1.1.13.4 Outputs:**

  The system shall display the car's fuel efficiency data in a clear and user-friendly format within the “car profile” interface. This could be a dedicated section labeled "Fuel Efficiency" with a numerical value and the corresponding unit (e.g., "25 MPG").
- **3.0.1.1.1.13.5 Error Handling:**

  The system may encounter the following situations and handle them accordingly:
  - **Invalid User Input:** If the user enters an invalid value for fuel efficiency (e.g., negative numbers, non-numeric characters), the system shall display an error message prompting them to enter a valid value.
  - **External Data Retrieval Failure:** If the attempt to retrieve fuel efficiency data from an external database fails due to network issues or database unavailability, the system shall inform the user and display any user-entered data if available.

3.0.1.1.1.14 Display Car #14

- **3.0.1.1.1.14.1 Introduction:**

This section details the requirements for displaying the car's name or nickname within the Car Station. This functionality allows users to personalize their car profiles and easily identify their car among others.
- **3.0.1.1.1.14.2 Inputs:**

  The system shall provide a text field for users to enter a name or nickname for their car. This can be any user-preferred text to represent their car.
- **3.0.1.1.1.14.3 Processing:**
  1. Store the user-entered car name/nickname in the car's profile within the database (Section 3.2.1.1.2.5.1).
  2. This car name/nickname shall be displayed prominently throughout the car profile interface, as well as in relevant sections within the Car Station (e.g., feed posts mentioning the car).
- **3.0.1.1.1.14.4 Outputs:**

The system shall display the user-entered car name/nickname in various locations within the Car Station:
  - **Car Profile:** The car name/nickname should be displayed prominently at the top of the car profile page, easily identifiable by the user.
  - **Feed Posts & Mentions:** When a user mentions their car in a feed post or another user refers to a car in a comment, the car name/nickname shall be used for clarity and personalization.
- **3.0.1.1.1.14.5 Error Handling:**

  The system may not require specific error handling for car name/nickname input as it deals with free-form text. However, the system could:
  - Set a character limit to prevent excessively long names.

3.0.1.1.1.15 Share Update to the Feed #15

- **3.0.1.1.1.15.1 Introduction:**

This section details the requirements for users to share updates about their car to the “Car Station feed”. This functionality allows users to connect with other car enthusiasts, share experiences, and potentially seek advice from the community.

- **3.0.1.1.1.15.2 Inputs:**
  The system shall provide a dedicated interface for users to create feed posts about their car. This interface could include:
  - Text field for composing the post content.
  - Option to attach pictures or videos related to the post (consider integration with Section 3.0.1.1.1.8 Add Car additional pictures).
  - Option to tag other users or their cars within the post (relevant if user profiles and car profiles are linked).
- **3.0.1.1.1.15.3 Processing:**
  1. Store the user-created post content, including text, attached media (if any), and user tags, within the system's database.
  2. Publish the post to the user's feed and potentially the Car Station's main feed (depending on privacy settings and relevance algorithms).
- **3.0.1.1.1.15.4 Outputs:**
  Upon successful post creation, the system shall display the newly created post within the user's feed and potentially the Car Station's main feed (depending on settings).
- **3.0.1.1.1.15.5 Error Handling:**
  The system shall implement error handling for various scenarios:
  - **Empty Content:** If the user attempts to create a post with empty text content, the system shall display an error message prompting them to add some content.
  - **Large Media Files:** If uploaded media files exceed a predefined size limit, the system shall display an error message and suggest resizing or using alternative media.

3.0.1.1.2 Suggestion and tips: to be supplied by Data Station and Service Station later #16

- ## 3.0.1.1.3 Car Station Feed

3.0.1.1.3.1 Share post to other platforms #17

- **3.0.1.1.3.1.1 Introduction:**
  This section details the requirements for users to share their Car Station feed posts to other social media platforms. This functionality allows users to expand their reach and potentially connect with a wider car enthusiast audience.
- **3.0.1.1.3.1.2 Inputs:**
  The system shall provide icons or buttons that link to popular social media platforms (e.g., Facebook, Instagram) within the feed
- **3.0.1.1.3.1.3 Processing:**
  1.Depending on the chosen platform, the system might:
  - **Direct Sharing:** If the platform has a native sharing API, the system could leverage it to pre-populate the post content (text, pictures/videos) on the chosen platform for the user to edit and finalize the share.
  - **Indirect Sharing:** If direct sharing is not available, the system could provide the user with a sharable link or pre-composed message containing the post content to be copied and pasted manually into the chosen platform.
- **3.0.1.1.3.1.4 Outputs:**

The system's primary output is facilitating the user's sharing process. Depending on the chosen platform and sharing method:

- **Direct Sharing:** The system could redirect the user to the chosen platform's interface with the post content pre-populated for editing and sharing.
- **Indirect Sharing:** The system would display the sharable link or pre-composed message, allowing the user to copy and paste it manually into their preferred platform.

- **3.0.1.1.3.1.5 Error Handling:**

The system may encounter limitations depending on the chosen platform and the user's device/app configuration. Potential error handling scenarios include:

- **Platform Unavailable:** If the chosen platform's app is not installed or the user is not logged in, the system could inform the user and offer alternative sharing options (e.g., copy link).
- **Sharing API Issues:** In case of technical difficulties with the platform's sharing API, the system could fallback to indirect sharing with a sharable link.

3.0.1.1.3.2 Comment, Like and re-share on your Car Profile #18

- **3.0.1.1.3.2.1 Introduction:**

This section details the requirements for user interaction with other posts within the Car Station feed. This functionality fosters engagement and allows users to express opinions, provide support, and share posts they find relevant.

- **3.0.1.1.3.2.2 Inputs:**

The system shall provide interactive elements within the feed post interface to allow users to:

- **Leave comments:** A text field for users to compose and submit comments on a specific post.
- **Like a post:** A like button or icon for users to express appreciation or agreement with a post.
- **Re-share a post:** An option for users to share an existing post from their feed to their own profile or potentially the main Car Station feed (depending on privacy settings).

- **3.0.1.1.3.2.3 Processing:**
  - Store user comments on the corresponding post within the database.
  - Track user likes on a post and update the post's like count displayed in the feed.
  - Upon re-sharing a post, the system shall create a new post entry in the user's profile or the main feed (depending on settings), referencing the original post and potentially including a short message from the re-sharing user (optional).
- **3.0.1.1.3.2.4 Outputs:**

The system shall update the user interface of the corresponding post to reflect user interactions:

  - Display newly submitted comments chronologically below the post.
  - Update the like count to reflect the number of users who liked the post.
  - If a user re-shares a post, the re-shared post should appear in the relevant feed (user profile or main feed) with appropriate attribution to the original author.
- **3.0.1.1.3.2.5 Error Handling:**

The system may encounter minimal error handling in this section:

- **Empty Comment:** If a user attempts to submit an empty comment, the system could prompt them to enter some text before submitting.

3.0.1.1.3.3 Create new posts, add pictures from your car profile or from device, write caption, add tags #19

- **3.0.1.1.3.3.1 Introduction:**
  This section details the requirements for creating new posts within the Car Station feed, allowing users to share content, pictures, and interact with the community.
- **3.0.1.1.3.3.2 Inputs:**
  The system shall provide a user interface for creating new posts, including:
  - **Text field:** For composing the post content (caption).
  - **Image/Video selection:**
    - **Car profile pictures:** An option to select pictures from the user's car profile (referencing Section 3.0.1.1.1.8 Add Car additional pictures).
    - **Device storage:** An option to select pictures or videos from the user's device storage.
  - **Tagging:** An option to tag other users or their cars within the post (referencing user profiles and car profiles).
- **3.0.1.1.3.3.3 Processing:**
  1. Allow users to select and upload pictures/videos from their chosen source (car profile or device storage).
  2. Store the uploaded media securely within the system.
  3. Process the user-composed caption text.
  4. Associate the post with the user's profile and car profile (if a car is tagged).
  5. Store any user tags (references to other users or cars).
  6.
- **3.0.1.1.3.3.4 Outputs:**
  - Display the newly created post within the user's profile feed.
  - Depending on privacy settings and potential algorithmic filtering (referencing Section 3.0.1.1.2 Data Station), the post might also appear in the Car Station's main feed.
  - The post interface shall display the caption text, uploaded media (thumbnails for pictures, preview for videos), and user tags (clickable links to profiles or car profiles).
- **3.0.1.1.3.3.5 Error Handling:**
  The system shall implement error handling mechanisms for:
  - **Large media file size limitations:** If uploaded media exceeds a predefined size limit, the system shall display an error message and suggest resizing or using alternative media.
  - **Upload failures:** In case of technical issues during media upload, the system shall inform the user and allow them to retry.

3.0.1.1.3.4 Interact with posts from other platforms such as a Facebook Group from the same niche #20

- **3.0.1.1.3.4.1 Introduction:**

  This section explores the possibility of interacting with content from external platforms within the Car Station feed. This functionality could potentially broaden user

engagement and allow discussions to flow between the Rabbeshly Station app and other online communities.

- **3.0.1.1.3.4.2 Inputs:**
  - This section might require further investigation to determine the feasibility and user experience of direct interaction with external platforms. Potential input methods could include:
  - Designated buttons or icons within the Car Station feed to link or reference posts from external platforms (e.g., "Discuss on Facebook").
  - User-generated content that includes links or mentions of relevant posts from external platforms.
- **3.0.1.1.3.4.3 Processing:**
  - **Designated buttons/icons:** Clicking such elements could:
  - Open the linked external platform (Facebook group) within a web browser or redirect the user to download the external platform's app if not already installed.
  - Potentially display a preview or summary of the linked external post within the Car Station feed (if the external platform offers an API for such functionality).
  - **User-generated content with mentions:** The system could attempt to identify and highlight mentions of external platforms or posts within user-created content in the Car Station feed. Clicking such highlighted mentions could function similarly to designated buttons/icons.
- **3.0.1.1.3.4.4 Outputs:**
  The primary output would be facilitating user interaction with content on external platforms. Depending on the processing method:
  - The system might redirect users to the external platform itself.
  - The system could display previews or summaries of external posts within the Car Station feed (if supported by external platforms).
- **3.0.1.1.3.4.5 Error Handling:**
  - **External platform unavailable:** If a linked external platform is unavailable or the user cannot access it (e.g., login required, app not installed), the system The system could inform the user about the issue and offer alternative actions, such as copying the link for manual access later.
  - **External API limitations:** If the chosen method relies on APIs from external platforms, limitations or errors from those APIs could hinder functionality. The system should gracefully handle such errors and inform the user if attempts to display previews or summaries of external posts fail.

3.0.1.2 Shop Station Requirements:

- **3.0.1.2.1 Shop Profile:**

3.0.1.2.1.1 General Requirements of Shops:

- **3.0.1.2.1.1.1 Add Shop Details through filling a form (1.2.1) #21**
  - **3.0.1.2.1.1.1.1 Introduction:**
    This section details the requirements for shops to create and manage their profiles within the Shop Station. A well-defined shop profile allows users to discover the shop, understand its offerings, and establish trust.
  - **3.0.1.2.1.1.1.2 Inputs:**

- Shop name and logo
- Contact information (address, phone number, website)
- Shop description (including specialties and areas of expertise)
- Working hours
- Categories of products or services offered (referencing Section 3.0.1.2.1.1.3 Add Categories of products)
- Links to social media profiles (optional)

- **3.0.1.2.1.1.1.3 Processing:**
  1. Store the entered shop profile details securely within the system's database.
  2. Associate the shop profile with the user or administrator who created it.
  3. The system might perform basic validation checks on some data (e.g., phone number format, website URL validity).
- **3.0.1.2.1.1.1.4 Outputs:**

  Upon successful profile creation, the system shall display the shop profile within the Shop Station. The profile should include the entered details in a user-friendly format, allowing potential customers to learn about the shop.

- **3.0.1.2.1.1.2 Add Product/Car (sale price, name and cost) (1.2.3) #22**
  - **3.0.1.2.1.1.2.1 Introduction:**

    This section details the requirements for shops to add products or cars to their Shop Station profiles. This functionality allows shops to showcase their inventory and enable users to browse and potentially purchase these items.
  - **3.0.1.2.1.1.2.2 Inputs:**

    The system shall provide an interface for shops to add products or cars to their profiles. This interface could include:
    - Product/Car name and description
    - Category selection (referencing Section 3.0.1.2.1.1.3 Add Categories of products)
    - Sale price
    - Cost price (optional, for internal shop management)
    - Stock quantity (number of units available)
    - Pictures of the product/car (referencing Section 3.0.1.2.1.1.6 Respond to customer inquiries of parts and prices)
  - **3.0.1.2.1.1.2.3 Processing:**
    1. Store the entered product/car details within the shop's profile in the database.
    2. Associate the product/car with the corresponding shop profile.
    3. Manage product/car stock quantity.
  - **3.0.1.2.1.1.2.4 Outputs:**
    - The system shall display the added product/car within the shop's profile on the Shop Station. The product/car listing should include details like name, description, category, price, and pictures. Stock quantity information might be displayed or hidden depending on shop preferences.
    - The system might also integrate product/car listings into search functionalities within the Shop Station, allowing users to discover products or cars based on their needs.

- **3.0.1.2.1.1.3 Add Categories of products (1.2.3) #23**
  - **3.0.1.2.1.1.3.1 Introduction:**
    This section details the requirements for defining product categories within the Shop Station. A well-organized category system helps users navigate the shops and find products efficiently.
  - **3.0.1.2.1.1.3.2 Inputs:**
    The system might provide two approaches for managing product categories:
    - **Predefined categories:** The system could offer a pre-defined set of general product categories relevant to the car enthusiast market (e.g., Engine Parts, Performance Upgrades, Interior Accessories). Shops can utilize these categories when adding products to their profiles (referencing Section 3.0.1.2.1.1.2 Add Product/Car).
    - **Shop-specific categories (optional):** In addition to predefined categories, the system could allow shops to define their own custom categories for more granular product organization within their shops (e.g., Performance Camshafts, Aftermarket Lighting).
  - **3.0.1.2.1.1.3.3 Processing:**
    - Manage the predefined category list and ensure its consistency across the Shop Station.
    - Allow shops to select predefined categories when adding products.
    - Optionally, allow shops to create and manage their own custom categories within their profiles.
  - **3.0.1.2.1.1.3.4 Outputs:**
    - Display predefined categories as options for shops when adding products.
    - If enabled, allow shops to define and manage custom categories within their shop profiles.
    - Product listings within the Shop Station should utilize categories for organization, potentially using a combination of predefined and shop-specific categories (depending on shop preferences).

3.0.1.2.1.1.4 Import spreadsheet to add past sale data (1.2.3) #24

- **3.0.1.2.1.1.4.1 Introduction:**
  This section details the functionality for shops to import past sales data from spreadsheets into their Shop Station profiles. This allows shops to populate their product listings efficiently and potentially gain insights from historical sales information.
- **3.0.1.2.1.1.4.2 Inputs:**
  The system shall provide a mechanism for shops to upload spreadsheet files containing product data. This functionality could include:
  - A designated upload button or area within the shop profile management interface.
  - Support for common spreadsheet file formats (e.g., CSV, XLSX).
  - Template download option: The system could offer a downloadable template spreadsheet with pre-defined columns for product details (e.g., product name, category, price, stock quantity) to ensure compatibility during import.

- **3.0.1.2.1.1.4.3 Processing:**
  1.Allow shops to upload spreadsheet files containing product data.
  2.Perform data validation on the uploaded spreadsheet to ensure it adheres to the expected format (e.g., matching column headers with product details).

3.If validation is successful, the system shall extract product information from the spreadsheet and:
    - Create new product entries within the shop's profile for each data row.
    - Update existing product entries if matching identifiers (e.g., product name) are found in the spreadsheet data.

- **3.0.1.2.1.1.4.4 Outputs:**
    - Provide feedback to the shop regarding the upload process, including success messages or error reports if data validation fails.
    - If the import is successful, the system shall update the shop's profile with the imported product data, potentially reflecting newly added products or updated information for existing ones.

3.0.1.2.1.1.5 Export sales and storage reports (1.2.3) #25

- **3.0.1.2.1.1.5.1 Introduction:**
  This section details the functionality for shops to export sales and storage data from their Shop Station profiles. This allows shops to analyze sales trends, manage inventory, and potentially generate reports for accounting purposes.
- **3.0.1.2.1.1.5.2 Inputs:**
  The system shall provide options for shops to initiate data exports:
    - **Report selection:** The system could offer options for exporting different types of reports, such as:
    - Sales reports by product, category, or date range.
    - Inventory reports showing stock levels and potential low-stock alerts.
    - **Export format:** The system could allow shops to choose the desired export format for the report data, such as CSV or XLSX spreadsheets.
- **3.0.1.2.1.1.5.3 Processing:**
    1. Based on the shop's chosen report type and date range (if applicable), generate a report containing relevant sales and/or storage data.
    2. Prepare the report data in the chosen export format (e.g., CSV or XLSX).
- **3.0.1.2.1.1.5.4 Outputs:**
    - Provide the shop with the generated report as a downloadable file.
    - The report should be formatted for easy import into external spreadsheet applications or analysis tools.

3.0.1.2.1.1.6 Respond to customer inquiries of parts and prices (1.2.3) #26

- **3.0.1.2.1.1.6.1 Introduction:**
  This section highlights the importance of shops being able to respond to customer inquiries about parts and prices within the Shop Station. Effective communication builds trust and facilitates sales.
- **3.0.1.2.1.1.6.2 Not directly addressed by functionalities:**
  While the system cannot directly handle customer inquiries, it can provide the foundation for shops to establish communication channels:
    - **Shop contact information:** The shop profile creation process (Section 3.0.1.2.1.1.1 Add Shop Details) should capture contact information (phone number, email) that allows customers to reach the shop directly.
    - **Optional messaging system (future consideration):** As a potential future enhancement, the Shop Station could explore integrating a messaging system to facilitate direct communication between shops and customers within the platform.

3.0.1.2.1.1.7 Make offers and flash sales (1.2.3) #27

- **3.0.1.2.1.1.7.1 Introduction:**
  This section details the requirements for shops to create and promote special offers and flash sales within the Shop Station. This functionality allows shops to attract customers with discounts and temporary price reductions, potentially boosting sales.
- **3.0.1.2.1.1.7.2 Inputs:**
  The system shall provide an interface for shops to define and manage offers and flash sales. This interface could include:
  - Selection of products or categories to include in the offer.
  - Discount type (percentage or fixed amount).
  - Discount value.
  - Start and end date/time for the offer (flash sale duration).
  - Optional title or description for the offer to be displayed to customers.
- **3.0.1.2.1.1.7.3 Processing:**
  1. Store the defined offer details within the shop's profile and associate them with the selected products or categories.
  2. During the offer validity period:
     - The system shall apply the defined discount to the product prices displayed to customers within the Shop Station.
     - The system might visually highlight offer items on the shop's profile or within search results to grab user attention.
- **3.0.1.2.1.1.7.4 Outputs:**
  - Display offer details and discounted prices to customers browsing the shop's profile or searching for products.
  - Offer listings could be included in a dedicated "Offers" section within the Shop Station (if applicable).
  - Shops can manage and monitor the status of their active offers within their shop profile interface.

3.0.1.2.1.1.8 Add employees (1.2.1) #28

- **3.0.1.2.1.1.8.1 Introduction:**
  This section details the requirements for shop owners or administrators to add employees to their Shop Station profiles. This allows for managing staff accounts and potentially assigning different permission levels within the shop's operations.
- **3.0.1.2.1.1.8.2 Inputs:**
  The system shall provide an interface for shop administrators to add employees. This interface could include:
  - Employee information: name, email address, and potentially additional relevant details (phone number, role).
  - Password creation for the new employee account (secure password management practices are crucial).
- **3.0.1.2.1.1.8.3 Processing:**
  1. Create a new employee account within the Shop Station linked to the specific shop.
  2. Store employee information securely within the system's database.
  3. Assign a default permission level to the new employee account (referencing Section 3.0.1.2.1.1.9 Decide Employees permissions).
- **3.0.1.2.1.1.8.4 Outputs:**
  - Inform the shop administrator about the successful creation of the new employee account.

- The newly created employee account will require separate login credentials for the employee to access the Shop Station with their assigned permissions.

3.0.1.2.1.1.9 Decide Employees permissions in shop system (1.2.1) #29

- **3.0.1.2.1.1.9.1 Introduction:**
  This section details the requirements for defining permission levels for employee accounts within a shop on the Shop Station. A granular permission system allows shop owners to control access to sensitive functionalities and data.
- **3.0.1.2.1.1.9.2 Inputs:**
  - The system shall provide options for shop administrators to define and manage permission levels for employee accounts. This could involve:
  - Predefined permission levels (e.g., "Sales Staff", "Inventory Manager", "Shop Administrator") with varying access rights.
  - The ability to create custom permission levels with specific functionalities enabled/disabled for each level (e.g., allowing inventory management but restricting access to financial reports).
- **3.0.1.2.1.1.9.3 Processing:**
  - Manage predefined permission levels and their associated functionalities.
  - Allow shop administrators to assign permission levels to individual employee accounts.
- **3.0.1.2.1.1.9.4 Outputs:**
  - Provide a clear overview of permission levels and their associated functionalities within the ` shop profile management interface.
  - Shop administrators can assign appropriate permission levels to employees based on their roles and responsibilities within the shop.

3.0.1.2.1.1.10 Track productivity of each employee (1.2.3) #30

- **3.0.1.2.1.1.10.1 Introduction:**
  This section explores the possibility of tracking employee performance within the Shop Station. This data can be valuable for shop owners or administrators to assess employee contributions and potentially optimize shop operations.
- **3.0.1.2.1.1.10.2 Inputs:**
  - This section requires further definition based on the desired level of employee activity tracking. Potential input sources could include:
  - **System generated data:** The system could track actions performed by employees within the Shop Station, such as:
    - Processing orders (including fulfillment status)
    - Responding to customer inquiries
    - Adding or updating products
    - Managing offers and promotions
  - **Manual input (optional):** Shop administrators might have the option to add manual feedback or performance ratings for employees.
- **3.0.1.2.1.1.10.3 Processing:**
  The system's processing would depend on the chosen data collection methods:
  - **System generated data:** The system would automatically track relevant employee actions within the Shop Station and store this data securely.
  - **Manual input:** The system would allow shop administrators to submit performance ratings or feedback for employees, potentially associating it with specific timeframes or actions.
- **3.0.1.2.1.1.10.4 Outputs:**

The system's output would depend on the desired level of detail for employee performance tracking:

- **Basic reporting:** The system could generate reports summarizing employee activity within the Shop Station, potentially showing the number of processed orders, interactions with customers, or product management actions for each employee during a chosen period.
- **Advanced analytics (optional):** As a future enhancement, the system could explore more advanced analytics, such as associating sales figures with specific employees or analyzing customer satisfaction based on interactions handled by different employees.

3.0.1.2.1.1.11 Add Discount percent/flat amount to the purchase (1.2.3) #31

- **3.0.1.2.1.1.11.1 Introduction:**
  This section details the functionality for shops to offer discounts (percentage or flat amount) during the checkout process within the Shop Station. This allows shops to provide additional promotions or incentives to customers at the point of purchase.
- **3.0.1.2.1.1.11.2 Inputs:**
  The system shall provide options for shops to define discount rules within their shop profile:
  - **Discount type:** Choose between a percentage discount or a fixed discount amount.
  - **Discount value:** Enter the specific percentage or fixed amount to be applied.
  - **Discount application:** Define how the discount should be applied:
  - **Entire purchase:** Apply the discount to the total order value.
  - **Specific products or categories:** Apply the discount only to selected products or product categories within the shop.

  1. **Optional trigger (future consideration):** As a potential future enhancement, the system could allow defining triggers for automatic discount application (e.g., minimum purchase amount required to qualify for the discount).
- **3.0.1.2.1.1.11.3 Processing:**
  - Store the defined discount rule within the shop's profile.
  - During checkout, the system shall:
    - Identify if any discount rules are applicable based on the customer's cart contents.
    - Apply the defined discount to the order total according to the chosen discount type and application method.
- **3.0.1.2.1.1.11.4 Outputs:**
  - Clearly display the applied discount and the discounted order total during the checkout process for customer transparency.
  - The final order confirmation should reflect the discounted price.

3.0.1.2.1.2 Auto Parts Shop:

3.0.1.2.1.2.1 Add shelf position of the products (1.2.3) #32

- **3.0.1.2.1.2.1.1 Introduction:**
  This section explores the functionality for auto parts shops to manage shelf locations for their products within the Shop Station. This information could be beneficial for shops with physical stores and for customers who might prefer in-store pickup.
- **3.0.1.2.1.2.1.2 Inputs:**
  The system could provide an optional feature for auto parts shops to define shelf locations for their products:
  - During product creation or editing (referencing Section 3.0.1.2

- During product creation or editing (referencing Section 3.0.1.2.1.1.2 Add Product/Car), auto parts shops could have an additional field to specify the shelf location for the product. This field could involve:
- A text field for a simple location description (e.g., "Aisle 3, Shelf 2").
- A visual selection method (optional): The system could potentially explore a visual representation of the shop's layout (e.g., a shelf diagram) where shop administrators can click on specific locations to assign them to products.

- **3.0.1.2.1.2.1.3 Processing:**
  - If shelf location information is provided, store it alongside the product data within the shop's profile.

- **3.0.1.2.1.2.1.4 Outputs:**

  The system's output regarding shelf locations depends on its implementation:
  - **Basic text display (optional):** If shelf locations are stored as text descriptions, the system could display them alongside product information within the shop profile or search results (clearly indicating it's a shelf location, not part of the product description).
  - **Visual representation (optional):** If a visual shop layout is implemented, the system could highlight product locations on the diagram when a specific product is viewed within the shop profile.

3.0.1.2.1.2.2 Respond to customer inquiries of parts and prices (1.2.3) #33

- **3.0.1.2.1.2.3.1 Introduction:**

This section details the functionalities for auto parts shop owners to respond to customer inquiries about parts and prices within the Rabbeshly Station system. This feature allows shop owners to efficiently manage customer interactions and potentially convert inquiries into sales.

- **3.0.1.2.1.2.3.2 Inputs:**
  - **Customer Inquiry:** The shop owner receives a notification or message from a customer through the Rabbeshly Station messaging system. The inquiry might include:

    The name or description of the car part the customer is interested in.

    Any additional details provided by the customer (e.g., vehicle model, budget).
- **3.0.1.2.1.2.3.3 Processing:**
  1. The shop owner reviews the customer inquiry and identifies the relevant car part(s) within their inventory.
  2. The shop owner can access the customer's profile (optional) to understand their purchase history or preferences (if privacy settings allow).
- **3.0.1.2.1.2.3.4 Outputs:**
  - The shop owner can respond to the customer's inquiry through the Rabbeshly Station messaging system, including:
    - Confirmation of part availability and stock level.
    - The part's price and any applicable discounts or promotions.
    - Additional information about the part (e.g., specifications, warranty).
    - High-quality images of the part (if available).
    - A link to the part's details page on the marketplace (optional).

- The shop owner can initiate a negotiation process with the customer regarding price or other terms (optional).
- The shop owner can mark the inquiry as "resolved" after completing the interaction with the customer.

3.0.1.2.1.2.3 Manage storage and capital of products stored (1.2.3) #34

- **3.0.1.2.1.2.3.1 Introduction:**

  This section details the requirements for auto parts shops to manage their product inventory and associated costs within the Shop Station. This functionality is crucial for shops to track stock levels, maintain accurate pricing, and potentially optimize their stock management.
- **3.0.1.2.1.2.3.2 Inputs:**

  The system shall provide functionalities for shops to manage product inventory and costs:
  - **Stock level updates:** The system should allow shops to update stock quantities for products within their profile. This could be done through dedicated buttons (e.g., "Add Stock", "Remove Stock") or by specifying a new quantity value.
  - **Cost price management:** Shops can define and update the cost price of each product (referencing Section 3.0.1.2.1.1.2 Add Product/Car). This information is for internal shop management and might not be directly visible to customers.
- **3.0.1.2.1.2.3.3 Processing:**
  - Track and store product stock levels based on shop updates.
  - Maintain the association between products and their defined cost prices.
  - The system might implement safeguards to prevent negative stock quantities (e.g., warnings or preventing stock reduction beyond available items).
- **3.0.1.2.1.2.3.4 Outputs:**
  - Display current stock levels for each product within the shop's profile (potentially indicating low stock if predefined thresholds are reached).
  - Cost prices are for internal shop management and wouldn't be directly visible to customers on the Shop Station.

3.0.1.2.1.3 Car Dealership shop:

3.0.1.2.1.3.1 Add car make, model and location of shipping (1.2.3) #35

- **3.0.1.2.1.3.1.1 Introduction:**

  This section details the requirements for car dealerships to list their cars within the Shop Station. The system should capture essential details to allow users to browse and potentially purchase vehicles.
- **3.0.1.2.1.3.1.2 Inputs:**

  The system shall provide an interface for car dealerships to add car listings to their Shop Station profiles. This interface could include:
  - **Car details:**
  - Car make and model year
  - Mileage

        - Additional details relevant to car dealerships (e.g., body type, transmission type, engine specifications)
        - **Car condition:** Options to specify the car's condition (e.g., new, used, certified pre-owned).
        - **Car pictures:** The system should allow uploading multiple pictures showcasing the car's exterior, interior, and any relevant features (referencing Section 3.0.1.1.3.3 Create new posts, add pictures from your car profile or from device).
        - **Car location (optional):** Depending on the dealership's setup, the system could offer options for specifying the car's location:
        - **Physical dealership location:** If the dealership has a physical location, they could specify the address or city where the car is available for viewing or pickup.
        - **Virtual showroom (future consideration):** As a potential future enhancement, the system could explore a virtual showroom concept where dealerships can showcase their car inventory in a 3D or interactive environment.
        - **Price:** The system should allow dealerships to define the asking price for each car.
- **3.0.1.2.1.3.1.3 Processing:**
    1. Store the entered car details securely within the dealership's profile on the Shop Station.
    2. Associate car pictures with their corresponding car listings.
    3. Manage car location information based on the chosen option (physical address or virtual showroom).
    4. Display car listings within the Shop Station, potentially categorized by car make, model, or other relevant criteria.
- **3.0.1.2.1.3.1.4 Outputs:**
    - The system shall:
    - Display car listings within the dealership's Shop Station profile, showcasing car details, pictures, location information (if provided), and price.
    - The system might integrate car listings into search functionalities within the Shop Station, allowing users to discover cars based on their preferences (make, model, price range, etc.).

3.0.1.2.1.3.2 Manage financing options (1.2.3) #36

- **3.0.1.2.1.3.2.1 Introduction:**

    This section explores the possibility of car dealerships offering financing options through the Shop Station. Simplifying the car buying process by integrating financing could be beneficial for both dealerships and customers.
- **3.0.1.2.1.3.2.2 Inputs:**

    This section requires further definition based on the desired level of financing integration:
    - **Informational display:** The system could allow dealerships to display basic information about available financing options (e.g., collaborating lenders, loan terms). This might involve static content or links to the dealership's website with detailed financing information.
    - **Direct integration (future consideration):** As a complex future enhancement, the system could explore integrating with financing partners, allowing users to pre-qualify for loans or initiate the financing application process directly within the Shop Station.
- **3.0.1.2.1.3.2.3 Processing:**

- **Informational display:** The system would store and display the provided financing information within the dealership's profile.
- **Direct integration (future consideration):** A more complex integration with financing partners would require establishing secure connections and data exchange protocols to facilitate loan applications within the Shop Station.

- **3.0.1.2.1.3.2.4 Outputs:**
  - The system's output would depend on the chosen approach:
  - **Informational display:** The system would display financing information for potential customers to review within the dealership's profile.
  - **Direct integration (future consideration):** In a future scenario with direct financing integration, the system could provide a user interface for loan applications and potentially display pre-approval information or estimated loan terms.

3.0.1.2.1.3.3 Track test drive requests (1.2.3) #37

- **3.0.1.2.1.3.3.1 Introduction:**

  This section details the functionality for car dealerships to track test drive requests from potential customers through the Shop Station. This allows dealerships to manage customer interactions and potentially schedule test drives.
- **3.0.1.2.1.3.3.2 Inputs:**

  The system could provide two options for requesting test drives:
  - **Dedicated request form:** The dealership profile could include a dedicated form where users can submit test drive requests. This form might capture user information (name, contact details) and preferred car details (specific model or selection criteria).
  - **Contact information:** Users could directly contact the dealership using the provided contact information within the shop profile (referencing Section 3.0.1.2.1.1.1 Add Shop Details).
- **3.0.1.2.1.3.3.3 Processing:**
  - Capture test drive requests submitted through the dedicated form, including user information and car preferences.
  - If users choose to contact the dealership directly, the system wouldn't be directly involved in capturing that interaction, but the dealership's contact information should be readily available within the profile.
- **3.0.1.2.1.3.3.4 Outputs:**
  - Display submitted test drive requests within the dealership's profile management interface, allowing them to review and manage these requests (e.g., contact requester, schedule test drives).
  - An alert or notification system could be implemented to inform dealerships about new test drive requests (optional).

3.0.1.2.1.4 Generic Shop Features (applicable to all shops):

3.0.1.2.1.4.1 Manage promotions and discounts (1.2.3) #38

This section builds upon the previously mentioned functionalities for managing offers and discounts (Section 3.0.1.2.1.1.7 Make offers and flash sales) but expands it to a more generic concept applicable to all shops within the Shop Station.

The specific functionalities would remain similar to those described in Section 3.0.1.2.1.1.7, allowing shops to define and promote special offers, discounts, or flash sales within their profiles.

3.0.1.2.1.4.2 Customer reviews and ratings (1.2.3) #39

- **3.0.1.2.1.4.2.1 Introduction:**
  - This section explores the possibility of integrating a customer review and rating system for shops within the Shop Station. This can help build trust and credibility for shops and inform purchasing decisions for other users.
- **3.0.1.2.1.4.2.2 Inputs:**
  - This section requires defining how customers can leave reviews and ratings:
  - **Post-purchase reviews:** The system could prompt users to submit reviews and ratings after completing a purchase from a shop. This review might include:
  - Overall rating for the shop (e.g., star rating system).
  - Optional text review where customers can elaborate on their experience (positive or negative).
  - **Direct review option (optional):** As an alternative, the system could explore allowing users to leave reviews even if they haven't made a purchase yet (e.g., based on customer service interactions).
- **3.0.1.2.1.4.2.3 Processing:**
  - Store submitted reviews and ratings associated with the corresponding shop.
  - The system might implement mechanisms to prevent abuse (e.g., limiting review frequency per user, flagging inappropriate content).
- **3.0.1.2.1.4.2.4 Outputs:**
  - Display customer reviews and ratings within the shop's profile. This could include an average rating score and the option to view individual reviews.
  - The system should implement mechanisms to ensure review authenticity and transparency (e.g., indicating if a review is from a verified purchaser).

3.0.1.2.1.4.3 Manage Favorite Shops (1.2.3) #40

- **3.0.1.2.1.4.3.1 Introduction:**
  This section details the functionality for users to create a list of favorite shops within the Shop Station. This allows users to easily revisit shops they're interested in and potentially receive notifications about promotions or new products.
- **3.0.1.2.1.4.3.2 Inputs:**
  The system shall provide a mechanism for users to mark shops as favorites:
  - **Favorite button:** A dedicated "favorite" button could be displayed within shop profiles, allowing users to add the shop to their favorites list with a single click.
  - **Favorites list management:** The system should provide a dedicated section within the user profile where they can view and manage their list of favorite shops.
- **3.0.1.2.1.4.3.3 Processing:**
  - Store user- favorited shops within the user's account data.
- **3.0.1.2.1.4.3.4 Outputs:**

- Indicate within shop profiles if a user has favorited the shop (e.g., a highlighted "favorite" button).
- Provide a dedicated section within the user profile to display the list of favorited shops for easy access.

3.0.1.2.1.4.4 Order Management (1.2.3) #41

- **3.0.1.2.1.4.4.1 Introduction:**
  This section details the functionalities for shops to manage customer orders placed through the Shop Station. Efficient order management is crucial for fulfilling purchases and maintaining customer satisfaction.
- **3.0.1.2.1.4.4.2 Inputs:**
  The system shall provide functionalities for shops to handle customer orders:
  - **Order list:** The system should display a list of all received orders within the shop's profile management interface. This list could include:
  - Order details (customer information, ordered products, total price).
  - Order status (e.g., pending, processing, shipped, delivered).
  - **Order status updates:** Shops should be able to update the status of orders as they progress through the fulfillment process (e.g., marking an order as "shipped" after sending the products).
- **3.0.1.2.1.4.4.3 Processing:**
  - Store order information securely within the system.
  - Track the status of each order based on shop updates.
  - The system could offer optional functionalities to integrate with shipping providers (future consideration). This integration could automate tasks like generating shipping labels or tracking shipment progress.

- **3.0.1.2.1.4.4.4 Outputs:**
  - Provide shops with a clear overview of their orders within the profile management interface.
  - The order list should display relevant information and allow shops to update order statuses as needed.
  - Order confirmation emails could be automatically sent to customers upon order placement (with order details and estimated delivery timeframe). Shops might also have the option to send additional notifications to customers regarding order status updates (e.g., shipment confirmation).

3.0.1.2.1.4.5 Manage communication with customers (1.2.3) #42

- **3.0.1.2.1.4.5.1 Introduction:**
  This section explores functionalities for shops to communicate with customers regarding their orders or inquiries. Effective communication builds trust and ensures a smooth shopping experience.
- **3.0.1.2.1.4.5.2 Inputs:**
  The system could provide two options for shop-customer communication:

- **Order-related messages:** The system could facilitate message exchange directly within the context of an order. This allows shops to communicate about order details, clarifications, or potential issues.
- **General inquiries:** The system might offer a general messaging functionality within the Shop Station, allowing customers to send inquiries to shops directly through their profiles (beyond order-specific communication).

◦ **3.0.1.2.1.4.5.3 Processing:**
- Store message threads between shops and customers securely within the system.
- The system might implement features like message notification or read receipts (optional).

◦ **3.0.1.2.1.4.5.4 Outputs:**
- Provide an interface for shops to access and manage order-related messages and potentially general inquiries from customers.
- Customers should be able to view message history and send new messages to shops through the Shop Station platform.

3.0.1.2.1.4.6 Search Functionality (1.2.3) #43

◦ **3.0.1.2.1.4.6.1 Introduction:**

This section highlights the importance of a robust search functionality within the Shop Station. This allows users to easily find products or shops based on their needs and preferences.

◦ **3.0.1.2.1.4.6.2 Inputs:**

The system shall provide a search bar that allows users to enter keywords or search criteria. This search functionality could encompass:
- **Product search:** Users can search for products by name, category, brand (if applicable), or specific keywords within product descriptions.
- **Shop search:** Users can search for shops by name, location (if provided), or category (e.g., auto parts shops, car dealerships).

◦ **3.0.1.2.1.4.6.3 Processing:**
- Utilize search algorithms to match user queries with relevant products or shops within the Shop Station.
- The search functionality might consider filtering options to refine results (e.g., price range, product condition, shop location).

◦ **3.0.1.2.1.4.6.4 Outputs:**
- Display search results based on user queries.
- Search results should be clear and informative, including product details (name, image, price, shop), or shop details (name, location, category) depending on the search type.
- The system might implement features like search history or suggested searches to enhance user experience (optional).

3.0.1.2.1.4.7 User Accounts (1.2.3) #44

- **3.0.1.2.1.4.7.1 Introduction:**

This section outlines the requirements for user accounts within the Shop Station. User accounts allow for personalized experiences, order tracking, and potentially managing favorite shops or reviews.

- **3.0.1.2.1.4.7.2 Inputs:**

The system shall provide a user registration process:

        - Users should be able to create accounts using an email address and password.
        - The system might require additional information during registration (e.g., name, shipping address for future purchases).
    - **3.0.1.2.1.4.7.3 Processing:**
        - Securely store user registration information within the system's database.
        - Implement mechanisms to ensure password security (e.g., password hashing).
    - **3.0.1.2.1.4.7.4 Outputs:**
        - Provide a login interface for users to access their accounts after successful registration.
        - User accounts should offer functionalities like:
        - Viewing order history (if any).
        - Managing favorite shops (referencing Section 3.0.1.2.1.4.3 Manage Favorite Shops).
        - Accessing and leaving reviews (referencing Section 3.0.1.2.1.4.2 Customer reviews and ratings).
        - Updating user information (name, email, etc.).

3.0.1.2.1.4.8 Shopping Cart and Checkout (1.2.3) #45

- **3.0.1.2.1.4.8.1 Introduction:**
  This section details the functionalities for users to add items to a shopping cart, proceed to checkout, and potentially complete purchases from shops within the Shop Station.
- **3.0.1.2.1.4.8.2 Inputs:**
  The system shall provide functionalities for users to manage their shopping cart:
  - **Adding products:** Users can add products to their cart from individual shop profiles. This might involve a "Add to Cart" button displayed with each product.
  - **Viewing cart:** Users can view the contents of their shopping cart, including product details, quantities, and total price.
  - **Modifying cart:** Users can modify the quantities of items in their cart or remove items entirely before proceeding to checkout.

- **3.0.1.2.1.4.8.3 Processing:**
  - Track items added to the user's shopping cart.
  - Calculate the total order price based on product prices and quantities in the cart.
  - The system might integrate with payment gateways to process purchases securely (future consideration).
- **3.0.1.2.1.4.8.4 Outputs:**
  - Provide a shopping cart interface for users to view and manage their cart contents.
  - The checkout process should guide users through steps like:
  - Reviewing cart items and quantities.
  - Providing shipping information (if applicable).
  - Choosing a payment method (future integration with payment gateways).
  - Order confirmation with a summary of purchase details.
-

3.2. Additional Considerations:

- **3.1. Security Considerations:**

  The following sections outline general security considerations for the Rabbeshly Station system:

  - **3.1.1 User Authentication:** The system shall implement strong user authentication mechanisms to prevent unauthorized access to accounts and data. This could involve password hashing, two-factor authentication (optional), and session management.
  - **3.1.2 Data Encryption:** Sensitive user data (e.g., passwords, payment information) should be encrypted at rest and in transit.
  - **3.1.3 System Access Control:** The system shall implement access controls to restrict unauthorized access to sensitive functionalities or data. This involves defining user roles and permissions within the system.
  - **3.1.4 Regular Security Audits:** The system should undergo regular security audits to identify and address potential vulnerabilities.
  - **3.1.5 Secure Coding Practices:** Secure coding practices should be followed during the development process to minimize security risks introduced by software vulnerabilities.
  - **3.2.1 Performance and Scalability:** The system should be designed to handle a growing user base and increasing shop registrations. This involves optimizing performance for search functionalities, product listings, and order processing to ensure a smooth user experience. Scalability considerations ensure the system can accommodate future growth without performance degradation.
  - **3.2.2 System Monitoring and Logging:** The system should implement monitoring and logging mechanisms to track system activity, identify errors, and troubleshoot potential issues.
  - **3.2.3 User Interface (UI) and User Experience (UX):** The Shop Station user interface should be intuitive, user-friendly, and visually appealing. This includes a focus on clear navigation, informative product listings, and a smooth shopping experience.
  - **3.2.4 Integrations (optional):** The system might consider future integrations with external services to enhance functionalities. This could involve:
  - **Payment gateways:** Secure integration with payment gateways would allow users to complete online purchases within the Shop Station.
  - **Shipping providers:** Integration with shipping providers could automate tasks like generating shipping labels or tracking shipment progress.
  - **Social media platforms:** Integration with social media platforms could allow users to share products or shops with their social circles.

- ## 3.3. Future Enhancements:

  - **3.3.1 Advanced Search Functionalities:** The system could explore implementing more advanced search functionalities. This could include filtering options based on specific product attributes (e.g., color, size, brand), price ranges, or shop location (for shops with physical locations).
  - **3.3.2 Personalized Recommendations:** The system might explore incorporating recommendation algorithms to suggest products to users based on their browsing history or purchase behavior.

- **3.3.3 Marketing and Promotions Management:** Shops could benefit from functionalities to manage marketing campaigns and promotions within the Shop Station. This could involve tools for creating targeted promotions, discount coupons, or loyalty programs.

- **3.3.4 Vendor Management Portal:** A vendor management portal could be implemented to streamline shop registration, product uploads, and order management for shop owners or administrators.

- **3.3.5 Mobile App Development:** The Shop Station platform could be extended to mobile app development, allowing users to browse products, shop, and manage accounts from their mobile devices.

3.4 Functional Requirments

Use Case #1: Add Car (Form) - FR #1

- **Actors:** 2.3.1 Owners, 2.3.2 Workers
- **Precondition:**
  - User has registered in the platform.
  - User has launched the (Car Station, Shop Station or Workshop Station) app/system and wants to add a new car.
- **Main Flow:**
  - User selects "Add Car" option.
  - User enters car details like make, model, year, license plate number, and VIN (FR #1.1, FR #1.2, FR #1.3, FR #1.4).
  - System validates information and retrieves additional details (if available) using an external API.
  - User reviews retrieved details (optional).
  - User confirms car information or makes corrections.
  - System creates a new car profile and redirects user to it.
- **Postcondition:** A new car profile is created with entered and retrieved information.
- **Alternative Flows:**
  - Network error: System displays an error message, and user retries data entry.
  - Invalid data: System prompts user to correct invalid fields.

Use Case #2: Add Car (Image Recognition) - FR #2

- **Actor:** 2.3.1.1 Car-Owner(user)
- **Precondition:** User has launched the Car Station app and wants to add a new car profile.
- **Main Flow:**
  - User selects "Add Car with Photo" option (FR #2.1).
  - User captures pictures of the car (FR #2.2).
  - System uses image recognition to extract car make, model, year and colore.
  - System retrieves additional details (if available) using an external API or web scrapping to add more details to the car, based on extracted information.
  - User reviews retrieved details.
  - User confirms extracted information or manually enters details.

    - System creates a new car profile and redirects user to it.
- **Postcondition:** A new car profile is created with extracted and user-confirmed/entered information.
- **Alternative Flows:**
    - Image recognition failure: System prompts user to try again or enter FR#1, or the data is kept until the system recovers from the issue, then it is sent for processing.
    - Network error: System displays an error message, the data is kept until the system recovers from the issue, then it is sent for processing..

Use Case #3: Add Regular Maintenance Schedule - FR #3

- **Actor:**2.3.1.1, 2.3.2, 2.3.3.2
- **Precondition:**
    1. User has opened a car profile and wants to set a reminder for regular maintenance.
    2. Worker has the car info inputted in the Shop Station portal of their shop.
    3. Car Dealership sets the default maintenance schedule of the car before its sold.
- **Main Flow:**
    - User selects "Set Maintenance Schedule" option (FR #3.1).
    - User inputs the matter that needs maintenance. (FR #3.2).
    - System displays options for pre-defined intervals (e.g., every 3 months, every 6 months).
    - User selects a pre-defined interval (optional).
    - User can enter a custom time interval (optional) - FR #3.3.
    - System confirms the chosen interval and stores it in the car profile.
- **Postcondition:**
    - The car profile has a stored reminder frequency for regular maintenance.
    - System schedules a notification for the Car Maintenance time to be displayed on set date.
- **Alternative Flows:**
    - User enters an invalid custom interval: System prompts user to enter a valid value.

Use Case #4: Add Time and Cost of Heavy Maintenance - FR #4

- **Actor:** 2.3.1.1, 2.3.2
- **Precondition:**
    - User has opened a car profile and wants to record upcoming heavy maintenance details.
    - Or, Car info is inputted at Shop Station Shop by 2.3.2.
- **Main Flow:**
    - User selects "Add Heavy Maintenance" option (FR #4.1).
    - User enters estimated cost for the maintenance (FR #4.2, FR #4.3).
    - System stores this information in the car profile.
- **Postcondition:** The car profile stores estimated time and cost for upcoming heavy maintenance.
- **Alternative Flows:**
    - 2.3.2 Adds the labor and parts cost.
    - 2.3.3.2 Adds cost of parts.

Use Case #5: Add Time and Cost of Precautionary Maintenance - FR #5

- **Actor:** 2.3.1.1, 2.3.2, 2.3.3.2
- **Precondition:**
  - User has opened a car profile and wants to record upcoming precautionary maintenance details.
  - Dealership has opened an account and already added the car to its list.
  - Worker inputted car info after it arrived for maintenance.
- **Main Flow:**
  - User selects "Add Precautionary Maintenance" option (FR #5.1).
  - User enters time for the maintenance (FR #5.2, FR #5.3).
  - System stores this information in the car profile.
- **Postcondition:**
  - The car profile stores estimated time and cost for upcoming precautionary maintenance.
  - System notifies the Car Owner when its time to change the part precautionaly.

Use Case #6: Add Time and Cost of Crash Maintenance - FR #6

- **Actor:** 2.3.1.1, 2.3.1.3, 2.3.2, 2.3.5.1
- **Precondition:**
  - User has opened a car profile and wants to record past crash maintenance details (optional).
  - Insurance company, Car Dealership, Worker, and Workshop has an account registered.
- **Main Flow:**
  - User selects "Add Crash Maintenance" option (FR #6.1).
  - User adds time of crash.
  - User enters estimated time and cost for the maintenance (optional) (FR #6.2, FR #6.3).
  - System stores this information in the car profile.
- **Postcondition:** The car profile stores estimated time and cost for past crash maintenance.

Use Case #7: Add Car Specs - FR #7

- **Actor:** 2.3.1.1, 2.3.3.2, 2.3.3.3
- **Precondition:**
  - User has opened a car profile and wants to add detailed car specifications.
  - Car has a profile in the system.
- **Main Flow:**
  - User selects "Add Car Specs" option (FR #7.1).
  - User enters various car specifications like engine type, displacement, transmission, and fuel efficiency (FR #7.2, FR #7.3, FR #7.4, FR #7.5).
  - System stores this data in the car profile.
- **Postcondition:** The car profile contains detailed car specifications entered by the user.

Use Case #8: Add Car Additional Pictures - FR #8

- **Actor:** 2.3.1, 2.3.2, 2.3.3
- **Precondition:** User has opened a car profile and wants to add additional pictures of their car.
- **Main Flow:**
  - User selects "Add Car Pictures" option (FR #8.1).
  - User selects pictures from their device or captures new ones (FR #8.2).
  - System uploads pictures securely and stores them in the car/shop/worker profile.

- **Postcondition:** The car profile contains additional pictures uploaded by the user.

Use Case #9: Add Special Car Specs - FR #9

- **Actor:** Car Station User
- **Precondition:** User has opened a car profile and wants to add details about modifications or unique features.
- **Main Flow:**
  - User selects "Add Special Specs" option (FR #9.1).
  - User enters details about modifications or unique features (FR #9.2).
  - System stores this information in the car profile.
- **Postcondition:** The car profile includes details about the user's car's special features.

Use Case #10: Add Persistent Issues/Accidents - FR #10

- **Actor:** 2.3.1, 2.3.2, 2.3.3
- **Precondition:** User has opened a car profile and wants to report persistent car issues or past accidents (optional).
- **Main Flow:**
  - User selects "Add Persistent Issues" option (FR #10.1).
  - User describes the persistent car issues (optional) (FR #10.2).
  - User can choose to report a past accident (optional) (FR #10.3).
  - System stores this information in the car profile.
- **Postcondition:** The car profile includes details about persistent issues and past accidents (optional).

Use Case #11: Add Car Chassis/VIN Number - FR #11

- **Actor:** 2.3.1, 2.3.2, 2.3.3
- **Precondition:**
  - User has opened a car profile and wants to add the car's chassis or VIN number.
  - Car has a profile on the platform'.
- **Main Flow:**
  - User selects "Add VIN Number" option (FR #11.1).
  - User enters the car's chassis or VIN number (FR #11.2).
  - System stores this information in the car profile.
- **Postcondition:**
  - The car profile includes the car's chassis or VIN number.
  - The shop station provides a list with all the parts that fit that car.

Use Case #12: Display Total Maintenance Costs - FR #12

- the car profile (including regular, heavy, precautionary, and crash maintenance entries, if any).
- System displays the calculated total estimated maintenance cost within the car profile interface (FR #12.1).
- **Postcondition:** The user can view the total estimated cost for maintaining their car.

Use Case #13: Display Car Efficiency - FR #13

- **Actor:** 2.3.1.1, 2.3.3.3
- **Precondition:** User has opened a car profile.

- **Main Flow:**
  - System attempts to retrieve fuel efficiency data for the car using the VIN number (with user consent) through web scrapping or external API (FR #13.1).
  - If VIN lookup fails or user prefers, the system allows the user to enter fuel efficiency data manually (FR #13.2).
  - System prioritizes user-entered data and will display both retrieved and user-entered data for comparison purposes (FR #13.3).
- System displays car's fuel efficiency data within the car profile interface (FR #13.4).
- **Postcondition:** The user can view their car's fuel efficiency data.

Use Case #14: Display Car - FR #14

- **Actor:** 2.3.1, 2.3.2, 2.3.3
- **Precondition:** User has opened a car profile.
- **Main Flow:**
  - System displays the car profile information, including the user-defined car name or nickname (FR #14.1).
- **Postcondition:** The user sees their car's profile details with a prominent car name/nickname.

Use Case #15: Share Update to the Feed - FR #15

- **Actor:** 2.3.1, 2.3.2, 2.3.3
- **Precondition:** User has opened the Car Station feed or their car profile.
- **Main Flow:**
  - User selects "Create Post" option (FR #15.1).
  - User composes a text message, attaches pictures or videos (optional) (FR #15.2, FR #15.3).
  - User can tag other users or cars in their post (optional) (FR #15.4).
  - User selects privacy settings for the post (public feed or private to followers) (FR #15.5).
  - System stores and publishes the post according to user-defined settings (FR #15.6).
- **Postcondition:** The user's post is shared with the Car Station community based on their privacy settings.

Use Case #16: Data Station and Service Station Suggestions and Tips

- **Actor:** Not directly specified (potentially System or Data Station/Service Station)
  - **Description:** This functionality would likely involve a mechanism for data stations and service stations to provide supplementary information or recommendations to car station users. This could be achieved through various methods:
  - Pre-defined informational articles or blog posts within the Shop Station platform.
  - Curated content feeds showcasing suggestions and tips from data stations or service stations.
  - Integration with external resources or knowledge bases for providing relevant information.

Use Case #17: Share Post to Other Platforms

- **Actor:** Car Station User
- **Description:** This use case would allow users to share their car profile posts (potentially including pictures and descriptions) to other social media platforms or external websites.

- **Main Flow:**
  - User creates or edits a car profile post within Shop Station.
  - The system offers options to share the post on other platforms (e.g., Facebook, Instagram).
  - User selects the desired platform and follows the platform's specific sharing process (potentially involving login credentials for the external platform).
  - The car profile post is shared on the chosen external platform.

Use Case #18: Comment, Like and Re-share on Your Car Profile

- **Actor:** Car Station User
- **Description:** This use case focuses on user interaction within the car profile section of Shop Station.
- **Main Flow:**
  - User visits another user's car profile or a feed displaying car profiles.
  - The system allows users to comment on car profile posts, like them, or re-share them within their own car profile feed.
  - User interactions (comments, likes, re-shares) are displayed on the car profile post, fostering a sense of community.

Use Case #19: Create New Posts, Add Pictures from Car Profile or Device, Write Caption, Add Tags

- **Actor:** Car Station User
- **Description:** This use case revolves around user-generated content within the car profile section.
- **Main Flow:**
  - User accesses the functionality for creating new posts within their car profile.
  - The system allows users to:
  - Add pictures or videos from their car profile or directly from their device.
  - Write captions describing the pictures or posts.
  - Add tags or categorize their posts for better searchability (e.g., car model, modifications, events).
  - User creates and publishes the new post within their car profile feed.

Use Case #20: Interact with Posts from Other Platforms such as a Facebook Group from the Same Niche

- **Actor:** Car Station User
- **Description:** This functionality seems to explore the possibility of interacting with content from external platforms directly within Shop Station.
- **Main Flow (Possible Scenario):**
  - The system displays a feed or section showcasing content from relevant Facebook groups (assuming these groups are public or the user is a member).
  - Users can potentially view this content, like or comment on posts from these external groups, without necessarily leaving Shop Station.

Use Case #21: Add Shop Details (FR #21)

- **Actor:** 2.3.1.2, 2.3.2.2, 2.3.3
- **Precondition:** User has access to a Shop Station account.
- **Main Flow:**
  - User navigates to the "Shop Profile" section within the Shop Station interface.
  - The system displays a form for entering shop details (FR #1.2.1).
  - User enters details such as shop name, logo, contact information (address, phone number, website), shop description, working hours, and product/service categories (referencing Use Case #23 for category selection).
  - User can optionally add links to the shop's social media profiles.
  - User submits the completed form.
- **Postcondition:**
- The system validates the entered data (e.g., phone number format, website URL).
- Upon successful validation, the system stores the shop profile details securely within the database and associates them with the user's account.
- The user's shop profile becomes visible within the Shop Station for browsing by potential customers.

Use Case #22: Add Product/Car (FR #22)

- **Actor:** 2.3.1.2, 2.3.2, 2.3.3
- **Precondition:** User has access to a Shop Station account and is logged into their shop profile.
- **Main Flow:**
  - User navigates to the product/car management section within their shop profile.
  - The system provides an interface for adding products or cars (FR #1.2.3).
  - User enters details such as:
    - Product/Car name and description
    - Category selection (referencing Use Case #23 for category selection)
    - Sale price
    - Cost price (optional, for internal shop management)
    - Stock quantity (number of units available)
  - User uploads pictures of the product/car (referencing Use Case #26).
- **Postcondition:**
  - The system stores the entered product/car details within the shop's profile database.
  - The product/car listing appears within the shop's profile on the Shop Station, including details like name, description, category, price, and pictures. Stock quantity information might be displayed or hidden depending on shop preferences.
  - The system might integrate the product/car listing into search functionalities within the Shop Station for better discoverability..

Use Case #23: Add Categories of Products (FR #23)

- **Actor:** 2.3.1.2, 2.3.2, 2.3.3
- **Precondition:** User has access to a Shop Station account and is logged into their shop profile.
- **Main Flow:**
  - User navigates to the product/car category management section within their shop profile.
  - The system offers two approaches for managing categories (FR #1.2.3):
  - Predefined categories: The system displays a list of predefined general product categories relevant to the car enthusiast market (e.g., Engine Parts, Performance Upgrades, Interior Accessories). The user can select these categories when adding products to their profile (referencing Use Case #22).
  - Shop-specific categories (optional): The system allows the user to define custom categories for more granular product organization within their shop (e.g., Performance Camshafts, Aftermarket Lighting).
- **Postcondition:**
  - The system manages the predefined category list and ensures its consistency across the Shop Station.
  - The system allows shops to select from predefined categories or create and manage their custom categories within their profiles.
  - Product listings within the Shop Station utilize categories for organization, potentially using a combination of predefined and shop-specific categories based on shop preferences.

Use Case #24: Import Past Sale Data via Spreadsheet (FR #24)

- **Actor:** 2.3.1.2, 2.3.1.3, 2.3.1.4
- **Precondition:**
  - User has access to a Shop Station account with the appropriate permissions for managing shop data (potentially assigned by the Shop Owner - Actor 2.3.1.2, 2.3.2, or 2.3.3).
  - System Backs up shop data to prepare for the process.
  - User is logged into the Shop Station system with access to the specific shop profile.
  - User has prepared a spreadsheet file containing past sale data for their shop. The spreadsheet should be formatted according to Shop Station's specifications (refer to FR #1.2.3 for details on required data fields and format).
- **Main Flow:**
  - The user navigates to the "Import Data" section within the shop profile management interface.
  - The system provides a functionality to import past sale data.
  - The user selects the prepared spreadsheet file containing past sale data from their local device.
  - The system offers a preview or mapping functionality (optional):
  - The system might display a preview of the data extracted from the spreadsheet, allowing the user to verify its accuracy and format before import.
  - The system might offer a mapping tool where the user can match specific data fields in the spreadsheet with corresponding fields in the Shop Station system (e.g., "Product Code" in the spreadsheet with "SKU" in Shop Station).
  - The user confirms the import process.

    - The system validates the uploaded spreadsheet data based on Shop Station's data format requirements (referencing FR #1.2.3).
    - Upon successful validation:
        - The system imports the past sale data from the spreadsheet into the shop's profile within Shop Station.
        - The system might provide a summary or confirmation message indicating the number of successfully imported sale records.
    - In case of validation errors:
        - The system clearly identifies any errors or inconsistencies found in the spreadsheet data format.
        - The user can review the error messages, make corrections to the spreadsheet file, and attempt the import process again.
- **Postcondition:**
    - Past sale data from the imported spreadsheet is integrated into the shop's profile within Shop Station.
    - The imported data populates relevant sections within the shop profile, potentially including:
    - Order history
    - Customer purchase details (depending on privacy settings and data included in the spreadsheet)
    - Product sales performance data (if applicable data points are included in the spreadsheet)
- **Alternative Flow:**
    - The user chooses to cancel the import process at any point before confirmation.
    - The system discards any uploaded data and maintains the existing shop profile data.

Use Case #25: Export Sales and Storage Reports (FR #25)

- **Actor:** Shop Owner/Administrator
- **Precondition:** User has access to a Shop Station account and is logged into their shop profile.
- **Main Flow:**
    - User navigates to the reporting section within their shop profile.
    - The system offers options for exporting different reports (FR #1.2.4). This might include:
    - Sales reports by product, category, or date range.
    - Inventory reports showing stock levels and potential low-stock items.
    - Customizable reports allowing selection of specific data points and date ranges.
    - User selects the desired report type and customizes the parameters if applicable (e.g., date range for sales reports).
    - The system generates the report in a downloadable format (e.g., CSV, PDF).
- **Postcondition:**
    - The system generates the report accurately, reflecting the chosen criteria and shop data.
    - The user can download the report for further analysis or record-keeping.

Use Case #26: Upload Product/Car Pictures (FR #26)

- **Actor:** 2.3.1, 2.3.2, 2.3.3

- **Precondition:** User is adding a new product/car through Use Case #22 or editing an existing one.
- **Main Flow:**
  - User navigates to the product/car picture upload section within the add/edit product/car interface.
  - The system provides an upload mechanism for product/car pictures (FR #1.2.3). This might include:
    - Drag-and-drop functionality.
    - Browse button to select pictures from the user's device.
    - Capability to upload multiple pictures for a single product/car.
    - User selects and uploads the desired pictures.
- **Postcondition:**
  - The system validates the uploaded pictures for size and format compatibility.
  - Upon successful validation, the system stores the pictures securely within the shop's product/car data.
  - The uploaded pictures are displayed within the product/car listing on the shop's profile and potentially in search results.

Use Case #27: Manage Customer Inquiries (FR #27)

- **Actor:** Shop Owner/Administrator
- **Precondition:** User has access to a Shop Station account and is logged into their shop profile.
- **Main Flow:**
  - User navigates to the customer inquiry management section within their shop profile.
  - The system displays a list of customer inquiries received through the Shop Station platform (FR #1.2.5). This might include:
    - Customer contact information (name, email, phone number).
    - Inquiry content (message text or attachment).
    - Timestamp of the inquiry.
    - Status indicators (e.g., new, replied, closed).
  - User can view individual inquiries and respond to customers directly through the Shop Station interface.
  - The system allows marking inquiries as resolved or closed for better organization.
- **Postcondition:**
  - The system provides a central location for managing all customer inquiries received through the Shop Station platform.
  - Shop owners can efficiently communicate with customers and address their questions or concerns.

Use Case #28: Track Order Fulfillment (FR #28)

- **Actor:** Shop Owner/Administrator (primary), Customer (secondary)
- **Precondition:** A customer places an order through a shop on the Shop Station platform (requires separate use case for order placement).
- **Main Flow:**
  - Shop owner receives notification of a new order within their Shop Station account (FR #1.2.5).
  - The system displays order details such as:
  - Customer information (name, shipping address).
  - Ordered products/cars with quantities.

    - Total order amount.
    - Shop owner can manage the order fulfillment process through the Shop Station interface:
    - Update order status (e.g., processing, shipped, delivered).
    - Provide tracking information to the customer (if applicable).
    - Optionally, the system might allow limited customer access to track their order status.
- **Postcondition:**
    - Shop owners have a streamlined system for managing order fulfillment and keeping customers informed.
    - Customers (if enabled) can conveniently track the status of their orders placed through the Shop Station platform.

Use Case #29: Manage Shop Promotions (FR #29)

- **Actor:** 2.3.2, 2.3.1.2, 2.3.1.3, 2.3.1.4, 2.3.3
- **Precondition:** User has access to a Shop Station account and is logged into their shop profile.
- **Main Flow:**
    - User navigates to the promotions management section within their shop profile.
    - The system provides functionalities for creating and managing shop promotions (FR #1.2.6). This might include:
        - Option to create different promotion types (e.g., percentage discounts, fixed amount discounts, free shipping thresholds).
    - Ability to define promotion details such as:
        - Target product/car categories or specific items.
        - Discount value or free shipping criteria.
        - Start and end date/time for the promotion.
        - Promotion name and description for customer display.
    - User configures the desired promotion details within the Shop Station interface.
    - The system allows previewing the promotion as it would appear to customers before activation.
- **Postcondition:**
    - The system stores and manages promotion details securely within the shop's profile.
    - Once activated, the defined promotions are automatically applied to eligible products/cars within the shop's listings on the Shop Station platform.
    - Customer browsing and search results will reflect the active promotions, potentially highlighting discounted products or free shipping offers.
- **Additional Considerations:**
    - The Shop Station might enforce limitations on the number of concurrent promotions a shop can run or require approval for certain types of promotions.
    - The system should provide clear communication to customers about ongoing promotions and their applicability to specific products/purchases.

Use Case #30: Track Employee Productivity (FR #30)

- **Actor:** 2.3.1.2, 2.3.1.3, 2.3.1.4, 2.3.3
- **Precondition:**
    - User has access to a Shop Station account and is logged into their shop profile.
    - The shop owner has added employees to their shop profile and assigned them appropriate permissions (referencing Use Case #28: Add Employees).
- **Main Flow:**

    - The system provides a dedicated section within the shop profile management interface for tracking employee productivity.
    - The system displays a list of employees associated with the shop.
    - The shop owner can select a specific employee to view their productivity details.
    - The system might offer various options for defining the time range for which productivity is tracked (e.g., daily, weekly, monthly).
    - Depending on the functionalities implemented within Shop Station, the system might display different metrics to assess employee productivity, such as:
    - Number of completed sales or orders processed
    - Average order value or revenue generated
    - Customer satisfaction ratings associated with the employee's interactions (if feedback is collected)
    - Time spent logged into the Shop Station system (potential indicator of activity)
    - (Optional) Completion rates for assigned tasks or projects (if task management features are available)
    - The shop owner can analyze the displayed information to gain insights into individual employee performance.
- **Postcondition:**
    - The shop owner has access to data and reports that can be used to evaluate employee productivity within their shop.
    - This information can be used for various purposes, such as identifying top performers, providing feedback for improvement, or making informed decisions regarding employee roles and responsibilities.
- **Alternative Flow:**
    - The shop owner chooses not to view any employee productivity details at this time.
    - The system remains idle within the employee productivity section.

Use Case #31: Apply Discount to Purchase (FR #31)

- **Actor:** 2.3.2
- **Precondition:**
    - User has access to a Shop Station account and is logged into their shop profile.
    - User has navigated to the "Promotions & Discounts" section within their shop profile management interface.
- **Main Flow:**
    - The system displays functionalities for defining discount rules (referencing FR #1.2.3).
    - The user can choose the type of discount to apply:
    - **Percentage Discount:** User specifies a percentage value (e.g., 10%, 20%) to be deducted from the total order value.
    - **Fixed Amount Discount:** User defines a fixed amount of discount to be subtracted from the total order value (e.g., $10, $25).
    - The user defines how the discount should be applied:
    - **Entire Purchase:** The discount applies to the total price of all items in the customer's cart.
    - **Specific Products or Categories:** The user selects specific products or product categories within their shop to which the discount will be applied. (Optional future enhancement: The

system might allow defining minimum purchase amounts required to qualify for the discount.)
    - The user sets the validity period for the discount rule (optional):
    - The discount can be active for a specific duration (e.g., one week, one month).
    - The discount can be limited to a specific date range (e.g., applied during a promotional event).
    - The user confirms the discount details and saves the rule.
    - The system stores the defined discount rule within the shop's profile.
- **Postcondition:**
    - The defined discount rule is associated with the shop's profile.
    - The system makes sure the sale isn’t in a loss, in which discount > profit margin.
    - During checkout, the system identifies if any applicable discount rules exist based on the customer's cart contents and the validity period of the rules.
    - The system applies the defined discount to the order total according to the chosen discount type, application method, and validity period.
    - The final order confirmation reflects the discounted price.
- **Alternative Flow:**
    - The user chooses to cancel discount rule creation at any point.
    - The system discards any incomplete discount rule definition.

Use Case #32: Manage Shelf Location for Auto Parts (FR #32)

- **Actor:** 2.3.2, 2.3.1.2, 2.3.1.3, 2.3.1.4
- **Precondition:**
    - User has access to a Shop Station account and is logged into their auto parts shop profile.
    - User is creating or editing a product listing within their shop profile.
- **Main Flow:**
    - The system provides an optional feature for managing shelf locations within the product creation/editing interface.
    - The user can define the shelf location for the product using one or both methods:
        - **Textual Description:** The user enters a simple text description of the location (e.g., "Aisle 3, Shelf 2").
        - **Visual Selection (Optional Future Enhancement):** The system might offer a visual representation of the shop's layout (e.g., a shelf diagram). The user can click on specific locations on the diagram to assign them to the product.
    - The user saves the product details, including the defined shelf location (if applicable).
- The system stores the shelf location information alongside the product data within the shop's profile.
- **Postcondition (Optional):**

- The system stores shelf location information for auto parts products within the shop's profile (if the feature is used).
- The system might display the shelf location information alongside product details within the shop profile or search results (clearly indicating it's a shelf location, not part of the product description).
- (Optional future enhancement) The system might highlight product locations on a visual shop layout diagram when a specific product is viewed within the shop profile.
- **Alternative Flow:**
- The user chooses not to define a shelf location for the product.
- The system saves the product details without shelf location information.

Use Case #33: Respond to Customer Inquiries (FR #33)

- **Actor:** 2.3.1.2, 2.3.1.4
- **Precondition:**
  - User has access to a Shop Station account and is logged into their auto parts shop profile.
  - The shop owner receives a notification or message from a customer through the Rabbeshly Station messaging system.
  - The message might include:
    - The name or description of the car part the customer is interested in.
    - Any additional details provided by the customer (e.g., vehicle model, budget).
- **Main Flow:**
  - The shop owner reviews the customer inquiry and identifies the relevant car part(s) within their inventory.
  - (Optional) The shop owner can access the customer's profile to understand their purchase history or preferences (if privacy settings allow).
  - The shop owner responds to the customer's inquiry through the Rabbeshly Station messaging system, including:
  - Confirmation of part availability and stock level.
  - Information on pricing and compatibility (if applicable).
  - Additional details about the product (referencing product description).
  - Offer to assist with further questions or complete a purchase.
  - The shop owner can propose alternative parts or solutions if the initially requested part is unavailable.
- **Postcondition:**
  - The customer receives a response to their inquiry through the Rabbeshly Station messaging system.
  - The shop owner has established communication with the customer and can potentially convert the inquiry into a sale.
- **Alternative Flow:**
  - The shop owner is unable to identify the requested car part within their inventory.
  - The shop owner informs the customer about the unavailability and offers to search for alternatives (if possible).

Use Case #34: Manage Storage and Capital of Products Stored (FR #34)

- **Actor:** 2.3.2, 2.3.1.2, 3.2.1.3, 2.3.1.4
- **Precondition:**
  - User has access to a Shop Station account and is logged into their auto parts Shop Station.
- **Main Flow:**
  - The system provides an inventory management section within the shop profile interface.
  - The system displays a list of the shop's auto parts products.
  - The list might include details such as:
    - Product name or description
    - Category
    - Current stock level
    - Cost price per unit (optional)
    - Total value of current stock (calculated based on cost price and quantity)
  - The shop owner and some employees can view and update stock levels for each product.
  - (Optional) The shop owner can update the cost price associated with a product.
  - The system automatically calculates the total value of the product stock based on the updated stock level and cost price.
  - The shop owner can monitor the overall value of their stored inventory based on the combined value of all products.
- **Postcondition:**
  - The system maintains accurate and up-to-date information about the shop's auto parts inventory, including stock levels, potentially associated cost prices, and the total value

Use Case #35: Add Car Make, Model and Location of Shipping (FR #35)

- **Actor:** Car Dealership Shop Owner (2.3.1.2)
- **Precondition:**
  - User has access to a Shop Station account and is logged into their car dealership shop profile.
  - User has navigated to the section for adding a new car listing within their shop profile.
- **Main Flow:**
  - The system provides a form for entering car details.
  - The user fills out the form, including mandatory fields such as:
  - Car make (e.g., Toyota, Ford)
  - Car model (e.g., Camry, F-150)
  - (Optional) The user can provide additional details about the car, such as:
  - Model year
  - Mileage
  - Trim level
  - Engine and transmission specifications

    - Color
    - Interior features
    - Exterior features
    - Vehicle history report (optional)
    - The user specifies the location from which the car will be shipped (if applicable):
    - The system might offer options for selecting a physical dealership location or a central storage facility.
    - The user can enter a custom shipping location if needed.
    - The user uploads high-quality pictures of the car from various angles (referencing FR #8).
    - The user can optionally add a detailed description of the car's condition and features.
    - The user reviews the entered information and uploaded pictures.
    - The user confirms the car listing creation.
- **Postcondition:**
    - A new car listing is created within the shop owner's profile, showcasing the car's make, model, shipping location, pictures, and potentially additional details.
    - This listing becomes visible to potential customers browsing the Shop Station marketplace (referencing FR #44).

Use Case #36: Add Pictures of Equipment (FR #36)

- **Actor:** Workshop Owner/Manager (3.0.1.3.1.1)
- **Precondition:**
    - User has access to a Shop Station account and is logged into their workshop profile.
    - User has navigated to the workshop profile management section.
- **Main Flow:**
    - The system provides a dedicated section for managing workshop information.
    - The system offers functionality to upload pictures of the workshop's equipment.
    - The user can select images from their device that showcase the workshop's equipment (e.g., diagnostic tools, repair machinery).
    - The system might allow the user to add captions or descriptions to the uploaded pictures (optional).
    - The user confirms adding the pictures to the workshop profile.
- **Postcondition:**
    - Pictures of the workshop's equipment are displayed within the workshop profile on Shop Station.
    - This information can help potential customers understand the capabilities and expertise of the workshop.

Use Case #37: Add Employees and Technicians by Name (FR #37)

- **Actor:** Workshop Owner/Manager (3.0.1.3.1.1)
- **Precondition:**
    - User has access to a Shop Station account and is logged into their workshop profile.
    - User has navigated to the workshop profile management section, specifically for managing employees and technicians.

- **Main Flow:**
  - The system provides a dedicated interface for managing workshop personnel.
  - The user can add new employees or technicians by entering their names.
  - (Optional) The system might offer additional fields for capturing employee details such as:
  - Contact information (limited based on privacy settings)
  - Specialization or area of expertise
  - Years of experience
- **Postcondition:**
  - A list of employees and technicians associated with the workshop is maintained within the workshop profile.
  - This information can be used for internal management purposes or potentially displayed on the public workshop profile (depending on privacy settings).

Use Case #38: Control Employees/Technicians Access (FR #38)

- **Actor:** Workshop Owner/Manager (3.0.1.3.1.1)
- **Precondition:**
  - User has access to a Shop Station account and is logged into their workshop profile.
  - User has navigated to the workshop profile management section for managing employees and technicians.
  - A list of employees and technicians is displayed within the system.
- **Main Flow:**
  - The user selects a specific employee or technician from the list.
  - The system provides options for managing the user's access permissions within the Shop Station workshop system.
  - The owner/manager can define the level of access for each employee/technician, such as:
  - View-only access: Allows viewing workshop information, schedules, and potentially customer car details (based on privacy settings).
  - Edit access: Allows modifying workshop information,
  - The owner/manager confirms the assigned access level for the selected employee/technician.
- **Postcondition:**
  - Each employee/technician associated with the workshop has a designated access level within the Shop Station system.
  - This access control helps ensure data security and restricts functionalities based on the employee's role within the workshop.

Use Case #39: Record Arrival of Cars for Repairs (FR #39)

- **Actor:** Workshop Technician (3.0.1.3.1.1)
- **Precondition:**
  - User has access to a Shop Station account and is logged into their workshop profile.
  - The technician is aware of a car arriving for repairs (e.g., through appointment scheduling or workshop communication).
- **Main Flow:**

    - The system provides a functionality for technicians to manage incoming cars for repairs.
    - The technician can search for the car using details like customer name, license plate number, or repair order ID.
    - The system might display basic information about the car and the repair job (e.g., customer details, reported issues, preferred service).
    - The technician confirms the car's arrival at the workshop within the Shop Station system.
    - (Optional) The technician might add initial notes or observations about the car's condition upon arrival.
- **Postcondition:**
    - The system records the car's arrival at the workshop, including the date and time.
    - This information updates the repair job status and can be used for internal tracking and communication within the workshop.

Use Case #40: Technician Marks the Start of Diagnoses and Repair (FR #40)

- **Actor:** Workshop Technician (3.0.1.3.1.1)
- **Precondition:**
    - User has access to a Shop Station account and is logged into their workshop profile.
    - A car is registered as arrived for repairs within the system (referencing Use Case #39).
    - The technician is ready to begin diagnosing the car's issues.
- **Main Flow:**
    - The technician selects the specific car from the list of ongoing repairs within the Shop Station system.
    - The system displays information about the car and the repair job.
    - The technician initiates the "Start Diagnosis" function within the system.
    - (Optional) The technician can add initial notes or observations about the diagnostic process.
- **Postcondition:**
    - The system marks the repair job status as "In Diagnosis."
    - This update reflects the progress of the repair and can be communicated to the customer (depending on workshop practices).

Use Case #41: Order Parts (FR #41)

- **Actor:** Workshop Technician (3.0.1.3.1.1) (or Workshop Owner/Manager 3.0.1.3.1.1)
- **Precondition:**
    - User has access to a Shop Station account and is logged into their workshop profile.
    - A car is registered for repairs within the system (referencing Use Case #39).
    - The technician (or owner/manager) has identified parts needed for the repair that are not currently available in the workshop's inventory.
- **Main Flow:**
    - The technician (or owner/manager) selects the car with the required parts from the ongoing repairs list.
    - The system displays information about the car and the repair job.
    - The user accesses the parts ordering functionality within Shop Station.

    - The system might offer options for searching and selecting parts from a parts marketplace or through integrated connections with parts suppliers.
    - The user specifies the required quantity of each part.
    - The system displays estimated delivery time and cost for the parts.
    - The user confirms the parts order.
- **Postcondition:**
    - An order for the missing parts is placed through the Shop Station system.
    - The system tracks the order status and estimated delivery timeframe.
    - This information can be used to update the repair job timeline and potentially communicate with the customer.
    - **Note:** Depending on the specific implementation, workshop owners/managers might have additional permissions related to approving parts orders or managing supplier relationships.

Use Case #42: Technician Marks the End of Repair (FR #42)

- **Actor:** Workshop Technician (3.0.1.3.1.1)
- **Precondition:**
    - User has access to a Shop Station account and is logged into their workshop profile.
    - A car is registered for repairs within the system (referencing Use Case #39).
    - The repair work on the car has been completed.
- **Main Flow:**
    - The technician selects the specific car from the list of ongoing repairs within the Shop Station system.
    - The system displays information about the car and the repair job.
    - The technician initiates the "Repair Completed" function within the system.
    - (Optional) The technician can add detailed notes or a repair report summarizing the work performed, identified issues, and replaced parts.
    - The technician might indicate the total cost of repairs based on labor charges and parts used (if applicable).
    - The technician confirms the repair completion.
- **Postcondition:**
    - The system marks the repair job status as "Completed."
    - The system might automatically generate a repair invoice based on the information entered by the technician (depending on the implemented functionalities).
    - This update triggers communication workflows within Shop Station (e.g., notifying the customer about repair completion and potential next steps for payment and car pick-up).

Use Case #43: View Repair Statistics (FR #43)

- **Actor:** Workshop Owner/Manager (3.0.1.3.1.1)
- **Precondition:**
- User has access to a Shop Station account and is logged into their workshop profile.
- **Main Flow:**
    - The system provides a dedicated section for workshop analytics and reporting within the profile management interface.

    - The system offers functionalities to view repair statistics categorized by various criteria, such as:
        - **Timeframe:** Daily, weekly, monthly, or custom date range selection.
        - **Car Make/Model:** Analyze repairs performed on specific car makes and models.
        - **Repair Type:** View statistics for different repair categories (e.g., engine repairs, electrical repairs, bodywork).
        - **Technician:** Analyze the performance of individual technicians (e.g., number of repairs completed, average repair time).
    - The system displays relevant data visualizations or reports based on the chosen criteria.
    - This might include charts illustrating the number of repairs performed, average repair costs, or revenue generated.
    - The system might allow exporting reports in various formats (e.g., CSV, PDF) for further analysis.
- **Postcondition:**
    - The workshop owner/manager gains insights into the workshop's repair activity through data and reports generated by the system.
    - This information can be used for various purposes, such as identifying trends, optimizing repair processes, evaluating technician performance, or making informed business decisions.

Use Case #44: Share Product, Offer, or Owned Car to the Marketplace (FR #44)

- **Actor:** Shop Owner (2.3.1.2, 2.3.2, 2.3.3) or Car Dealership Shop Owner (2.3.1.2)
- **Precondition:**
    - User has access to a Shop Station account and is logged into their shop profile.
    - The user has created product listings, promotions, or car listings within their shop profile (referencing FR #22, #27, or #35).
- **Main Flow:**
    - The system provides functionalities for sharing shop inventory or car listings on the Shop Station marketplace.
    - The user selects the product(s), offer(s), or car listing(s) they want to share on the marketplace.
    - The system might offer options to customize the marketplace listing with additional details or promotions (if applicable).
    - The user confirms sharing the selected items on the marketplace.

- **Postcondition:**
    - The selected products, offers, or car listings from the shop profile become visible to users browsing the Shop Station marketplace.
    - This increases the shop's exposure to potential customers and facilitates online sales or inquiries.

Use Case #45: Search with Filters to Look for Desired Parts or Service (FR #45)

- **Actor:** Customer User

- **Precondition:**
- User has access to the Shop Station platform (website or mobile app).
- **Main Flow:**
  - The system provides a search bar or interface for users to look for parts or services.
  - The user enters keywords or applies filters to refine their search, such as:
    - Product category (for parts)
    - Car make, model, and year (for parts)
    - Repair service type (e.g., engine repair, oil change)
    - Location (to find nearby shops)
    - Price range (optional): Users can define a minimum and/or maximum price range for parts or services.
    - Brand (for parts): Users can filter by specific part brands.
    - Shop rating (optional): Users might be able to filter shops based on average customer ratings.
  - The system displays search results based on the applied filters and keywords.
  - The search results might include:
  - Listings for parts from various shops within the Shop Station marketplace.
  - Profiles of workshops offering repair services matching the user's criteria.
  - Additional information such as product details, prices, shop locations, and potentially customer reviews (depending on implementation).
- **Postcondition:**
  - The customer user has a filtered view of parts or services available on the Shop Station platform.
  - This allows them to compare options, find suitable parts or service providers, and potentially initiate purchases or contact workshops for further inquiries.

3.5 Non-Functional Requirements

- ### 3.5.1 Performance
  - **Search Response Time:** Search queries should return results within 2 seconds for at least 90% of user requests.
  - **Product Listing Load Time:** Product listings should load within 3 seconds for at least 95% of users on a stable internet connection.
  - **Order Processing Time:** Orders should be processed and confirmed within 5 seconds of users completing the checkout process.

- ### 3.5.2 Reliability
  - **System Uptime:** The Shop Station system should achieve an uptime of 99.5% over a monthly period.
  - **Mean Time Between Failures (MTBF):** The system should have an MTBF of greater than 30 days.
  - **Data Loss Prevention:** Data loss incidents should not result in the loss of more than 1 hour of transactional data. Backups should be performed regularly to ensure data recovery in case of failures.

- ### 3.5.3 Availability
  - The Shop Station platform should be available to users 24/7 with minimal scheduled maintenance downtime.
  - Scheduled maintenance windows should be clearly communicated to users in advance and should not exceed 4 hours per quarter.

- ### 3.5.4 Security
  - **User Authentication:** The system should implement strong user authentication mechanisms using password hashing and two-factor authentication (optional) to prevent unauthorized access to accounts.
  - **Data Encryption:** All sensitive user data (passwords, payment information) should be encrypted at rest and in transit using industry-standard encryption algorithms.
  - **System Access Control:** The system should implement role-based access control (RBAC) to restrict access to sensitive functionalities and data based on user roles and permissions.
  - **Regular Security Audits:** The system should undergo regular security audits (at least annually) to identify and address potential vulnerabilities.
  - **Secure Coding Practices:** Secure coding practices should be followed during the development process to minimize security risks introduced by software vulnerabilities.

- ### 3.5.5 Maintainability
  - The system code should be well-documented, modular, and easy to understand to facilitate future maintenance and updates.
  - The system should be designed to allow for easy bug fixes and feature deployments with minimal downtime.

- ## 3.5.6 Portability
  - The Shop Station system should be designed with platform independence in mind, allowing for future migration to different cloud platforms or hosting environments if necessary. (Portability is less critical in this case compared to other requirements, but it can be a future consideration)

- ## 3.5.10 Disaster Recovery:
  - A disaster recovery plan should be established to ensure quick recovery and minimal data loss in case of unforeseen events like natural disasters or cyberattacks.
- **3.5.11 Logging and Monitoring**
  - The system should implement comprehensive logging and monitoring mechanisms to track user activity, system performance metrics, and identify potential issues before they impact users.