**1.Test Case**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precondition | Operation process | Expected result |
| Refund request form submission | The customer has a completed service order in the system | The customer fills out a refund request form, including name, phone number, order ID, and reason for refund | The system successfully receives and stores the refund request, triggering the refund process |
| Membership status check | Customers are non-member users | The system collects customer information and checks membership status | If the customer is not a member, the system provides the option to join the member; If you are a member, 20% discount is offered |
| Trailer service quote generated | The customer's vehicle has broken down and needs towing service | Customer requests tow truck service | The system generates the trailer service quotation and the customer receives the quotation information |
| Deposit collection | Vehicle diagnosis completed | Customer requests repair service | The system generates a quotation based on the vehicle fault level |
| Vehicle diagnostic report | Vehicle arrives at repair shop | The mechanical team diagnoses the vehicle | The system records the diagnosis result and generates the maintenance plan |
| Maintenance completed and customer acceptance | Vehicle maintenance completed | The customer inspects and accepts the vehicle | If the customer is satisfied, the customer takes the car; If not satisfied, trigger subsequent repair or refund process |
| Customer satisfaction survey | Service completed | Collect customer satisfaction survey data | The system records customer comments for subsequent service improvement |
| Form validation error handling | The customer omits information in the form or enters invalid or incomplete information | The user submits a form by missing required fields or entering data that does not meet the requirements | The system should display an error message, prompting the customer to modify or complete the information, the form submission is invalid |

**2.** **Automated test scripts**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Objective | Script | Expected result |
| Customer reception simulation | Verify that the receptionist can correctly register customer and vehicle information, and provide membership options | Simulate a customer arriving at a repair shop or requesting service over the phone, automatically register vehicle information, and check if the system offers an option to join | The system should correctly record customer information and provide membership options |
| Trailer service request automation | Test the trailer service request process, including quote generation and deposit processing | Simulating a customer's request for a towing service, the system automatically generates a quotation and handles the deposit payment process | The system generates an accurate trailer quotation and successfully processes deposit payments |
| Automation of vehicle maintenance processes | Ensure vehicle maintenance process (diagnosis, maintenance plan) is determined and quotation is generated | Simulate vehicle arrival at the repair shop for automatic diagnosis, maintenance plan determination and quote generation | The system should provide a maintenance quotation and prepare for the maintenance work |
| Customer acceptance and feedback automation | Verify customer acceptance processes, including vehicle inspection, after-sales service and refund processes | Simulate customer acceptance of vehicles, automatically conduct customer satisfaction surveys and process refund requests | Record customer feedback, process refund request, provide follow-up repair or refund |

**3. Test data management**

Data generation: Create test data manually or use automated scripts to make sure you cover all possible scenarios.

Data import: Import test data into the system to ensure data consistency.

Data maintenance: During the test process, adjust and update the data according to the data usage to maintain the accuracy and timeliness of the data

Data clearing: After a test is completed, the data of this test is cleared to avoid impact on subsequent tests

Data security management: Periodically back up data and encrypt it for management

**4. Function and process coverage**

Process coverage: Ensure that major business processes such as vehicle maintenance, customer service, and financial management are covered by testing

Functional verification: Verify that each functional module, such as member identity check, quotation calculation, maintenance plan update, etc., works as expected

**5. Performance and safety testing**

Performance test: Test the performance response of the system under the simultaneous use of multiple customers to ensure stability and timeliness

Security testing: Testing data protection measures to ensure the security of customer information and transaction data

**6. Live test**

Test execution: Invite users to participate in system testing and collect feedback

Feedback integration: Adjust system functionality and customer interface based on feedback

**7. Test report and problem tracking**

Report generation: Automatically generates test reports, including process coverage, performance data, and issues encountered

Problem tracking: Use tracking tools to monitor and record problems encountered during testing

**8. Regression and integration testing**

Regression testing: Perform regression testing after the feature changes to ensure that no new problems arise

Integration testing: Ensure that newly developed modules can integrate with existing systems and interact correctly, maintain functional stability, and reduce the cost of fixing problems later