Name of Testers:

1. Hrushikesh
2. Harshvardhan
3. Asha
4. Shilpa
5. Vinith
6. Jayavardhini

Test Plan for Infotainment Service Integrated with a Car Using AUTOSAR Layered Architecture

Version: 1.0

Created: 05/20/2024

Last Updated: 05/20/2024

Status: Currently working on the "TEST STRATEGY"

Table of Contents

1. Introduction

1.1. Purpose
The purpose of this test plan is to outline the testing strategy for the infotainment service integrated with a car, ensuring it meets the functional, integration, and performance requirements within the AUTOSAR (AUTomotive Open System ARchitecture) layered architecture.

1.2. Project Overview
The project involves integrating an infotainment system into a car's existing architecture. This includes media playback, navigation, connectivity features (Bluetooth, Wi-Fi), and user interface interactions, all implemented using the AUTOSAR layered architecture.

1.3. Audience
This document is intended for project managers, developers, testers, and stakeholders involved in the development and deployment of the infotainment system.

2. Test Strategy

2.1. Test Objectives
- Validate the functional requirements of the infotainment system.
- Ensure seamless integration with other vehicle systems.
- Test the system's performance under various conditions.
- Confirm the system's compliance with AUTOSAR standards.

2.2. Test Assumptions
- The hardware components are pre-tested and functioning correctly.
- The test environment mimics the real-world operating conditions of the car.
- Access to all necessary documentation and specifications.

2.3. Test Principles
- Adherence to AUTOSAR standards.
- Comprehensive coverage of all functional and non-functional requirements.
- Systematic and repeatable testing processes.

2.4. Data Approach
- Use of realistic data sets for media files, navigation routes, and connectivity scenarios.
- Test data should cover a range of use cases, including edge cases.

2.5. Scope and Levels of Testing

2.5.1. Unit Testing
Application Layer
  - Verify individual functions of the infotainment application.
  - Validate user interface responses to inputs.
Runtime Environment (RTE)
  - Check RTE configuration and data exchange between application and BSW.
  - Validate task scheduling and timing requirements.
Basic Software (BSW) Modules
  - Test communication services (COM, PDU Router).
  - Validate memory services (NvM, MemStack).
  - Check diagnostic services (DEM, DCM).
Microcontroller Abstraction Layer (MCAL)
  - Verify peripheral drivers (CAN, LIN, Ethernet).
  - Test microcontroller-specific configurations.

2.5.2. Integration Testing
Communication Interfaces
  - Validate CAN, LIN, Ethernet communication between infotainment components.
  - Test data exchange between head unit and external devices (e.g., smartphones).
User Interface Integration
  - Verify touchscreen, button, and voice command functionality.
  - Check display rendering and responsiveness.
Audio System
  - Validate audio playback, volume control, and equalizer settings.
  - Test audio output through speakers and Bluetooth devices.
Navigation System
  - Test GPS signal reception and processing.
  - Validate route calculation and map display.

2.5.3. System Testing
System Boot and Shutdown
  - Verify system start-up and shutdown sequences.
  - Check for error-free initialization and graceful shutdown.
Functional Scenarios
  - Test common use cases (e.g., media playback, phone calls, navigation).
  - Validate multi-functionality (simultaneous audio playback and navigation).
Stress and Load Testing
  - Assess system performance under high load conditions.
  - Validate system behavior under extended operation.

Failover and Recovery
  - Test system response to component failures.
  - Validate recovery mechanisms and data integrity.

2.5.4. User Acceptance Testing (UAT)
- Conduct UAT with real users to ensure the system meets their needs and expectations.
- Collect feedback and make necessary adjustments.

2.5.5. Performance Testing
- Measure response times for user inputs and system commands.
- Assess data transfer rates and communication latency.
- Monitor CPU, memory, and network usage.
- Validate system performance under varying resource availability.

2.5.6. Security Testing
- Ensure the infotainment system is secure from external threats.
- Test for vulnerabilities and potential breaches.

2.5.7. Regression Testing
- Re-run previous test cases to ensure new updates do not break existing functionalities.
- Validate bug fixes and new features.

2.6. Test Effort Estimate
- Exploratory Testing: 5 days
- Functional Testing: 15 days
- Integration Testing: 10 days
- System Testing: 12 days
- UAT: 8 days
- Performance Testing: 5 days
- Security Testing: 5 days
- Regression Testing: 8 days

3. Test Acceptance Criteria

- All high and medium severity defects are resolved.
- Functional test cases achieve a pass rate of 95%.
- Integration and system test cases achieve a pass rate of 98%.
- Performance benchmarks for response time, media playback quality, and connectivity are met.
- Security tests show no vulnerabilities.

4. Test Deliverables

- Test Plan Document
- Test Cases and Test Scripts
- Test Execution Reports
- Defect Logs and Resolution Reports
- UAT Feedback Reports
- Final Test Summary Report

5. Milestone List

Milestone	Description	Date
Test Plan Approval	Approval of the initial test plan document	06/01/2024
Completion of Exploratory Testing	Execution of exploratory test cases	06/06/2024
Completion of Functional Testing	Execution of all functional test cases	06/21/2024
Completion of Integration Testing	Integration testing completion	07/01/2024
Completion of System Testing	System testing completion	07/15/2024
UAT Execution	Completion of user acceptance testing	07/23/2024
Completion of Performance Testing	Performance testing completion	07/28/2024
Completion of Security Testing	Security testing completion	08/02/2024
Regression Testing Execution	Completion of regression testing	08/10/2024
Final Test Report	Compilation and review of all test documentation	08/15/2024

6. Risk Management

Identify Potential Risks Hardware failures, software bugs, communication breakdowns.

Mitigation Strategies:
  - Implement robust error-handling and recovery mechanisms.
  - Conduct thorough pre-testing of individual components.
  - Regularly update and review test plans based on feedback and findings.
  - Allocate additional time for unforeseen issues during test phases.

This structured and detailed test plan ensures comprehensive coverage of all aspects necessary for validating the infotainment service integrated with a car using the AUTOSAR layered architecture, focusing on software features and adhering to industry standards.

7. Diagnostics Approach of the Music System with Car

7.1. Diagnostics Objectives:
    - Ensure the music system's diagnostic capabilities are functioning correctly.
    - Validate the system's ability to detect, report, and handle errors effectively.

7.2. CAN Diagnostics:
    Communication Verification:
    - Verify the music system's communication with the car's Controller Area Network (CAN) for error reporting.
    - Check that the system can send and receive diagnostic messages through the CAN bus.
    Error Reporting:
    - Test the system's ability to report errors detected during operation.
    - Ensure accurate transmission of error codes and statuses to the vehicle's central diagnostic system.

7.3. Ethernet Diagnostics:
- Network Diagnostics:
  - Test the infotainment system's ability to handle diagnostic data over Ethernet.
  - Verify the system's responsiveness to diagnostic requests via Ethernet communication.
- Data Exchange:
  - Ensure proper data exchange and error reporting through Ethernet interfaces.
  - Validate the system's ability to maintain stable communication for diagnostics purposes.

7.4.OBD-II Integration:
- Diagnostics Interface:
  - Confirm integration with the car's On-Board Diagnostics (OBD-II) system.
  - Ensure the music system can retrieve, interpret, and display diagnostic trouble codes (DTCs).
- Real-Time Monitoring:
  - Test the music system's capability to monitor real-time vehicle data and report any discrepancies.
  - Validate the continuous monitoring and reporting of system health and performance.

7.5. User Interface Diagnostics:
- Error Display:
  - Check the diagnostic information displayed on the user interface for clarity and accuracy.
  - Ensure that users are informed of system errors and their severity.

- Troubleshooting Guidance:
  - Validate that the user interface provides troubleshooting steps or suggestions for resolving errors.
  - Ensure the diagnostic interface is user-friendly and accessible.

7.6. Diagnostic Logging and Reporting:
- Log Generation:
  - Ensure the system generates detailed diagnostic logs for all detected errors and anomalies.
  - Validate the accuracy and completeness of diagnostic logs.
- Reporting:
  - Test the system's ability to compile and report diagnostic information to external systems or maintenance personnel.
  - Verify that the reporting format is consistent with industry standards.

7.7. Periodic Self-Checks:
- Automated Diagnostics:
  - Validate the system's ability to perform periodic self-checks and diagnostics.
  - Ensure that these self-checks cover all critical components and functions of the music system.
- Alert Generation:
  - Test the system's ability to generate alerts for any issues detected during self-checks.
  - Ensure timely notification of potential problems to the user or maintenance system.

7.8. Remote Diagnostics:
- Remote Access:
  - Verify the capability for remote diagnostics, allowing technicians to access the system remotely for troubleshooting.
  - Ensure secure and reliable remote access protocols.
- Remote Updates:
  - Test the ability to perform remote software updates based on diagnostic findings.
  - Validate the system's capability to handle remote updates without disrupting normal operation.

7.9. Overall Diagnostics Efficiency:
- Diagnostic Speed:
  - Assess the speed and efficiency of the diagnostic processes.
  - Ensure quick identification and reporting of errors.
- System Stability:
  - Validate that the diagnostics approach does not interfere with the normal operation of the music system.
  - Ensure stable and reliable performance even during extensive diagnostics.

THANK YOU