multisoft
Systems

Android Automotive OS Training

COURSE CONTENT

GET IN TOUCH

Multisoft Systems
B - 125, Sector - 2, Noida

(+91) 9810-306-956

info@multisoftsystems.com

www.multisoftsystems.com

About Multisoft

Train yourself with the best and develop valuable in-demand skills with Multisoft Systems. A leading certification training provider, Multisoft collaborates with top technologies to bring world-class one-on-one and certification trainings. With the goal to empower professionals and business across the globe, we offer more than 1500 training courses, which are delivered by Multisoft's global subject matter experts. We offer tailored corporate training; project Based Training, comprehensive learning solution with lifetime e-learning access, after training support and globally recognized training certificates.

About Course

The Android Automotive OS Training by Multisoft Systems is a specialized program designed for professionals seeking to excel in automotive technology. This comprehensive course provides an in-depth understanding of Android Automotive OS, an advanced platform that powers next-generation vehicle infotainment systems.

Module 1: Android Automotive OS First Steps

1.1 Feature overview

- ✓ Introduction to features
- ✓ Comparison with traditional systems

1.2 Use cases in Android Automotive OS

- ✓ Infotainment
- ✓ Navigation

1.3 GAS vs. Non-GAS

- ✓ Definition and differences
- ✓ Implications for developers

1.4 Architecture big picture

- ✓ System components
- ✓ High-level architecture

Module 2: Android Automotive OS Build System

2.1 Setup the CI/CD build system

- ✓ Configuring CI/CD for Android Automotive
- ✓ Automation and workflows

2.2 Building Android Automotive from AOSP

- ✓ Steps to build
- ✓ Common issues and solutions

2.3 Build Android Automotive Linux kernel

- ✓ Kernel requirements
- ✓ Customization and optimization

Module 3: The Vehicle Hardware Abstraction Layer (VHAL)

3.1 Impact of Project Treble and the VHAL

- ✓ Project Treble explained
- ✓ Changes to VHAL

3.2 Hardware Interface Definition Language (HIDL)

- ✓ Introduction to HIDL
- ✓ Defining interfaces

3.3 VNDK - Run-time linking

- ✓ Purpose of VNDK
- ✓ Implementation in Automotive OS

3.4 Vehicle properties and signals

- ✓ Managing vehicle data
- ✓ Accessing vehicle signals

Module 4: The Android Automotive OS Framework

4.1 Understanding Android Automotive OS services

- ✓ Core services overview
- ✓ Service management

4.2 The role of binder and AIDL

- ✓ IPC mechanism
- ✓ Creating and using AIDL interfaces

4.3 Calling native code using JNI

- ✓ Integrating native code
- ✓ Best practices
- ✓ Android Automotive Applications

4.4 Car apps and the car service

- ✓ Developing car apps
- ✓ Interacting with the car service

4.5 Launcher app development

- ✓ Designing a launcher
- ✓ Implementation specifics

4.6 Hero application adaptation: Dialer, Media Center, Notification Center, etc.

- ✓ Adapting applications for Automotive OS
- ✓ UI/UX considerations

4.7 SystemUI customization

- ✓ Modifying SystemUI
- ✓ Adding custom elements

Module 5: Testing, Debugging, and Performance

5.1 Android tool suite

- ✓ Tools for development
- ✓ Using the tool suite for debugging

5.2 TestSuites – ATS/CTS/STS/VTS/BTS

- ✓ Overview of test suites
- ✓ Integration and execution

5.3 End-end testing – vehicle simulator

- ✓ Simulating vehicle environment
- ✓ Conducting end-to-end tests

Module 6: Android Automotive OS Best Practices

6.1 OTA Updates and smartphone projection

- ✓ Implementing OTA updates
- ✓ Integrating smartphone projection

6.2 Security, SELinux, and permissions enforcing

- ✓ Security architecture
- ✓ SELinux configuration

6.3 Hypervisor architecture for multi-domain control

- ✓ Hypervisor basics
- ✓ Application in Automotive OS

6.4 IPC and Vehicle-State-Machine

- ✓ Managing inter-process communication
- ✓ Designing state machines

6.5 Car communication interface (CAN, LIN, SOMEIP, etc.)

- ✓ Interfacing with vehicle networks
- ✓ Protocol specifics

6.6 Car brand, and variant development

- ✓ Customizing for brands
- ✓ Managing variants