

# Android AOSP Development

**Course Duration: 32 Hours**

**Course code: A-AOSP-D**

## 1. Course Overview

This course provides participants with the skills to develop, customize, and build Android OS from the Android Open Source Project (AOSP). It covers AOSP architecture, source code setup, system services, HAL, device drivers, and custom ROM development. Participants will learn to modify system components, integrate new hardware, and optimize Android builds for real-world devices.

## 2. What you'll learn?

- Understand the AOSP architecture (app framework, libraries, runtime, kernel).
- Set up and build AOSP from source for supported devices/emulators.
- Customize Android system components, UI, and frameworks.
- Work with Android HAL (Hardware Abstraction Layer) and drivers.
- Integrate custom hardware and develop board support packages (BSPs).
- Debug and troubleshoot AOSP builds.
- Develop custom ROMs and optimize Android OS performance.
- Understand security, OTA updates, and deployment best practices.

## 3. Target Audience

- Embedded System Engineers
- Mobile OS Developers
- Android App Developers transitioning into OS-level development
- OEM/ODM Engineers customizing Android for devices
- Professionals interested in custom ROM and system-level Android development

## 4. Pre-Requisites

- Strong knowledge of Linux/Unix environment
- Good understanding of C/C++ and Java
- Familiarity with Git, Bash scripting, and debugging tools
- Prior exposure to Android app development (recommended but not mandatory)

## 5. Course content

### Module 1: Introduction to AOSP

- What is AOSP and its ecosystem?
- AOSP architecture overview
- Android versions and branching model
- Understanding device tree and build targets

### Module 2: Setting Up AOSP Build Environment

- Hardware/software requirements
- Installing prerequisites (Linux, JDK, Git, Python, Make)
- Downloading AOSP source code with repo
- Configuring environment variables

### Module 3: Building AOSP

- Build system overview (Soong/Make)
- Building for emulator vs physical devices
- Understanding product configuration
- Flashing builds to devices/emulator

### Module 4: Android System Architecture Deep Dive

- Application Framework & System Services
- Libraries and Android Runtime (ART/Dalvik)
- Linux Kernel integration

- Role of init, system server, and Zygote

### **Module 5: Customizing AOSP Framework**

- Modifying system UI and settings
- Adding/removing system apps
- Framework APIs customization
- Customizing Android permissions and policies

### **Module 6: Hardware Abstraction Layer (HAL)**

- Introduction to HAL and HIDL/AIDL
- Writing HAL modules for hardware components
- Vendor partition and device drivers
- Integrating new hardware into AOSP

### **Module 7: Device Tree and Board Support Package (BSP)**

- Device tree structure explained
- Creating and customizing BSPs
- Working with bootloader and kernel
- Porting Android to new hardware platforms

### **Module 8: Security and OTA Updates**

- Android security model
- SELinux policies in AOSP
- Signing builds and secure boot
- OTA update mechanism and implementation

### **Module 9: Debugging and Troubleshooting AOSP**

- Using adb, logcat, and dmesg for debugging
- Common build errors and fixes
- Kernel debugging techniques

- Performance profiling tools

### **Module 10: Advanced AOSP Development**

- Building custom ROMs for real devices
- Integrating third-party apps and services
- Optimizing performance for low-memory devices
- Case studies of popular custom ROMs (LineageOS, Pixel Experience)

### **Module 11: Capstone Project**

- End-to-end project: Building and customizing a full Android ROM
- Example: Custom branding, UI changes, preloaded apps, and security features
- Testing on emulator and physical devices
- Final project presentation