

Module:11 App Deployment and Publishing

Q.1 Explain the app release process for both iOS and Android platforms.

Ans=> Android App Release

- 1. Generate Signed APK/AAB in Android Studio.**
- 2. Create Google Play Console Account (one-time \$25).**
- 3. Create App Listing → Name, Description, Screenshots, Icon.**
- 4. Upload AAB in Production (or Internal testing).**
- 5. Fill Store Details → App content rating, privacy policy, permissions.**
- 6. Review & Submit for Release.**
- 7. Google Reviews the App (few hours to few days).**
- 8. App Goes Live**

iOS App Release (Apple App Store)

- 1. Create Apple Developer Account (yearly \$99).**
- 2. Archive Build using Xcode → Upload to App Store Connect.**
- 3. Create App Listing → Name, Description, Screenshots, Icon.**
- 4. Add App Privacy Details + Permissions.**
- 5. Submit Build for Review.**
- 6. Apple App Review Process (1–3 days usually).**

7. **Approve & Release** → Manually or Automatically.
8. **App Goes Live on App Store.**

Q.2 Describe the steps involved in generating app bundles and APKs for deployment.

Ans=>Generate App Bundle (AAB) – Android Studio

1. Open your project in **Android Studio**.
2. Go to **Build > Build Bundle(s) / APK(s)**.
3. Select **Build Bundle(s)**.
4. Choose **Release** mode.
5. Wait for build → AAB is generated in:
`app/build/outputs/bundle/release/`

Generate APK – Android Studio

1. Open your project in **Android Studio**.
2. Go to **Build > Build Bundle(s) / APK(s)**.
3. Select **Build APK(s)**.
4. Choose **Release** mode.
5. APK generated in:
`app/build/outputs/apk/release/`

Q.3 Outline the best practices for submitting apps to the App Store and Google Play.

Ans=>Google Play Store – Best Practices

1. Use App Bundle (AAB)
2. Optimize App Performance
3. Add Proper App Metadata
4. Follow Google Policy
5. Add Privacy Policy
6. Run Pre-launch Report
7. Use Versioning Properly
8. Test In-app Purchases/Notifications

Apple App Store – Best Practices

1. Follow Apple's Human Interface Guidelines
2. Archive & Validate Build in Xcode
3. Prepare High-quality App Metadata
4. Fill App Privacy (App Tracking Transparency)
5. Test on Real Devices
6. Correct App Permissions
7. Follow Strict Review Guidelines
8. Use TestFlight