# App Store Publishing Instructions

## Publishing to Google Play Store

1. Create a Google Play Developer Account  
   - Register at Google Play Console (https://play.google.com/console) with a one-time fee.
2. Prepare App Signing Keys  
   - Generate a keystore file to sign your app (using keytool or Android Studio).  
   - Keep your keystore and passwords safe — required for app updates.  
   - Configure your app’s android/app/build.gradle to use the signing configs.
3. Build a Signed APK or AAB  
   - Run:  
    cd android  
    ./gradlew bundleRelease # For AAB (recommended)  
    ./gradlew assembleRelease # For APK  
   - This creates a signed release artifact in android/app/build/outputs/.
4. Prepare Store Listing Assets  
   - Create and upload:  
    • App title and full description  
    • App icon (512x512 px)  
    • Screenshots for various device sizes (phone, tablet, landscape)  
    • Feature graphic (1024x500 px)  
    • Privacy Policy URL (hosted on your website or a document hosting service)
5. Set Launcher Icons and Splash Screen  
   - Ensure your app uses proper launcher icons configured in android/app/src/main/res/mipmap-\*.  
   - Splash screen configuration is handled either via React Native config or native XML layouts.
6. Create a New App and Upload Release  
   - In Google Play Console, create a new app, fill in the metadata and upload your signed AAB or APK.
7. Configure Content Rating, Pricing, and Distribution  
   - Complete the content rating questionnaire.  
   - Set pricing and available countries.
8. Submit for Review and Publish  
   - Review everything carefully and submit your app.  
   - Google will review and then publish the app on Play Store.

## Publishing to Apple App Store

1. Enroll in Apple Developer Program  
   - Sign up at Apple Developer (https://developer.apple.com/programs/) (annual fee applies).
2. Configure iOS App Icons and Launch Screens  
   - Add all required app icons (various sizes) in Assets.xcassets.  
   - Configure the Launch Screen storyboard or static images for splash screens.
3. Set Up App Metadata  
   - Define your app’s display name, bundle identifier in Xcode project settings.  
   - Add privacy policy URL and support contact info in App Store Connect.
4. Run pod install  
   - Navigate to the iOS folder and run:  
    cd ios  
    pod install  
   - This installs and links all native dependencies.
5. Build and Archive Your App  
   - In Xcode, select Generic iOS Device or a connected device and create an archive (Product > Archive).
6. Upload Your Build to App Store Connect  
   - Use Xcode’s Organizer or Transporter app to upload the .ipa file.
7. Prepare App Store Listing  
   - Fill in the app title, description, keywords, support URL, privacy policy URL.  
   - Upload screenshots for all required device types (iPhone, iPad, etc.).
8. Set Pricing and Availability  
   - Choose your pricing tier or set to free.  
   - Select countries/regions where the app will be available.
9. Submit for Review  
   - Complete app privacy questionnaires and other forms.  
   - Submit for Apple’s review.
10. Release the App  
    - After approval, release the app immediately or schedule a future release.

## Additional Notes

• Privacy Policy:  
 Both stores require a valid privacy policy URL if your app collects any user data. You can host this on your website or use a privacy policy generator service.  
  
• Screenshots:  
 Screenshots must follow each store’s size and format guidelines to avoid rejection.  
  
• App Signing:  
 - Android: Signing with a keystore is mandatory for release builds.  
 - iOS: Code signing with a valid certificate and provisioning profile is mandatory.  
  
• Launcher Images:  
 Properly configure icons and splash screens on both platforms for a professional look.