**Expo development**

Install Node.js (includes npm)

* Go to: <https://nodejs.org>
* Download the LTS version (recommended for most users)
* Run the installer and check the box that says “Add to PATH” during setup

1. Install Expo CLI

npm install -g expo-cli

1. Create a New App

npx create-expo-app MyApp

cd MyApp

1. Install Dependencies

npm install axios @react-native-async-storage/async-storage

expo install expo-image-picker expo-camera expo-secure-store

1. Start the App

npx expo start

1. It opens a QR code in your terminal. Scan it with the Expo Go app (install from App Store or Google Play).

**Create Server**

1. Make sure you have Node.js and npm installed, then run:
2. npm init -y
3. npm install express multer cors

A screenshot of a computer

AI-generated content may be incorrect.4. Run command node server.js

**Public server**

1. Install ngrok

npm install -g ngrok

1. <https://dashboard.ngrok.com/get-started/your-authtoken>
2. ngrok config add-authtoken 2wiE22o3Y4N7B2n97CEk6UHLjoB\_514yeUP55CSXSkvauQcC2

Authtoken saved to configuration file: C:\Users\shis\AppData\Local/ngrok/ngrok.yml

1. Start the Node.js server In CMD Terminal 1: node server.js

This starts your backend server locally on: http://localhost:3000

1. In Terminal 2 (after the server is running): Expose port 3000 using ngrok:

ngrok http 3000

1. This exposes your local server to the internet with a URL like:

<https://8f7e-14-100-77-98.ngrok-free.app>

http://172.17.170.142:3000

**✅ Step-by-Step: Publish Expo App to Google Play Store**

**🔧 1. Set up your project for EAS Build**

If you haven’t already:

bash

CopyEdit

npm install -g eas-cli

eas login

cd your-project/

eas build:configure

Choose android platform and select either **managed** or **bare** workflow (most Expo apps are managed).

**📝 2. Configure eas.json**

Make sure you have a basic eas.json:

json

CopyEdit

{

"build": {

"production": {

"android": {

"workflow": "managed"

}

}

}

}

**📄 3. Set app metadata in app.json or app.config.js**

Make sure these fields are set under expo:

json

CopyEdit

{

"expo": {

"name": "Your App Name",

"slug": "your-app-slug",

"version": "1.0.0",

"android": {

"package": "com.yourcompany.yourapp",

"versionCode": 1,

"permissions": []

}

}

}

* package: must be **unique**
* versionCode: must increase with each update

**🏗️ 4. Build the Android app (APK or AAB)**

bash

CopyEdit

eas build -p android --profile production

This may take 10–20 minutes. When done, you’ll get a download URL for your .aab (recommended) or .apk.

**🔐 5. Set up your Google Play Developer Account**

* Go to https://play.google.com/console
* Pay a one-time $25 USD fee
* Create a new app and fill out store listing info (title, description, icons, etc.)

**📤 6. Upload your build to Google Play Console**

* Go to **Release > Production > Create Release**
* Upload the .aab file you got from eas build
* Fill in version notes, then review and roll out the release

**✅ 7. Wait for Google Review**

* First review usually takes 1–3 days
* After approval, your app will be live on the Play Store

**💡 Notes**

* **Use .aab** (Android App Bundle) for modern Play Store distribution (smaller app size)
* You can automate this process using eas submit (optional)
* For push notifications, set up **Firebase Cloud Messaging (FCM)** and update your Expo credentials