

Déploiement avec EAS + configuration des stores

ANDROID — Google Play Console

0. Enregistrement du Package Name (Validation développeur)

- Renseigner et nommer le package name
- Sélectionner la clé de signature SHA récupérable via eas credential
- Générer un prebuild et mettre la clé donnée en retour dans le dossier indiqué
- Générer un apk de test contenant le fichier et soumettre pour validation du package name

Res: <https://support.google.com/googleplay/android-developer/answer/16761053>

1. Créer l'application (1 seule fois)

- Nom de l'app
- Langue
- Gratuit ou payant
- Package name (⚠️ doit matcher EAS)

2. Fiche Play Store (OBLIGATOIRE)

Infos principales

- Nom public
- Description courte
- Description complète

Assets graphiques

- Icône (512x512)
- Bannière (feature graphic)
- Screenshots (téléphone obligatoire)

3. Politique & conformité

Obligatoire :

- URL de politique de confidentialité
- Déclaration des permissions



4. Content rating

- Questionnaire à remplir
- Donne une classification d'âge



5. Accès à l'app (TRÈS IMPORTANT)

Si ton app a login :

👉 Tu dois fournir :

- un compte test
- ou une vidéo démonstration

Sinon ❌ rejet direct



6. Upload via EAS

Quand tu fais :

```
npx expo eas submit -p android
```

👉 Ton build arrive dans :

Production > Releases



7. Tester avant publication

- Internal testing (recommandé)
- Closed testing



8. Publier

- Cliquer sur "**Publier la version**"
- Attendre validation (1–3 jours)



iOS — App Store Connect

1. Créer l'app

Dans App Store Connect :

- Nom
- Bundle ID (⚠️ identique à EAS)
- SKU (libre)

2. Fiche App Store

Infos

- Nom
- Sous-titre
- Description
- Mots-clés

Assets (plus strict que Android)

- Screenshots par taille :
 - iPhone
 - (optionnel iPad)

👉 Apple est exigeant là-dessus

3. Privacy (ULTRA IMPORTANT)

👉 Section "App Privacy"

Tu dois déclarer :

- données collectées
- tracking
- usage micro/caméra

4. Permissions iOS

Dans ton `app.json` (déjà vu mais critique) :

```
{
  "expo": {
    "ios": {
      "infoPlist": {
        "NSCameraUsageDescription": "Appels vidéo",
        "NSMicrophoneUsageDescription": "Appels audio"
      }
    }
  }
}
```

```
}  
}  
}
```

👉 Si vague ou absent → rejet

5. Lier le build (après EAS submit)

Après :

```
npx expo eas submit -p ios
```

👉 Ton build apparaît dans App Store Connect

Tu dois :

- créer une version
- attacher le build

6. Test avec TestFlight

- Ajouter testeurs
- Tester en conditions réelles

7. Soumettre à Apple

- Bouton "**Submit for Review**"
- Délai : 1 à 5 jours