



React Native Apps Deployment - Android

Chapter 7 - Topic 1

Selamat datang di **Chapter 7 Topic 1** online course
React Native dari Binar Academy!





Halo haloo..

Akhirnya kita sampai di Chapter 7 yaaa. Setelah di chapter sebelumnya kamu belajar tentang React Native Firebase, Geolocation, QR Code, Biometric, dan OAuth, sekarang kita masuk ke materi React Native Deployment untuk Android.

Apa sih React Native Deployment itu?

Langganan jajan es teh di warung sebelah toko sepatu, Cuss, langsung aja deh kita kepoin materi topic satu!





Detailnya, kita bakal bahas hal-hal berikut ini:

- Cara melakukan deployment app ke Playstore





Aplikasi yang sudah kamu bikin, tentu sayang banget kalau nggak di-publish hanya karena kamu nggak tahu caranya. Duh!

Karena itulah, kamu perlu belajar **cara melakukan deployment aplikasi Android-mu ke Playstore.**





Kamu sudah mempelajari banyak hal mengenai React Native, sampai bisa bikin suatu aplikasi dari nol sampai jadi.





Selanjutnya, agar aplikasi yang sudah kamu buat dapat di download dan di-install oleh banyak orang, kamu perlu melakukan deployment ke playstore atau app store.

Eh? Deployment tuh apa ya?





Pengertian Deployment

Deployment adalah kegiatan yang bertujuan untuk **menyebarkan aplikasi yang telah dikerjakan oleh para pengembang.**

Nah, pada chapter ini, kamu akan mempelajari proses-proses apa saja yang perlu kamu lalui untuk bisa melakukan deployment apps ke PlayStore (Android).





Pada React Native, ada beberapa tahapan untuk publish react-native app di antaranya :



Untuk tahu lebih detil cara publishing ke Google Play Stor klik [di sini](#) ya, guys.



Oke, sebagai langkah awal, yang perlu kamu lakukan adalah **menambahkan Keystore ke Project**. Bagaimana caranya?

Yuk kita bahas!





For Your Information

Android mengharuskan semua aplikasi yang dibuat, **ditandatangani secara digital dengan sertifikat sebelum diinstal penggunaanya.**

Untuk mendistribusikan aplikasi Android, melalui Google Play Store, aplikasi tersebut harus kamu tandatangani dengan kunci rilis yang nantinya perlu kamu gunakan untuk semua updates di masa mendatang.





Jadi ingat ya guys, sebelum aplikasi diunggah ke Google Play, aplikasimu harus ditandatangani dengan **upload key**.

Bagaimana caranya?

Yuk, cek di slides selanjutnya.





Cara membuat Upload Key

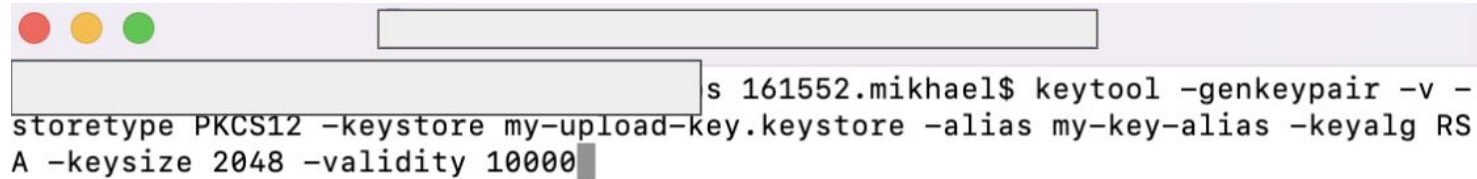
Untuk Windows, jalankan command di bawah ini di directory
`C:\Program Files\Java\jdkx.x.x._x\bin`

x.x.x._x yang dimaksud adalah versi dari jdk seperti contoh pada gambar di samping, kemudian jalankan command di bawah ini:

```
keytool -genkeypair -v -storetype PKCS12 -keystore  
my-upload-key.keystore -alias my-key-alias -keyalg  
RSA -keysize 2048 -validity 10000
```

```
C:\Program Files\Java\jdkx.x.x._x\
```

```
cd "c:\Program Files\Java\jdk1.8.0_152\bin"
```



```
s 161552.mikhael$ keytool -genkeypair -v -  
storetype PKCS12 -keystore my-upload-key.keystore -alias my-key-alias -keyalg RS  
A -keysize 2048 -validity 10000
```

Mari breakdown satu persatu command build di atas:

1. **PKCS12** adalah file archive format yang digunakan untuk menyimpan kunci cryptographic yang digunakan untuk standard kunci cryptography.
2. **my-upload-key.keystore** adalah file yg akan kita generate.
3. **my-key-alias** adalah nama identify untuk key.
4. **RSA** adalah sistem kriptografi yang banyak digunakan berdasarkan kunci privat-publik.
5. **2048** adalah ukuran keystore yang akan kita buat.
6. **validity** menandakan validitas kunci dalam jumlah hari. Dalam perintah ini, kita menyimpannya di 10.000.



```
admins-MacBook-Pro-2:documents 161552.mikhael$ /usr/libexec/java_home  
/Library/Java/JavaVirtualMachines/zulu-11.jdk/Contents/Home  
admins-MacBook-Pro-2:documents 161552.mikhael$
```

Untuk MacOS, kalau kamu nggak tahu di mana lokasi dari Java Home, maka jalankan command berikut :

```
/usr/libexec/java_home
```

Nantinya akan muncul tampilan hasil seperti di atas.

```
/Library/Java/JavaVirtualMachines/jdkX.X.X_XXX.jdk/Contents/Home
```

Lalu jalankan command **cd /your/jdk/path** dan ketik command:

```
sudo keytool -genkey -v -keystore my-upload-key.keystore -alias my-key-alias -keyalg RSA -keysize 2048  
-validity 10000
```



Selanjutnya bagaimana?

Ketika kamu menjalankan command di samping, kamu akan diminta password. Ingat ya, **password ini yang akan dipakai untuk app signing.**

Selanjutnya akan ditanyakan beberapa pertanyaan seperti di samping, kamu perlu mengisi pertanyaan tersebut.

```
Administrator: Command Prompt

C:\WINDOWS\system32>cd "c:\Program Files\Java\jdk1.8.0_152\bin"

c:\Program Files\Java\jdk1.8.0_152\bin>keytool -genkeypair -v -storetype PKCS12 -keystore my-upload-key.keystore -alias my-key-alias -keyalg RSA -keysize 2048 -validity 10000
Enter keystore password:
Re-enter new password:
They don't match. Try again
Enter keystore password:
Re-enter new password:
What is your first and last name?
  [Unknown]: Akash Mittal
What is the name of your organizational unit?
  [Unknown]: Demo Organization
What is the name of your organization?
  [Unknown]: Demo Organization
What is the name of your City or Locality?
  [Unknown]: New York
What is the name of your State or Province?
  [Unknown]: New York
What is the two-letter country code for this unit?
  [Unknown]: US
Is CN=Akash Mittal, OU=Demo Organization, O=Demo Organization, L=New York, ST=New York, C=US correct?
  [no]: yes

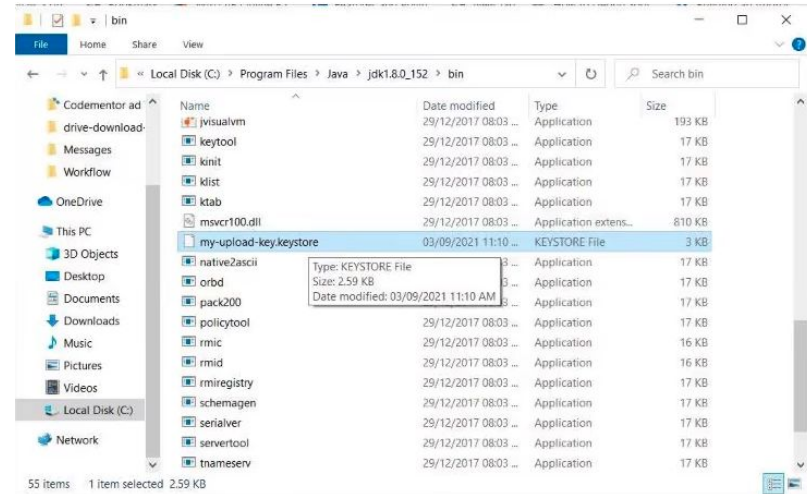
Generating 2,048 bit RSA key pair and self-signed certificate (SHA256withRSA) with a validity of 10,000 days
for: CN=Akash Mittal, OU=Demo Organization, O=Demo Organization, L=New York, ST=New York, C=US
[Storing my-upload-key.keystore]

c:\Program Files\Java\jdk1.8.0_152\bin>
```




Menambahkan Keystore ke Project

Nah, kamu bisa menemukan keystore di folder bin dari folder jdk-mu deh.





Next, yang perlu kamu lakukan adalah melakukan **Android Bundle Release**.

Apa sih Android Bundle Release itu? Dan seperti apa aplikasinya?

Cus, kita bahas~





Sebelum ada Android Bundle Release

FYI guys! Kalau dulu, sebelum developer merilis aplikasi android ke Google Playstore, mereka harus membuat semua kode di aplikasi mereka ke dalam format .apk

Tapi seiring berjalannya waktu, metode ini menimbulkan masalah karena aplikasi yang terus diupdate akan memiliki ukuran yang semakin besar.





Lahirlah Android App Bundle

Nah, untuk mengatasi masalah tersebut maka google merilis Android App Bundle. Ini adalah file berformat .aab yang harus dibuat sebelum aplikasi dirilis.

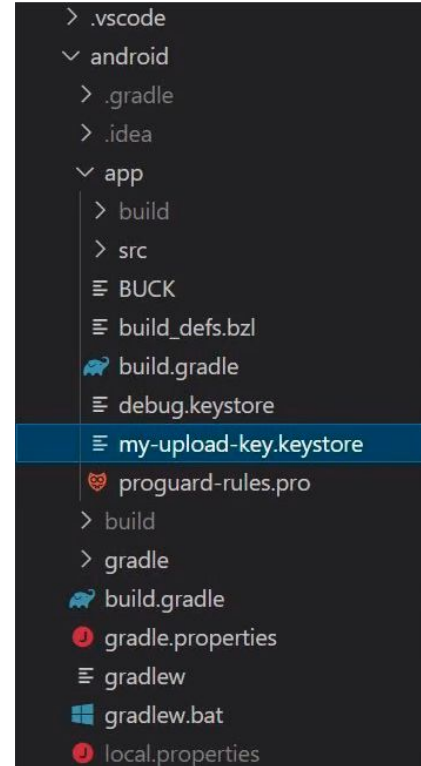
Dengan Android App Bundle ini aplikasi yang terus diupdate nggak akan memiliki ukuran yang sangat besar sehingga nggak menimbulkan masalah di pengguna.





Begini nih guys, cara membuat Android Bundle Release

- Tambahkan copy paste upload keystore ke folder android/app pada project React Native-mu.





- Selanjutnya buka `~/gradle/gradle.properties` atau `android/gradle.properties` dan tambahkan baris berikut :

```
MYAPP_UPLOAD_STORE_FILE=my-upload-key.keystore
```

```
MYAPP_UPLOAD_KEY_ALIAS=my-key-alias
```

```
MYAPP_UPLOAD_STORE_PASSWORD=*****
```

```
MYAPP_UPLOAD_KEY_PASSWORD=*****
```

Ubah bagian upload key password dan store password dengan **password yang kamu buat untuk signing keystore sebelumnya**.





Begini tampilan codingnya

```
17 # This option enables only the build with decoupled projects. More details, visit
18 # http://www.gradle.org/docs/current/userguide/multi\_project\_builds.html#sec:decoupled\_projects
19 # org.gradle.parallel=true
20
21 # AndroidX package structure to make it clearer which packages are bundled with the
22 # Android operating system, and which are packaged with your app's APK
23 # https://developer.android.com/topic/libraries/support-library/androidx-rn
24 android.useAndroidX=true
25 # Automatically convert third-party libraries to use AndroidX
26 android.enableJetifier=true
27
28 # Version of flipper SDK to use with React Native
29 FLIPPER_VERSION=0.54.0
30 MYAPP_UPLOAD_STORE_FILE=my-upload-key.keystore
31 MYAPP_UPLOAD_KEY_ALIAS=my-key-alias
32 MYAPP_UPLOAD_STORE_PASSWORD=put_store_password_here
33 MYAPP_UPLOAD_KEY_PASSWORD=put_store_password_here
```



- Buka android/app/build.gradle dan tambahkan code ini di bagian release :

```
if
(project.hasProperty('MYAPP_UPLOAD_STORE
_FILE')) {
    storeFile
file(MYAPP_UPLOAD_STORE_FILE)
    storePassword
MYAPP_UPLOAD_STORE_PASSWORD
    keyAlias
MYAPP_UPLOAD_KEY_ALIAS
    keyPassword
MYAPP_UPLOAD_KEY_PASSWORD
}
```





Begini tampilan codingnya

```
146 }
147
148 signingConfigs {
149     debug {
150         storeFile file('debug.keystore')
151         storePassword 'android'
152         keyAlias 'androiddebugkey'
153         keyPassword 'android'
154     }
155     release {
156         if (project.hasProperty('MYAPP_UPLOAD_STORE_FILE')) {
157             storeFile file(MYAPP_UPLOAD_STORE_FILE)
158             storePassword MYAPP_UPLOAD_STORE_PASSWORD
159             keyAlias MYAPP_UPLOAD_KEY_ALIAS
160             keyPassword MYAPP_UPLOAD_KEY_PASSWORD
161         }
162     }
163 }
164 buildTypes {
165     debug {
166         signingConfig signingConfigs.debug
167     }
168     release {
169         // Caution! In production, you need to generate your own keystore file.
170         // see https://reactnative.dev/docs/signed-apk-android.
171         signingConfig signingConfigs.release
172         minifyEnabled enableProguardInReleaseBuilds
173         proguardFiles getDefaultProguardFile("proguard-android.txt"), "proguard-rules.pro"
174     }
175 }
176 }
```



Setelah proses Android Bundle Release, yang selanjutnya harus kamu lakukan adalah melakukan **Generating APK**.





Generating APK

- Setelah semua langkah di atas sudah selesai, selanjutnya kamu akan membuat build release pada aplikasi.

Ini optional ya, tapi kamu boleh coba run-android pada project react-native untuk ngetes bahwa nggak ada kesalahan atau error pada aplikasi untuk build.

Jalankan command berikut :

```
cd android  
  
./gradlew bundleRelease
```





- Command di atas akan membuat bundle aplikasi dengan AAB format, sesuai dokumentasi Google Play. FYI, sejak Agustus 2021 format AAB dibutuhkan untuk upload ke Google Play.

Dengan format file AAB, akan membantu mengoptimasi size dari APK yang bermacam-macam sesuai dengan konfigurasi. Format AAB terbilang lebih kecil dari biasanya.





- Selanjutnya jalankan `npx react-native run-android --variant=release`.

Pastikan nggak ada aplikasi sebelumnya yang terinstall pada device atau emulator, kemudian kamu bisa melakukan testing ke aplikasi dalam versi release sebelum kamu upload ke Google Play.





Last but not least, kamu akan belajar cara merilis aplikasimu alias **Deploy ke Google PlayStore.**

Are you ready?

Pssst, kamu boleh langsung bertanya pada Fasil kalau ada yang kurang kamu pahami ya, guys :)





Saatnya deploying ke Google Playstore!

Yaps! Sekarang saatnya kamu mencoba upload APK yang sudah kamu generate ke Google Play. Ini langkahnya:

1. Sebelum diupload, kamu sudah harus punya akun Google Console (paid) dulu ya.





2. Kemudian kamu perlu mengisi beberapa informasi seperti:

1. Appname
2. Category
3. Description
4. Bahasa
5. Negara
- 6.

Dll.

Saat mengupload kamu akan diminta screenshot dan video juga.

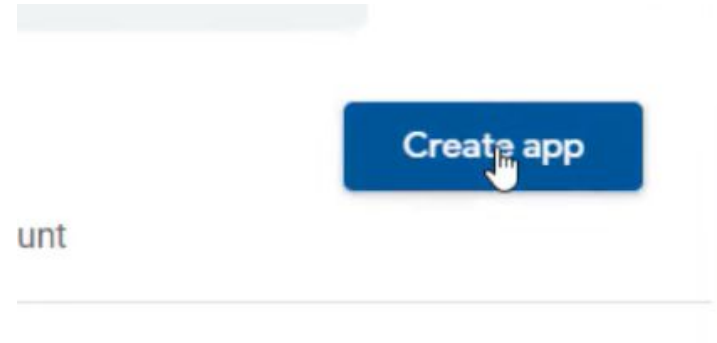




3. Selanjutnya buka <https://play.google.com/console/>.

Kalau kamu sudah register akun menjadi Google Developer Account maka seharusnya kamu sudah bisa mengakses dashboard Google Play Console.

Di pojok kanan atas, klik tombol Create App.





4. Setelah itu, kamu akan diarahkan ke tampilan seperti di samping.

Di sini, kamu akan diminta untuk mengisi app detail , seperti app name dan default language, app or game , serta apakah aplikasi kalian free or paid.

The screenshot shows the Google Play Console interface. On the left is a sidebar menu with options: All apps, Inbox (10), Policy status, Users and permissions, Order management, Download reports, Account details, Developer page, and Associated developer accounts. The main area is titled 'Create app' and contains the 'App details' section. This section has three fields: 'App name' with a text input box and a note below it stating 'This is how your app will appear on Google Play. It should be concise and not include price, rank, any emoji or repetitive symbols.'; 'Default language' with a dropdown menu currently set to 'English (United States) - en-US'; and 'App or game' with a note stating 'You can change this later in Store settings'. At the bottom right of the form are 'Cancel' and 'Create' buttons.



5. Lalu kalau kamu scroll ke bagian bawah lagi, kamu akan diminta untuk menyetujui beberapa aturannya.

Ceklis semua rules pada kotak di samping, lalu klik Create App.

The application meets [Developer Program Policies](#). Please check out [these tips on how to create policy compliant app descriptions](#) to avoid some common reasons for app suspension. If your app or store listing is [eligible for advance notice](#) to the Google Play App Review team, [contact us](#) prior to publishing.

Play App Signing

☒ Accept the Play App Signing Terms of Service
To publish [Android App Bundles](#) on Google Play you need to accept the [Play App Signing Terms of Service](#). You will be able to choose your app signing key when creating a release. [Learn more](#)

US export laws

☐ Accept US export laws
I acknowledge that my software application may be subject to United States export



6. Selanjutnya, kamu akan dibawa ke app dashboard berisi langkah-langkah lanjutan, di antaranya kamu perlu mengisi informasi mengenai app yang akan kamu rilis.

Dashboard

Set up your app



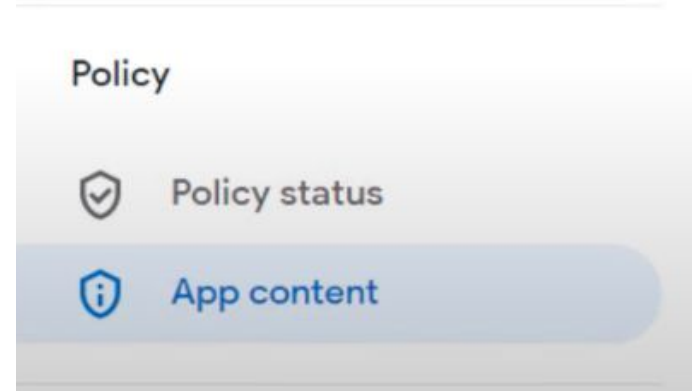
Provide information about your app and set up your store listing

Let us know about the content of your app, and manage how it is organized and presented on Google Play


[View tasks](#) ▾



7. Kemudian lihat di bagian Policy pada menu App Content. Scroll ke bawah App content sampai menemukan App Access, jika sudah ketemu klik tombol Start.



App access

 Not started • Provide instructions on how to access restricted parts of your app

If parts of your app are restricted based on login credentials, memberships, location, or other forms of authentication, provide instructions on how to access them

[Start](#)



8. Ceklis pada pilihan 'All functionality is available without special access'.

← App content

App access

If parts of your app are restricted based on login credentials, memberships, location, or other forms of authentication, provide instructions on how to access them. Make sure this information is kept up to date.

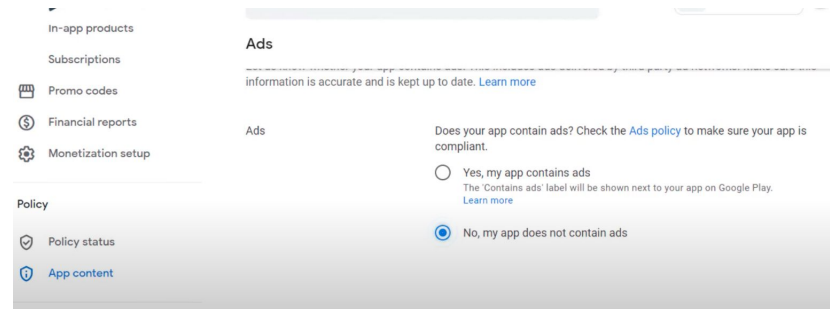
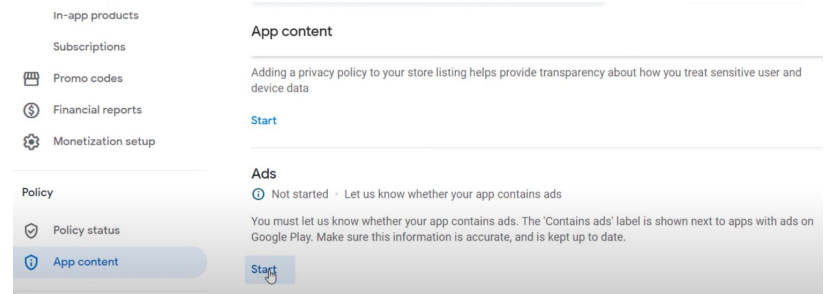
Google may use this information to review your app. It won't be shared, or used for any other reason. [Learn more](#)

☒ All functionality is available without special access

☐ All or some functionality is restricted

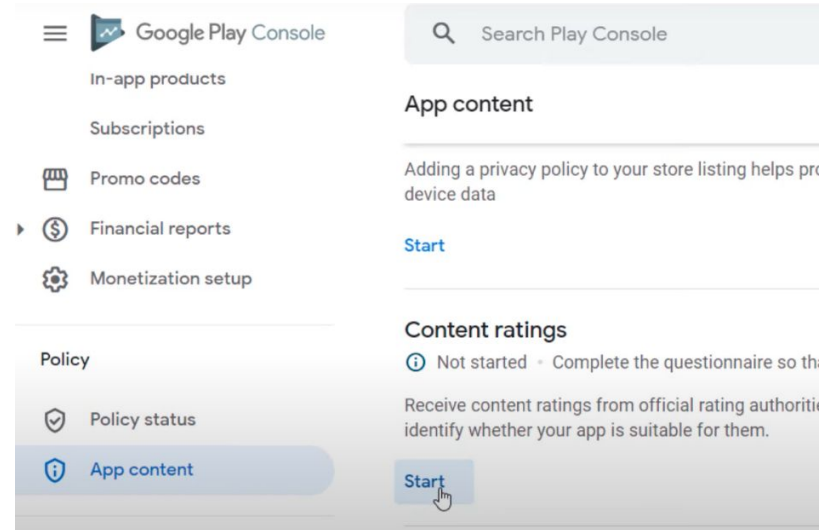


9. Selanjutnya kembali ke App Content dan pilih bagian Ads. Ceklis bagian yang menyatakan bahwa apps nggak memiliki contain Ads.





10. Kemudian kembali ke App Content dan klik start Contentt Ratings.





11. Isi informasi email address, kemudian kategori dari app.

← App content

Content ratings

Discard changes

1 Category — 2 Questionnaire — 3 Summary

Category

Email address

This will be used to contact you about your content ratings. It may be shared with rating authorities and IARC.

Category ☐ Reference, News, or Educational

The primary purpose of the app is to present factual information in a neutral



12. Kemudian kamu akan diberikan pertanyaan yes/no question, yang bertujuan untuk memastikan bahwa aplikasimu sudah sesuai dengan peraturan Google, seperti nggak ada material kejahatan, atau drugs sell atau aktivitas yang berkaitan dengan menjual barang yg dilarang dll.

Kamu bisa mengisi ini semua dengan pilihan No, kemudian klik Save dan Next.

Content ratings

[Discard changes](#)

Utility, Productivity, Communication, or Other

Violence

Does the app contain violent material? [Learn more](#)

Please note that this question does **not** refer to user-generated content.

☐ Yes ☐ No

Sexuality



13. App kamu akan dinilai untuk rating sesuai kategori user, seperti misalnya di samping terating 13+ yang menunjukkan bahwa aplikasi family friendly, kemudian klik Submit.

Content ratings

Disca

Summary

Ratings shown below may differ from the ratings shown to users on Google Play.

- Google may reject your app or app update for misrepresentation of your app's content
- Google may use your questionnaire responses to generate ratings for specific territories as required by local law
- Rating authorities participating in IARC may change your app's rating after they review it

Google and IARC will share your contact information, questionnaire responses, ratings, developer support requests, and app details with rating authorities. [Learn more](#)

Category

Utility, Productivity, Communication, or Other



Your changes have been saved

[Back](#)



14. Kemudian kembali ke App Content dan pilih bagian target audience and content.

Policy



Policy status



App content

Target audience and content



Not started

You must let us know the target age group of your app, and other information about its contents. This helps us ensure that apps designed for children are safe and appropriate.

[Start](#)





15. Di sini kamu akan diminta mengisi rentang umur dari user aplikasimu, dan beberapa pertanyaan mengenai konten pada aplikasi. Isi sesuai kebutuhan dan klik Save.

Target audience and content

[Discard change](#)

- ☐ 5 and under
- ☐ 6-8
- ☐ 9-12
- ☐ 13-15
- ☐ 16-17
- ☐ 18 and over

Target audience and content

[Discard changes](#)

characters or animations. The 'Not designed for children' label may be shown next to your app on Google Play.

Answer 'No' if you're unsure, prefer not to answer, or think your store listing doesn't unintentionally appeal to children.

Appeal to children

Could your store listing unintentionally appeal to children?

☐ Yes

The 'Not designed for children' label may be shown next to your app on Google Play. [Learn more](#)

☐ No

If Google disagrees with your answer, you won't be able to update your app.



16. Kembali ke App content dan News app. Di section ini kamu akan ditanya apakah aplikasimu adalah jenis News app atau bukan. Klik bagian No jika bukan.

← App content

News apps

Let us know whether your app is a news app. This helps us make sure you comply with the Google Play News policy. [Learn more](#)

News apps

Is your app a news app?



No



Yes

I confirm my app complies with the [Google Play News policy](#)



17. Selanjutnya kamu perlu set Privacy Content. Klik App Content dan klik Privacy Setting. Isi Privacy Content, lalu klik Next.

App content

Let us know about the content of your app. This is to make sure your app complies with Google Play policies.

[Learn more](#)

To do

Privacy policy

 Not started · Add a privacy policy to your store listing

Adding a privacy policy to your store listing helps provide transparency about how you treat sensitive user and device data

[Start](#)





18. Selanjutnya kembali ke App Dashboard kemudian scroll pada bagian App Setup. Kamu akan melihat progres dari aplikasimu sebelum di upload.

Set up your app



Provide information about your app and set up your store listing

Let us know about the content of your app, and manage how it is organized and presented on Google Play

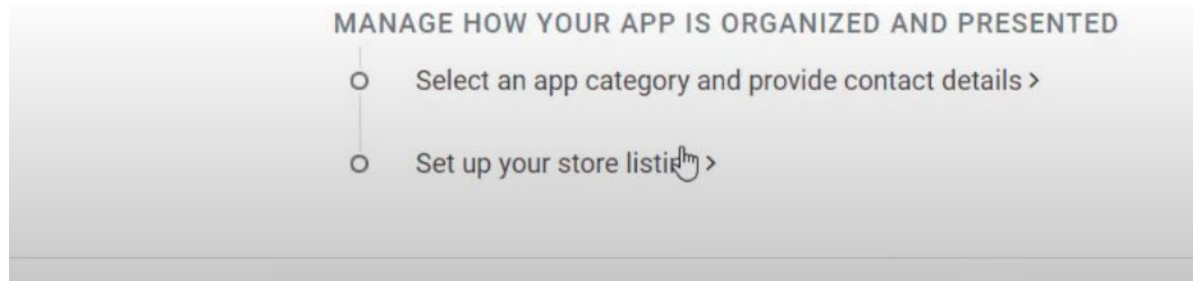
 6 of 8 complete ^

LET US KNOW ABOUT THE CONTENT OF YOUR APP

- ✓ Set privacy policy
- ✓ App access

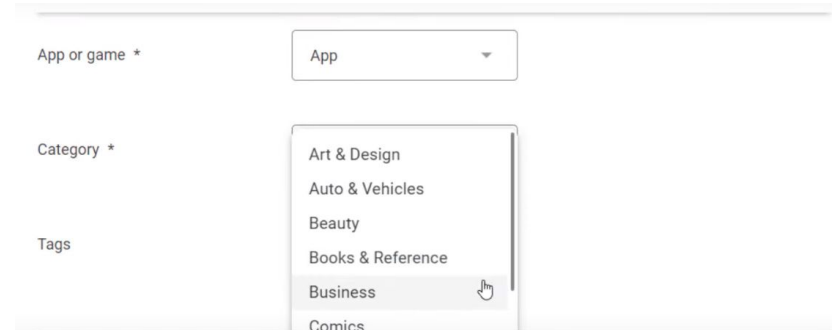


19. Scroll ke bawah dan klik 'Select an app category and provide contact details'.





20. Kamu akan diminta untuk mengisi tipe app atau game dan kategorinya, serta kontak developer dan tags. Pilihlah yang sesuai dengan aplikasimu, lalu klik next.

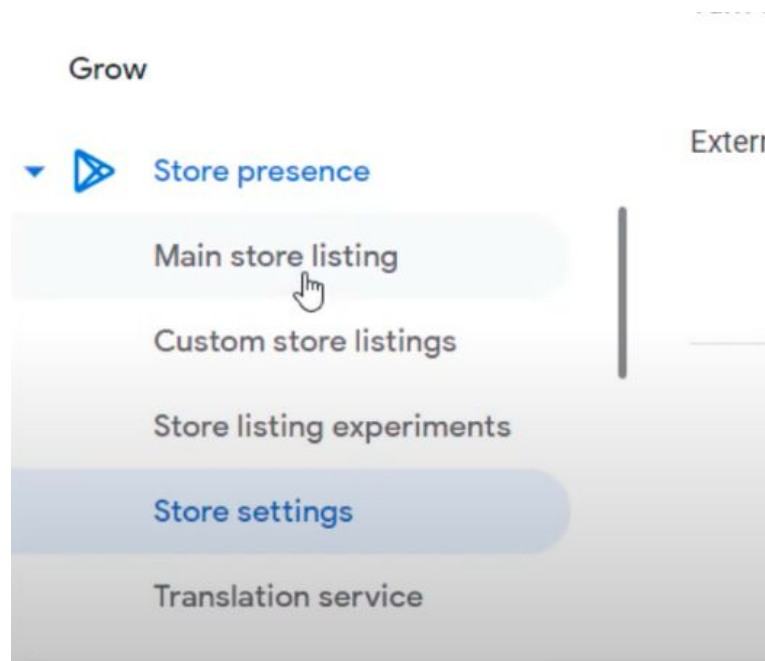


The screenshot shows a portion of the Google Play Store app submission form. It includes the following fields:

- App or game ***: A dropdown menu with 'App' selected.
- Category ***: A dropdown menu with a list of categories: Art & Design, Auto & Vehicles, Beauty, Books & Reference, Business (highlighted with a mouse cursor), and Comics.
- Tags**: A text input field.



21. Pada menu sebelah kiri, klik 'Main store listing'.





22. Di section ini kamu perlu memberi nama aplikasi, short description, full description, app icon, feature graphic, screenshot aplikasi, dan lainnya, sesuai dengan aplikasimu. Bagian ini akan muncul pada PlayStore, kemudian klik Save.

Advanced settings

Grow

- Store presence
- Main store listing**
- Custom store listings
- Store listing experiments
- Store settings
- Translation service

Main store listing

Default – English (United States) – en-US Manage translations ▼

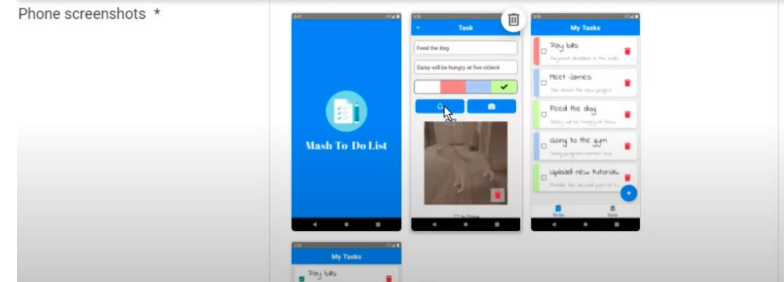
App name * Mash To-Do List

Short description *
A short description for your app. Users can expand to view your full description.

Full description *

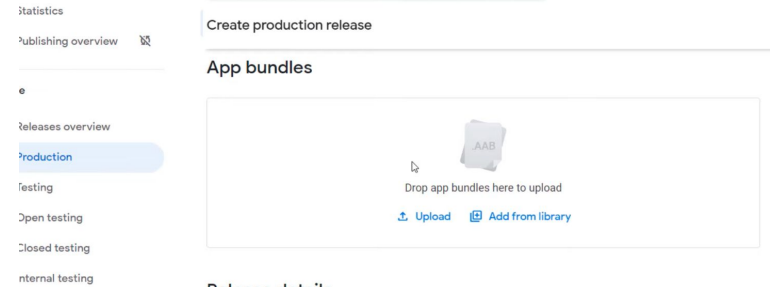
Main store listing

Default – English (United States) – en-US Manage translations ▼





23. Kemudian pada menu di kiri pilihlah Production. Selanjutnya, kamu akan diminta untuk mengupload app dalam format aab. Cuss, upload deh file aplikasimu.





24. Selanjutnya kamu akan diminta mengisi release information. Biasanya ini berisikan update atau feature. Untuk sekarang isikan aja First Release, kemudian klik Review Release di bagian bawah kanan.

Create production release

Release notes

[Copy from a previous release](#)

<en-US>
Enter or paste your release notes for en-US here
</en-US>


Release notes provided for 0 language

Let users know what's in your release. Enter release notes for each language within the language tags.



25. Selanjutnya klik 'Start rollout to Production' pada tombol di bagian bawah kanan.

Create production release

 1 Warning

Show less ^

1 MESSAGE FOR VERSION CODE 1

 Warning

This App Bundle contains native code, and you've not uploaded debug symbols. We recommend you upload a symbol file to make your crashes and ANRs easier to analyze and debug. [Learn More](#)

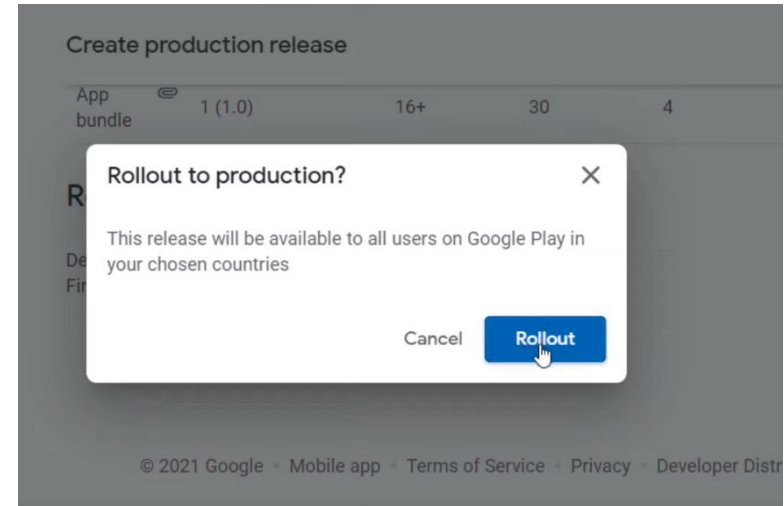
New app bundles

File type	Version	API levels	Target SDK	Screen layouts	ABIs	Required features
Review your release before rolling it out				Edit release	Start rollout to Production	



26. Klik Rollout pada pop up dan selanjutnya Google akan mereview aplikasimu paling lama 7 hari sebelum di-release dan bisa di akses di PlayStore.

Selesai deh! Sekarang kamu sudah tau caranya rilis aplikasimu ke PlayStore. Good job, guys!



Saatnya kita
Quiz!





1. Langkah apa yang harus kamu ambil untuk melakukan digital signing?

- A. Create bundle release
- B. Create aab app
- C. Create .keystore file



2. Apa format file untuk aplikasi yang sudah kamu build yang dapat di upload ke PlayStore?

- A. .aab
- B. .apk
- C. .rar



3. Apa keuntungan menggunakan format aab?

- A. Review Google cepat ketika upload
- B. Dapat dilakukan testing sebelum upload ke production
- C. Size apk menjadi lebih kecil dibanding apk



4. Apa langkah selanjutnya setelah kamu berhasil mengupload app ke Google Console?

- A. Mengisikan konten untuk aplikasi
- B. Menunggu sampai Google selesai mereview app
- C. Generate apk file



5. Sebelum menggunakan format .aab, apa format file untuk aplikasi yang sudah di-build yang dapat di upload ke PlayStore?

- A. .aab
- B. .apk**
- C. .ipa

Terima Kasih!



Next Topic

loading...