

CS 473 - MDP

Mobile Device Programming

© 2021 Maharishi International University

All course materials are copyright protected by international copyright laws and remain the property of the Maharishi International University. The materials are accessible only for the personal use of students enrolled in this course and only for the duration of the course. Any copying and distributing are not allowed and subject to legal action.



Maharishi International
University

CS 473 - MDP

Mobile Device Programming

MS.CS Program

Department of Computer Science

Renuka Mohanraj , Ph.D.



Maharishi International
University

CS 473 – MDP

Mobile Device Programming

Lesson 13
Publish APK on Google Play Store



Maharishi International
University

Wholeness of the lesson

In this lesson we show you how to publish your application on Google Play. Start by generating and uploading the APK file to the completion of the Google Play page information. *Similarly, during the transcendental meditation one can do the self-actualization and then bring into all-knowing awareness in the cosmic consciousness.*

Useful Resources for the career

- Android Weekly is a free newsletter that helps you to stay cutting-edge with your Android Development also job openings around the world.
 - The newsletter comes once a week and covers a broad range of topics like tutorials, screencasts, news... just everything that's awesome in the Android Development world!
 - <https://androidweekly.net/>
- Koltin Weekly is a free newsletter to get your weekly Kotlin Dose
 - <http://kotlinweekly.net/>
- Google Developer our Alumni tech talk link about Modern Android Engineering and Google Interview Process
<https://www.youtube.com/watch?v=gXHq3dIAyqo>

Technologies support career

- Essential Knowledge covered in this course

- Kotlin programming
- Strong base on Android Fundamentals and Components
- Basic and Advanced UIs
- Data Persistence
- Android Jetpack
- Material Design(Demo added on your Extras--Teams folder)
- Kotlin Coroutines
- Retrofit for Web Services

- Popular Android Design Pattern

- Model View Presenter (MVP), Model View View Model(MVVM)(Demo added on your Extras --Teams folder), Model View Intent(MVI)

Technologies support career

- Dependency Injection with Hilt

- <https://codelabs.developers.google.com/codelabs/android-hilt/#0>
- <https://developer.android.com/training/dependency-injection/hilt-android>

- RxJava

- Reactive Extensions are a set of interfaces and methods for composing asynchronous and event-based programs by using observable sequences.

- Firebase

- **Firebase** is a mobile platform that helps you quickly develop high-quality apps, grow your user base, and earn more money.
- <https://codelabs.developers.google.com/codelabs/firebase-android/#0>

- Testing Framework

- <https://developer.android.com/codelabs/advanced-android-kotlin-training-testing-test-doubles#9>

- JetPack Compose

- It's an added advantage. Currently its not mandatory to get Job.

How to share your app

Ways to distribute your app to your friends or team or higher authorities

- Zip it up and share, they can extract and import the project
- Share the source code through cloud drive like one drive, google drive etc.,
- Publish to GitHub
- Make an APK

Publish .apk worldwide

- After you have created, and fully debugged, your application, you might want to deploy it to the Google Store for others to enjoy.
- Before uploading the application in Google Play Store(android market), the basic points to be noted are as follows:
 - Testing the application.
 - Checking the application performance.
 - SDK Compatibility.
 - Remove log data.
- The following sections outline the steps for publishing your applications.

Steps for publishing your app

- Prepare app for release
- Generate signed APK
- Upload to Google Play
- Run alpha and beta tests(optional)
- Publish to the world
- Refer : <https://www.thedroidsonroids.com/blog/how-publish-app-google-play-guide-and-checklist> from the Android weekly newsletter.

What is an APK?

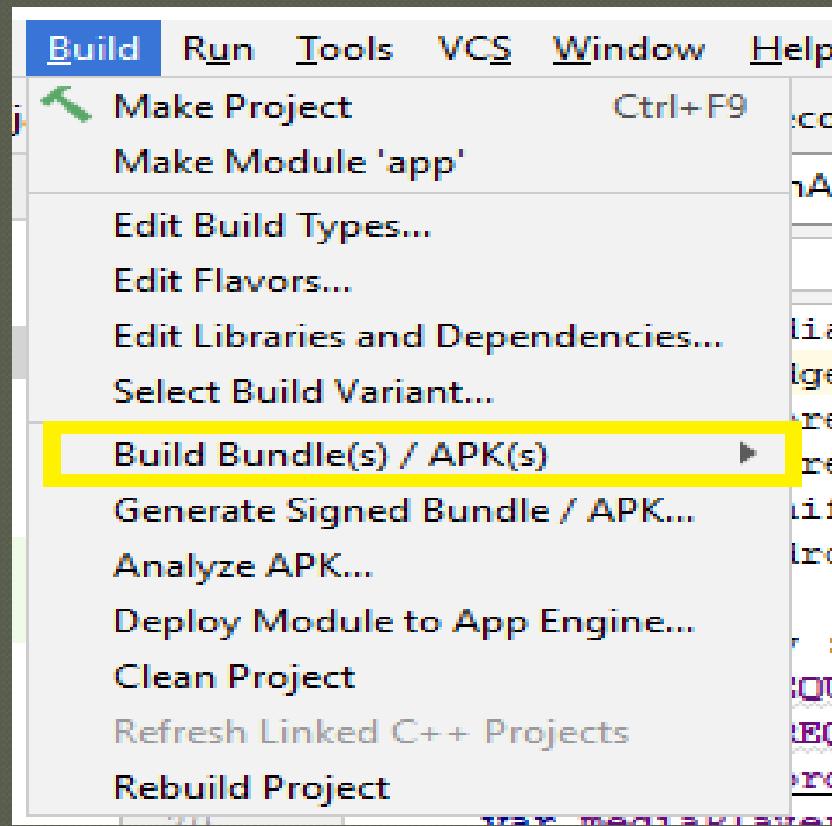
- ◎ **Android Application Package file (.apk)**
- ◎ It's like an executable file
- ◎ Each Android application is compiled and packaged in a single file that includes all the app's code, resources, assets, and manifest file
- ◎ You need an APK to publish on Google Play Store

Main point 1

The publishing process is typically performed after you finish testing your application in a debug environment. Also, as a best practice, your application should meet all your release criteria for functionality, performance, and stability before you begin the publishing process. *Science of Consciousness: Establishing awareness in the home of all the laws of nature results in a life spontaneously lived in accord with natural law.*

Build APK for Testing Purpose not for Deployment

Step 1 : Click Build → Build Bundles(s)/APK(s) APK, will provide APK to test into various Android Device

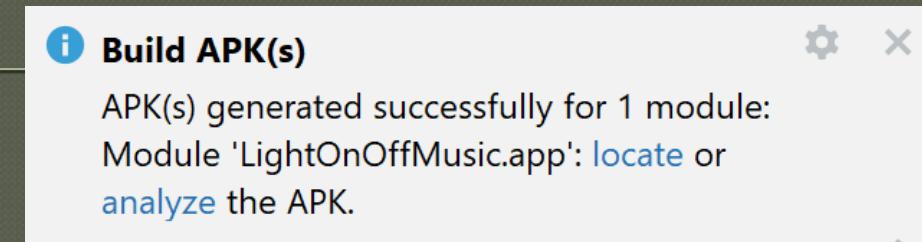


Build APK for Testing Purpose not for Deployment

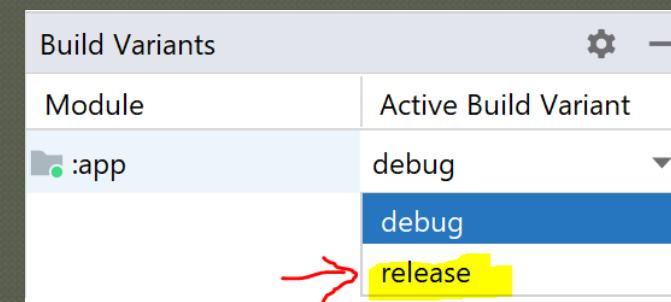
Step 2 : Once the APK is generated you will get this message.

Step 3 : Click on **locate** in Folder to see the APK provided for the debug purpose. Click **analyze** to get the apk and each resources size.

To get the release version, Go to Build → Select Build Variant, then select release. This is an unsigned apk version. To upload it on Google Play Store, you need a Signed apk



Name	Date modified	Type	Size
app-debug	10/19/2022 3:47 PM	APK File	3,213 KB
output	10/19/2022 3:47 PM	JSON File	1 KB



Generating a Signed APK

- To publish your finished application on the Google Play Store, you must generate a signed APK.
- APK is the compiled, executable version of your application.
- Signing it is much like signing your name to a document.
- The signature identifies the app's developer to Google and the users who install your application.

If your application has more than 1000 lines of code add the highlighted part of code in your app before making signed APK.

GravitySensorDemo - [E:\ReMo\Andriod_Coding\Lesson10\GravitySensor\GravitySensorDemo] - app - Android Studio 2.3.2

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

GravitySensorDemo app build.gradle

1: Project 2: Structure 3: Captures 4: Favorites

Gradle files have changed since last project sync. A project sync may be necessary for the IDE to work properly. [Sync Now](#)

app

Gradle Scripts

- build.gradle (Project: GravitySensorDemo)
- build.gradle (Module: app)**
- gradle-wrapper.properties (Gradle Version)
- proguard-rules.pro (ProGuard Rules for app)
- gradle.properties (Project Properties)
- settings.gradle (Project Settings)
- local.properties (SDK Location)

```
1 apply plugin: 'com.android.application'
2
3 android {
4     compileSdkVersion 25
5     buildToolsVersion "25.0.0"
6     defaultConfig {
7         multiDexEnabled true
8         applicationId "com.example.rmohanraj.gravitysensordemo"
9         minSdkVersion 15
10        targetSdkVersion 25
11        versionCode 1
12        versionName "1.0"
13        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
14    }
15    buildTypes {
16        release {
17            minifyEnabled false
18            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
19        }
20    }
21
22    dependencies {
23        compile fileTree(dir: 'libs', include: ['*.jar'])
24        androidTestCompile('com.android.support.test.espresso:espresso-core:2.2.2', {
25            exclude group: 'com.android.support', module: 'support-annotations'
26        })
27        compile 'com.android.support:appcompat-v7:25.3.1'
28        compile 'com.android.support.constraint:constraint-layout:1.0.2'
29        compile 'com.android.support:multidex:1.0.0'
30        compile 'com.android.support:multidex:1.0.0'
31        testCompile 'junit:junit:4.12'
```

Platform and Plugin Updates
Android Studio is ready to update.

TODO Android Monitor Messages Terminal

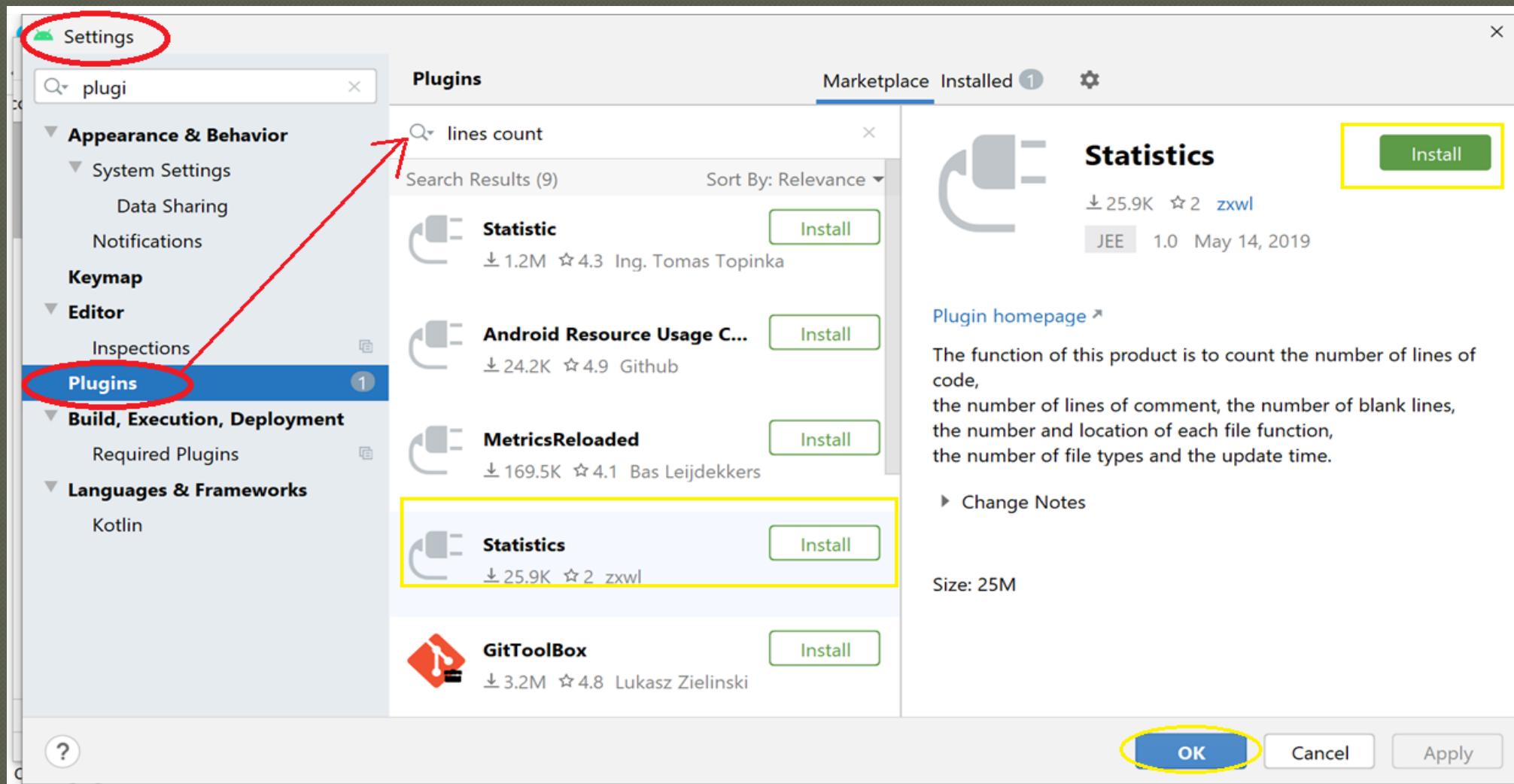
Gradle build finished in 8s 808ms (today 2:27 PM)

Type here to search

e C S W P W A

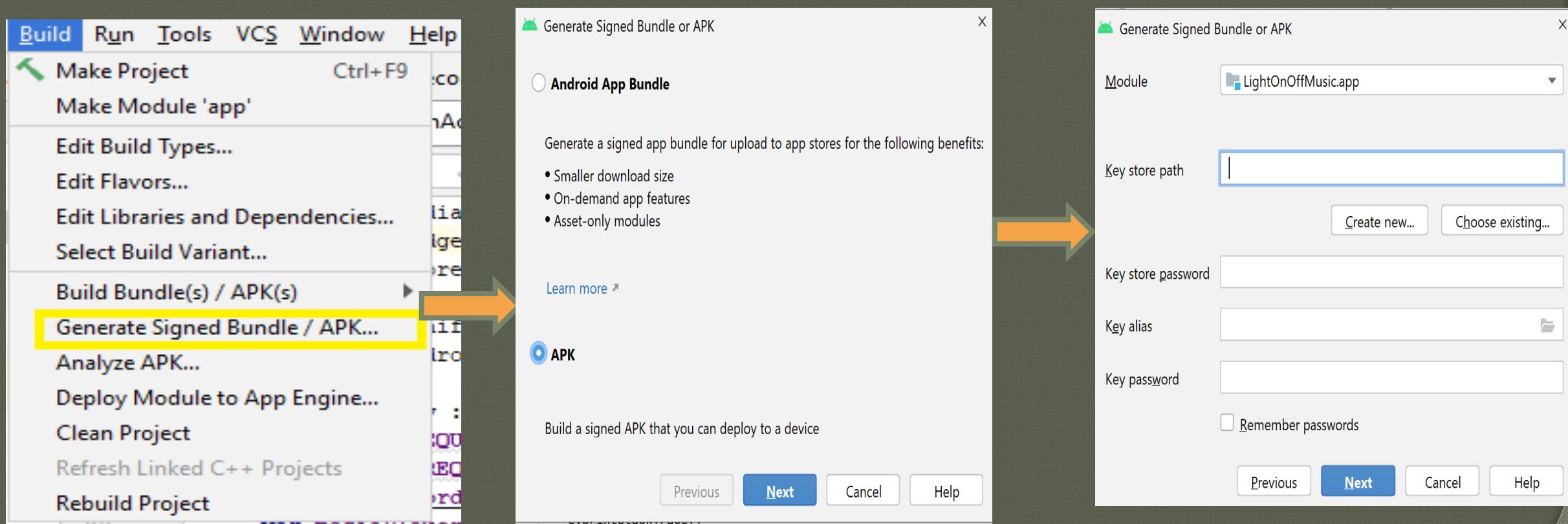
5:45 PM 8/3/2017

Plugin to know the lines count for your App



Generating a Signed APK steps

1. Generate a signed APK from your code by selecting Build Generate Signed APK from the Menu bar to bring up the Generate Signed APK window as shown here and click **create New**.



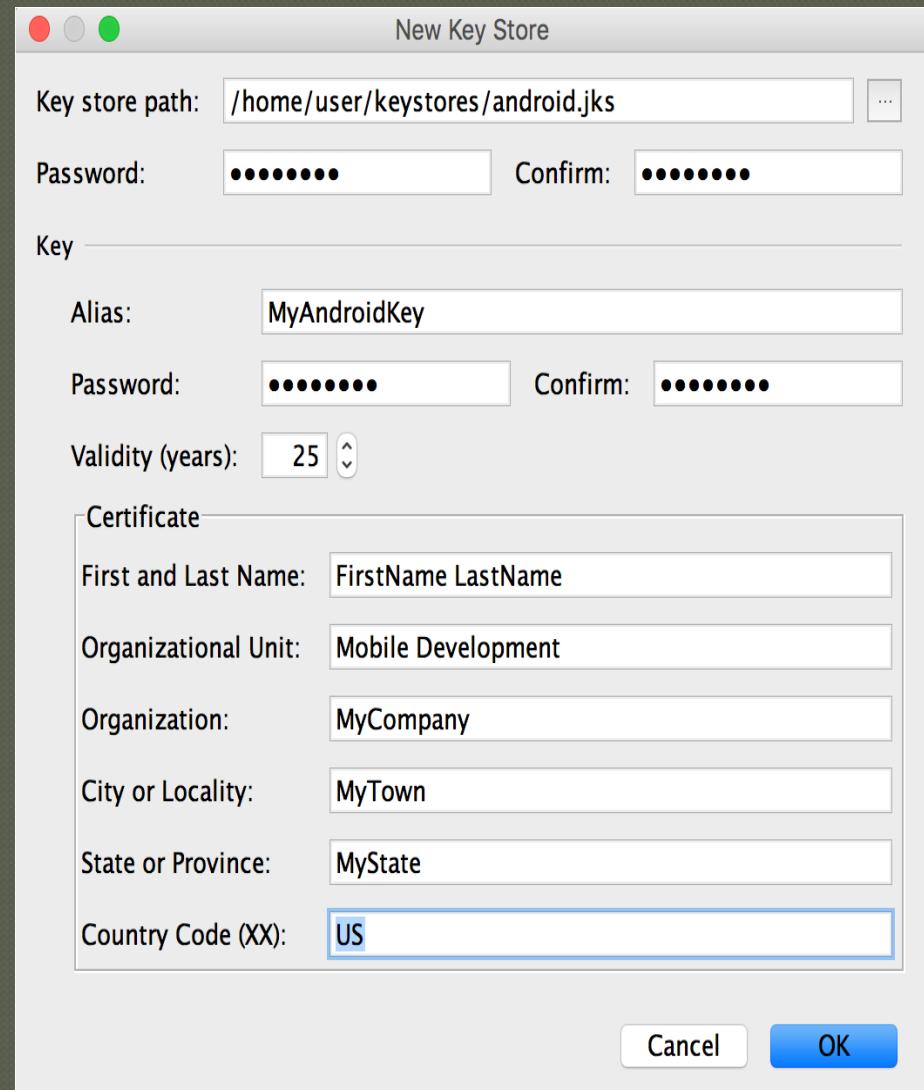
Cont...

2. Assuming you have never published an application from Android Studio, you need to create a new key store. Click the Create New button to display the New Key Store window.

- Signed APK provides two things.
 - a. Signed Key
 - b. APK file.
- The Key is very confidential, it will be useful in future to make any update.
- It should not be shared.
- Signed Key file is stored with the extension .jks(Java Key Store)

Cont..

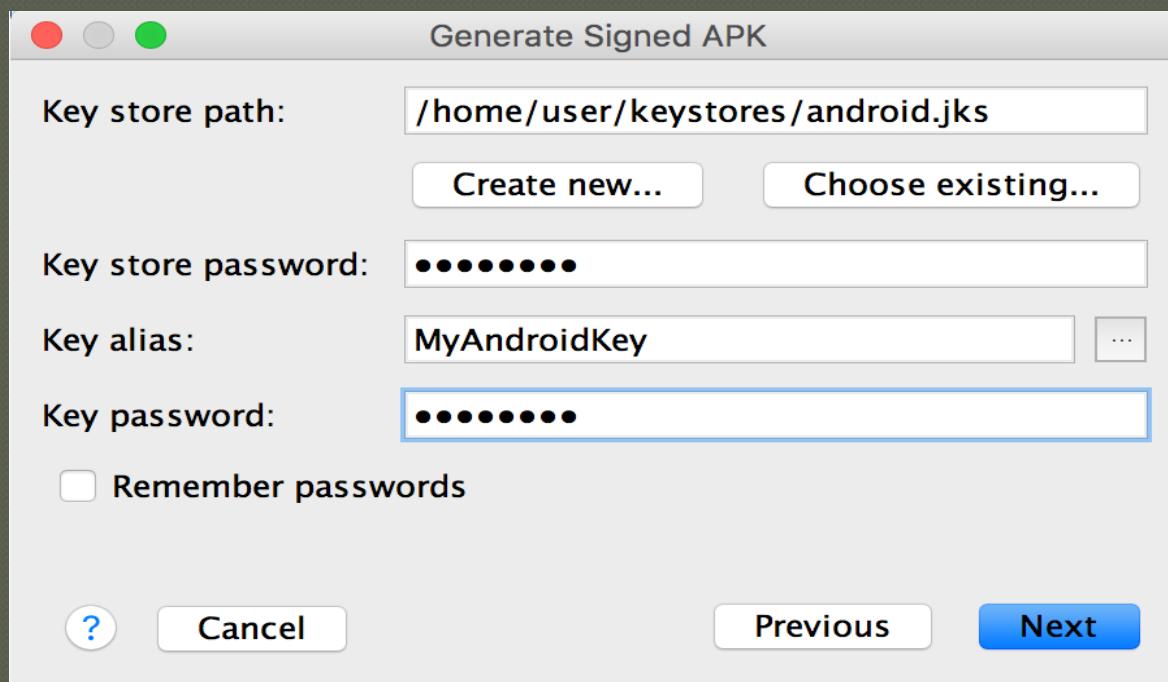
- 3. Fill out all of the information on this form because it pertains to your entity and application. Notice that there are two places for a password. These are the passwords for your key store and your key, respectively. Use the same password in both the places.
- Click OK to return to the Generate Signed APK window.



Cont..

4. Click Next button

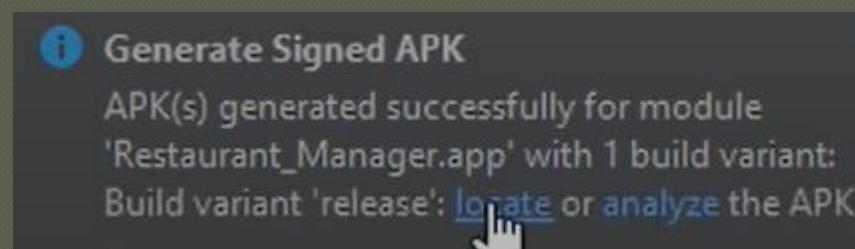
The **Key Alias** is a just a commonplace name that points to a specific certificate. You can create one for each release, or just use the same one for all your apps.



Cont...

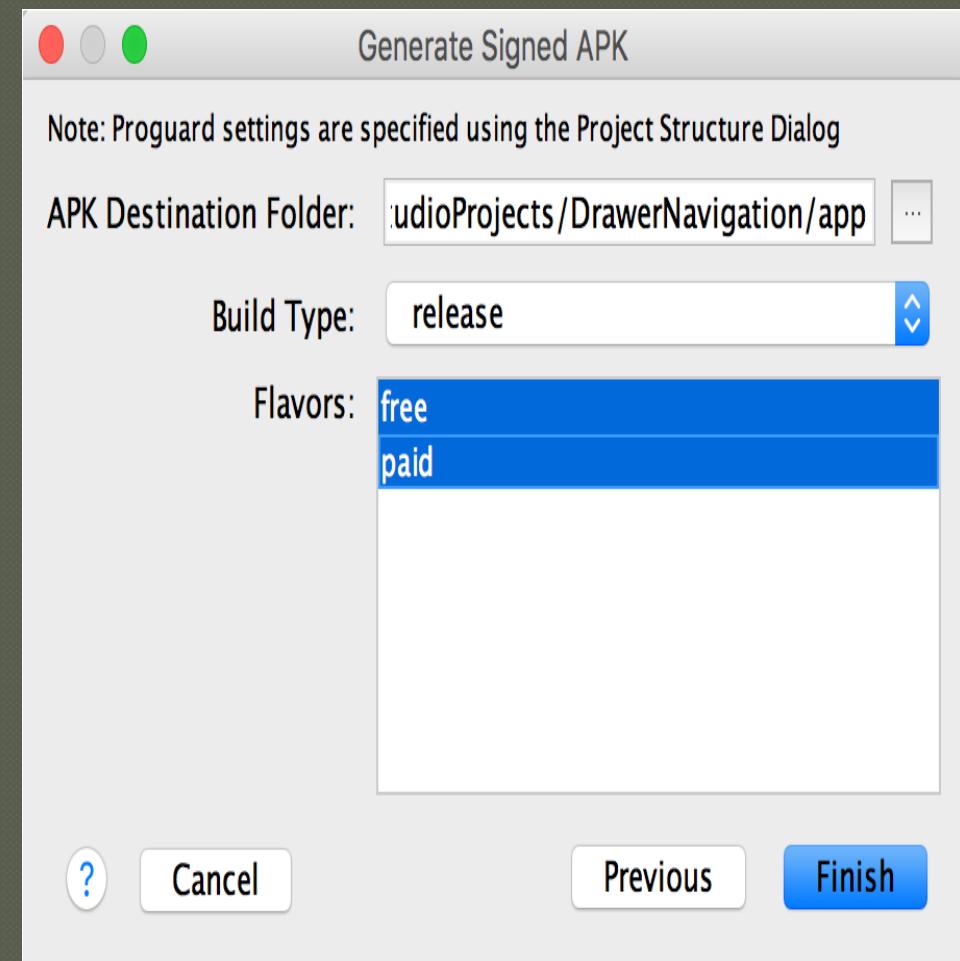
5. On the next window, select a destination for the signed APK(s), select the build type release and choose the product flavor(s), and click **Finish**.

6. You will get, Click on **locate** to see the signed APK



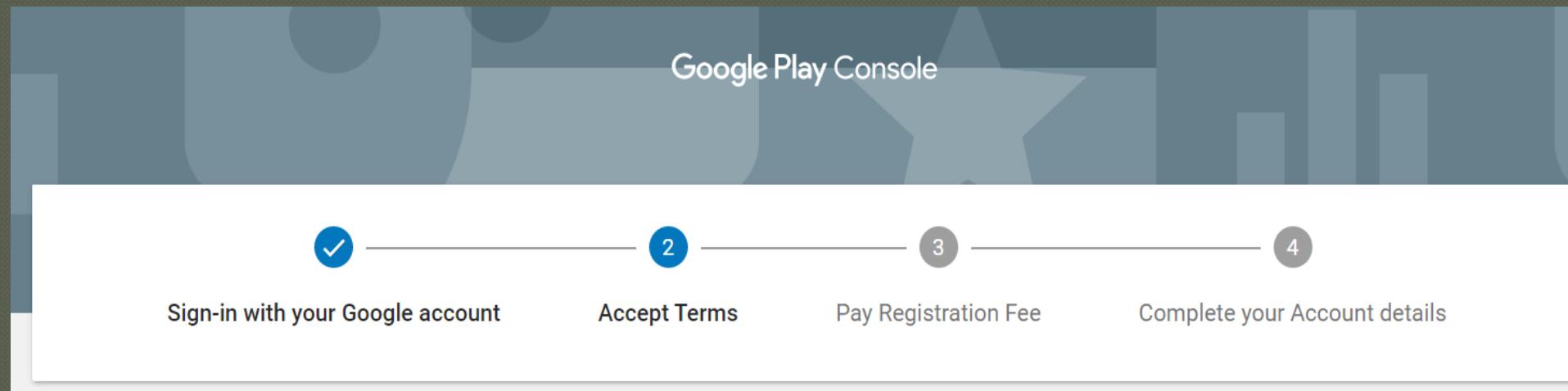
Reference :

<https://developer.android.com/studio/publish/app-signing>



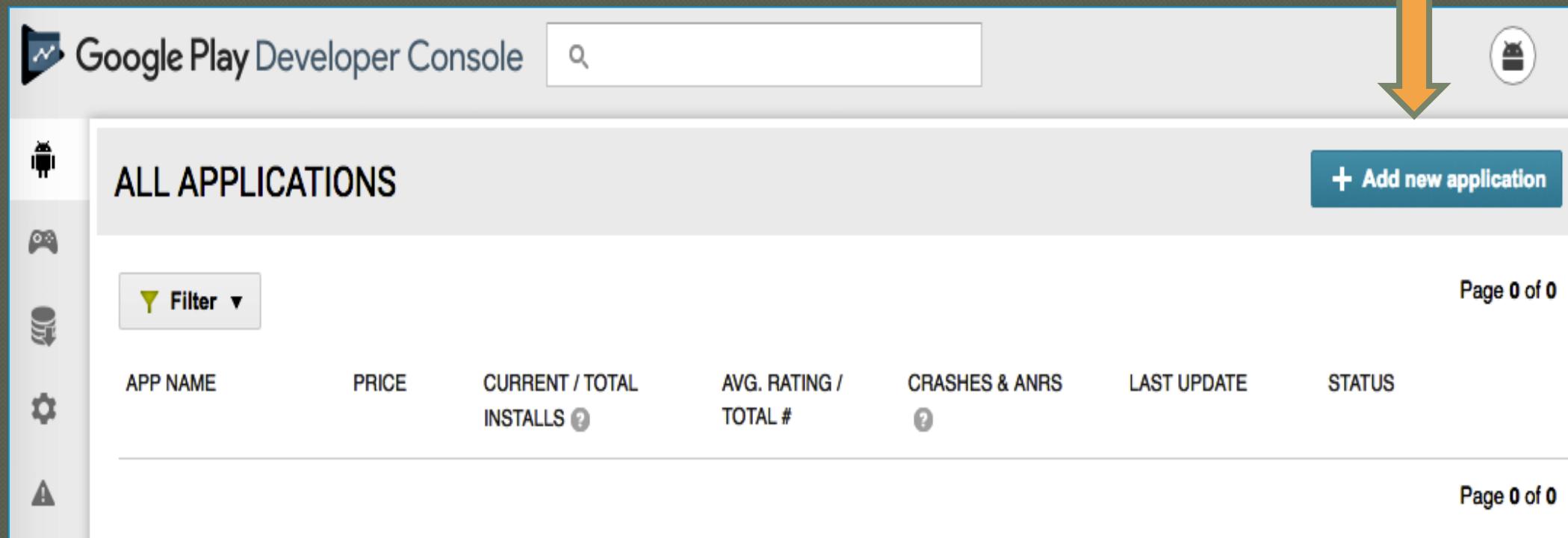
Signup for Google Play

- Now you have a signed APK, you can upload it to the Google Play Store using the developer console at <https://play.google.com/apps/publish/>
- Once you clicked on the link, need to sign in with your google account.
- You have to create a Play store account and pay some amount(\$25) for this account.
- You will get lifetime membership. You can upload N number of Apps.



Google Play Store uploading Steps

1. Click Add new Application/Create App depends on the Google Play store console to upload your APK file.



Continued

- ◎ Step 2: The next step is to finish the Dashboard requirements step by step
 - App categorization and store listing details
 - Select an app category, provide contact details, and set up your store listing.
 - Help users discover your app or game by choosing a category and applying tags to your app or game.
 - Support your app's users by providing contact information.
 - Create a compelling store listing by adding screenshots and a video that showcase your app.
- ◎ Refer the following slides for the sample Dashboard screens, the orders may vary depends the Google play store console updates.

3. Finish the remaining steps in the order how it appeared and fill out the required information from the highlighted part.

The screenshot shows the Google Play Developer Console interface. The top navigation bar includes tabs for 'APK - Test app - Google Play Developer Cons...', 'Google Play', 'Android 7 tutorial for beginners | Udemy', 'apk downloader - Google Search', and 'Apktool - A tool for reverse engineering An...'. Below the navigation is a search bar and a loading indicator. The main content area displays the 'Test app' (com.hussienalrubaie.myapp) in 'DRAFT' mode. On the left, a sidebar lists categories: 'APK' (highlighted with a yellow box), 'In-app Products', 'Services & APIs', and 'Optimization Tips'. Under 'APK', sub-options include 'Store Listing' (with a hand cursor icon), 'Content Rating', and 'Pricing & Distribution'. The central panel shows 'APK' status with 'PRODUCTION' version 1 and 'BETA TESTING' and 'ALPHA TESTING' options. A 'PRODUCTION CONFIGURATION' section features a 'Upload new APK to Production' button. Below it, 'CURRENT APK' was uploaded on Sep 21, 2016, 4:09:06 PM. The 'Supported devices' count is 10786, and 'Excluded devices' count is 0. A table at the bottom lists the single APK entry: Version 1 (1.0), Uploaded on Sep 21, 2016, Status Draft in Prod, and Actions. A 'Switch to advanced mode' button is located in the top right of the central panel. The bottom of the page includes a 'TRANSLATION SERVICE' logo and a date indicator 'Thursday, 21 September 2016'.

3.a. Store Listing Information

Provide Title, Short Description and Full Description about your app.

The screenshot shows the Google Play Developer Console interface. The top navigation bar includes the title "Google Play Developer Console" and the URL "play.google.com". On the left, there's a sidebar with various icons and links: APK, Manage Releases, **Store Listing** (which is highlighted with a yellow circle), Content Rating, Pricing & Distribution, In-app Products, Services & APIs, and Optimization Tips. The main content area is titled "STORE LISTING". It has three sections: "PRODUCT DETAILS", "Title*", "Short description*", and "Full description*". Each section has a language dropdown set to "English (United States) - en-US". The "Title*" field contains "Bluetooth Scanner" (17 of 30 characters). The "Short description*" field contains "Bluetooth Scanner app is used to manage BT, used to get near by BT devices..." (77 of 80 characters). The "Full description*" field contains the same text as the short description (77 of 4000 characters). At the bottom, there are footer links for "Metadata policy", "program policies", and "contact us prior to publishing". There are also "Save draft" and "Publish app" buttons in the top right corner.

3.a. Store Listing Information

Provide minimum of two screenshots of your app in the Graphic Assets is mandatory. Under graphic assets, you can add screenshots, images, videos, promotional graphics, and icons that showcase your app's features and functionality.

The screenshot shows the 'Store Listing' tab selected in the left sidebar. The main area is titled 'PRODUCT DETAILS' and includes language settings for 'English (United States) – en-US' and a 'Manage translations' dropdown. Below this is the 'GRAPHIC ASSETS' section, which contains instructions about localized graphics and a 'Screenshots' section. The 'Screenshots' section includes a note about tablet screenshots and a 'Drop image here' area for uploading screenshots. At the bottom, there are sections for 'Hi-res Icon', 'Feature Graphic', and 'Promo Graphic' with their respective size requirements.

APK

Manage Releases

Store Listing

Content Rating

Pricing & Distribution

In-app Products

Services & APIs

Optimization Tips

STORE LISTING

PRODUCT DETAILS

English (United States) – en-US

Manage translations ▾

GRAPHIC ASSETS

If you haven't added localized graphics for each language, graphics for your default language will be used.
[Learn more about graphic assets.](#)

Screenshots *

Default – English (United States) – en-US
JPEG or 24-bit PNG (no alpha). Min length for any side: 320px. Max length for any side: 3840px.
At least 2 screenshots are required overall. Max 8 screenshots per type. Drag to reorder or to move between types.

For your app to be showcased in the 'Designed for tablets' list in the Play Store, you need to upload at least one 7-inch and one 10-inch screenshot. If you previously uploaded screenshots, make sure to move them into the right area below.
[Learn how tablet screenshots will be displayed in the store listing.](#)

Please check out our [Impersonation and Intellectual Property policy](#) to avoid common violations.

Phone Tablet Android TV Android Wear

+ Add screenshot
Drop image here

Hi-res Icon *
Default – English (United States) – en-US
512 x 512
32-bit PNG (with alpha)

Feature Graphic *
Default – English (United States) – en-US
1024 w x 500 h
JPG or 24-bit PNG (no alpha)

Promo Graphic
Default – English (United States) – en-US
180 w x 120 h
JPG or 24-bit PNG (no alpha)

3.a. Store Listing Information

- The next step is to provide Hi-res-icon for your app with specified 512 x512 size. You can create an icon using falticon.com, then upload your icon.
- You should add Feature graphic with 1024 w x 500 h size. Other sizes are not allowed.(like Banner)

Hi-res icon * Default – English (United States) – en-US 512 x 512 32-bit PNG (with alpha)	Feature Graphic * Default – English (United States) – en-US 1024 w x 500 h JPG or 24-bit PNG (no alpha)
	

TV Banner Default – English (United States) – en-US 1280 w x 720 h JPG or 24-bit PNG (no alpha)	Daydream 360 degree stereoscopic image Default – English (United States) – en-US 4096 w x 4096 h JPG or 24-bit PNG (no alpha)
---	---

3.b. Content Rating

Your app's content rating level inform users of it's maturity level. Click on the Continue button.

Google Play Developer Console

Bluetooth Scanner
cubexsoft.btex
DRAFT Delete app

APK

Manage Releases

Store Listing

Content Rating

Developer responsibilities:

- Complete the content rating questionnaire for each new app submitted to Developer Console, for all existing apps that are active on Google Play, and update where there has been a change to app content or features that would affect the responses to the questionnaire.
- Provide accurate responses to the content rating questionnaire. Misrepresentation of your app's content may result in removal or suspension from Google Play.

Your rating will be used to:

- Inform consumers about the age appropriateness of your app.
- Block or filter your content in certain territories or to specific users where legally required.
- Evaluate your app's eligibility for special developer programs.

The content rating questionnaire and the new Content Ratings Guidelines are a condition of your participation in the Google Play store. [Learn more](#)

Continue

3.b. Content Rating

Fill out the Email address and Confirmation email address, then choose the category of your app. Once you choose the category you will get another screen.

Manage Releases

Store Listing

Content Rating

Pricing & Distribution

In-app Products

Services & APIs

Optimization Tips

Welcome to the Content Rating Questionnaire

The Google Play content rating system for apps and games is designed to deliver reputable, locally relevant ratings to users around the world. It includes official ratings from the International Age Rating Coalition (IARC) and its participating bodies. Get started by entering the email address you would like to use for rating related communications.

Email address * test@gmail.com
Please enter a valid email address.

Confirm email address * test@gmail.com

Select your app category

 **REFERENCE, NEWS, OR EDUCATIONAL**
The primary purpose of the app is to present factual information in a neutral way, alert users to current events, or educate users. Examples include Wikipedia, BBC News, Dictionary.com, and Medscape. Apps that mainly focus on sexual advice or instruction (such as "iKama" or "Best Sex Tips") should be categorized as "Entertainment" apps and not listed here. [Learn more](#)

 **SOCIAL NETWORKING, FORUMS, AND UGC SHARING**
The primary purpose of the app is to enable users to share content or communicate with large groups of people. Examples include 9Gag, Google Plus, YouTube, Twitter. Apps that only facilitate communication between a limited number of people (such as Skype) should be categorized as "Communication" apps and not listed here. [Learn more](#)

3.b. Content Rating

Answer the questions appeared on the screen and click Save questionnaire button. Then click Calculate rating button, you will get next screen.

APK

- Manage Releases
- Store Listing
- Content Rating**
- Pricing & Distribution
- In-app Products
- Services & APIs
- Optimization Tips

CONTENT RATING

Please complete the questionnaire so that we can calculate your app rating.

 REFERENCE, NEWS, OR EDUCATIONAL
App is a news, reference, or educational app. [Edit Category](#)

VIOLENCE

Can the app contain violent material? [Learn more](#)
Please note that this question does not refer to user-generated content.

Yes No

SEXUALITY

Can the app contain sexual material or nudity (except in a natural or scientific setting)? [Learn more](#)
Please note that this question does not refer to user-generated content.

Yes No

LANGUAGE

Can the app contain any potentially offensive language? [Learn more](#)
Please note that this question does not refer to user-generated content.

Yes No

MISCELLANEOUS

Does the app natively allow users to interact or exchange content with other users through voice communication, text, or sharing images?

Yes No

Does the app share user-provided personal information with third parties? [Learn more](#)

Yes No

Does the app share the user's current physical location to other users? [Learn more](#)

Yes No

Does the app allow users to purchase digital goods? [Learn more](#)

Yes No

[Calculate rating](#) [Save questionnaire](#)

3.b. Content Rating

This screen give the information about different country rating. Finally click Apply rating button in the bottom of the screen.

The screenshot shows the 'Content Rating' section of the App Store Connect dashboard. On the left, a sidebar lists navigation options: Manage Releases, Store Listing, Content Rating (which is highlighted with a yellow oval), Pricing & Distribution, In-app Products, Services & APIs, and Optimization Tips. The main content area displays the 'REFERENCE, NEWS, OR EDUCATIONAL' rating category, which is described as an app being a news, reference, or educational app. It shows a blue circular icon with a white document symbol. Below this, there's a 'CALCULATED RATING' section with a 'Learn more' link, followed by a 'Rating System' section. Further down, there are sections for 'Classificação Indicativa (ClassInd) Brazil', 'Entertainment Software Rating Board (ESRB) North America', and 'Pan-European Game Information (PEGI) Europe'. At the very bottom of the page, there is a prominent blue 'Apply rating' button, which is also circled in yellow to indicate it as the final step.

3.c. Pricing & Distribution

- You can publish your app as free to download or priced.
- Free apps can be downloaded by any Android user in Google Play.
- Priced apps can be downloaded only by users who are in a country that supports paid downloads and have registered a form of payment in Google Play, such as a credit card or Direct Carrier Billing.
- Priced apps can be changed to free at any time.
- If you do decide on a priced app or are selling in-app products, you'll need to setup a **Google payments merchant account** before you can publish.

3.c. Pricing & Distribution

Choose the countries for your app.

Answer the Mandatory option and click Save draft button on the top.

The screenshot shows the Google Play Developer Console interface. At the top, the app title 'Bluetooth Scanner' and developer name 'cubexsoft.blx' are displayed. Below the title, there are buttons for 'READY TO PUBLISH' and 'Delete app'. On the right side of the header are 'Save draft' and 'Publish app' buttons. A notification bell icon with a '2' is also present.

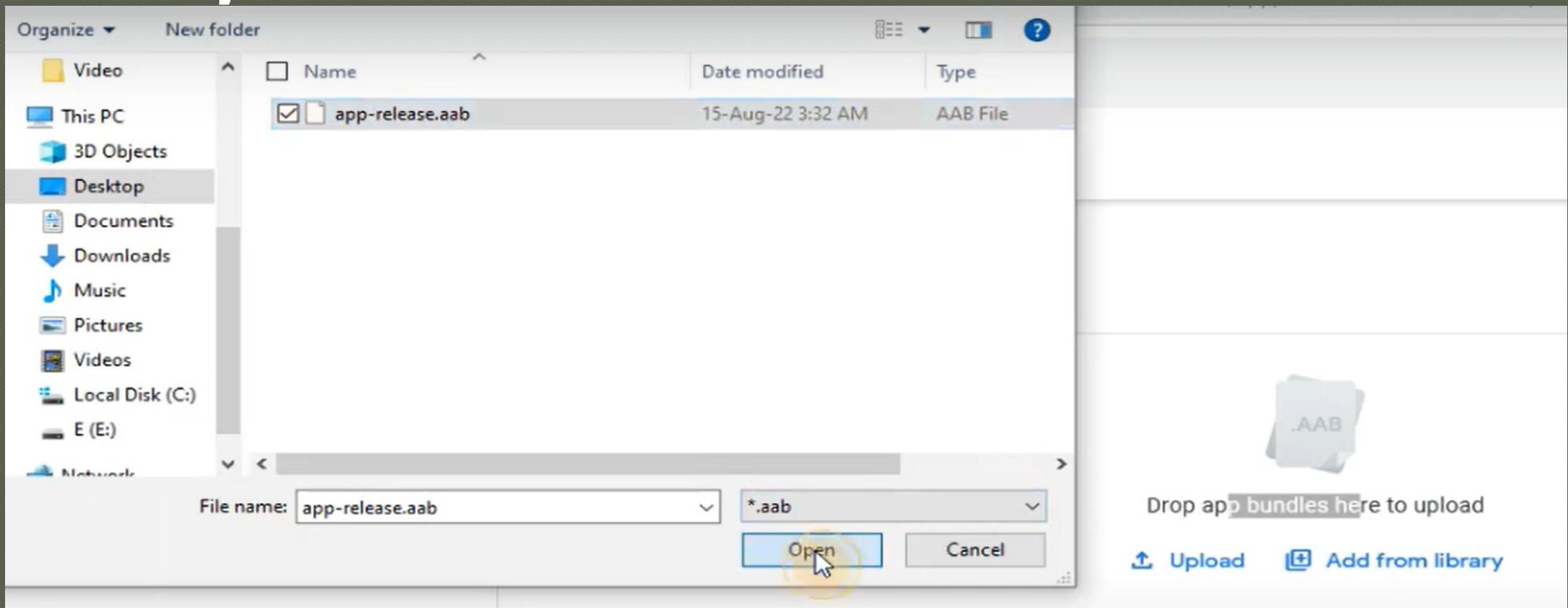
The left sidebar contains navigation links: APK, Manage Releases, Store Listing, Content Rating, **Pricing & Distribution** (which is highlighted with a yellow oval), In-app Products, Services & APIs, and Optimization Tips.

The main content area is titled 'PRICING & DISTRIBUTION'. It features several distribution icons: 'Designed for Families' (star), 'Google Play for Education' (blue circle with 'EDU'), 'Google Play for Work' (red briefcase), 'Daydream' (cloud), 'Android Wear' (wristwatch), 'Android TV' (TV), and 'Android Auto' (car). Below these icons, the status 'This application is Free' is shown, with a note that publishing paid applications requires a merchant account.

The 'Countries' section indicates the app is distributed to '141 countries + Rest of the world'. A 'Manage countries' button is available. A dropdown menu titled 'SELECT ALL COUNTRIES' lists several countries with checkboxes: Turkmenistan, Uganda, Ukraine, United Arab Emirates, United Kingdom, and United States. 'Show options' buttons are located next to the United Kingdom and United States entries.

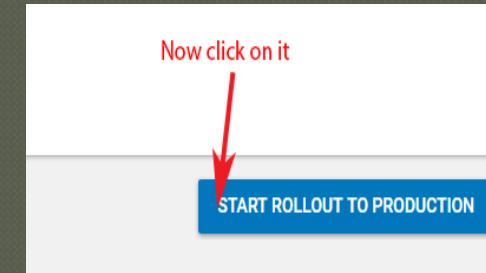
Continued

- Then upload your signed apk file to the console. Next Review your release.



Manage Release

- After review click on Start Rollout to production. Now you need to confirm. After confirm you will need to wait for few hours for approval.



Publish and update App useful resources

- Publishing guidelines link

- <https://support.google.com/googleplay/android-developer/#topic=3450769>
- <https://developer.android.com/distribute/best-practices/launch>

- To update the apps on the Play Store read this resource.

- <https://support.google.com/googleplay/answer/113412?hl=en>

Main Point 2

When developing on the Android operating system, you use the Android application package (APK) format to distribute apps. Android requires that APKs are digitally *signed* with a certificate before they can be installed. The certificate is used to identify the author of the app. *Science of Consciousness: In a similar way, creation itself is structured in layers; the activity at each layer has its own unique set of governing laws; laws that pertain to one level or layer may no longer be applicable at another level.*

UNITY CHART

CONNECTING THE PARTS OF KNOWLEDGE WITH THE WHOLENESS OF KNOWLEDGE

Purification leads to progress

1. During the preparation step you build a release version of your application, which users can download and install on their Android-powered devices.
2. During the release step you publicize, sell, and distribute the release version of your application to users. The infinite reservoir of creative intelligence at the source of thought can be open to experience.

3. **Transcendental Consciousness:** TC is the home of all the laws of nature, the home of "right action".
4. ***Impulses within the Transcendental field:*** *These impulses of change move within the eternally stable field of pure consciousness.*
5. ***Wholeness moving within Itself:*** *Action in the state of Unity Consciousness is spontaneously right and uplifting to the creation as a whole.*

