Chapter 8

8. Publishing, Distributing, Monetizing, and Promoting Applications

8.1. Signing and Publishing Android Application

Android applications are distributed as Android package files (.APK). In order to be installed ona device or emulator, Android packages need to be signed.

During development, your applications will be signed using a debug key that is automaticallygenerated by the ADT tools. Before distributing your application beyond your testing environment,you must compile it as a release build and sign it using a private release key — typically using a selfsignedcertificate.

The JDK includes theKeytool and Jarsigner command-line tools necessary to create a newkeystore/signing certificate, and to sign your APK, respectively. Alternatively, you can use the Generate signed bundle/apk wizard, as described in the next section.

The importance of maintaining the security of your signing certificate can't beoverstated. Android usesthis certificate as the means of identifying the authenticityof applicationupdates,andapplyinginter-processsecurityboundariesbetweeninstalledapplications.Using a stolen key, a third party could sign and distribute applications thatmaliciously replace your authentic applications.Similarly, your certificate is the only way you can upgrade your applications.If you lose your certificate, it is impossible to perform a seamless update on adevice or from within Google Play. In the latter case, you would need to createa new listing, losing all the reviews, ratings, and comments associated with yourprevious package, as well as making it impossible to provide updates to theexisting users of your application.

1. Create Your Android App

First thing for uploading your App on Play Store is to Develop your App and make it Store-Ready.

Use Proper theme, colors, graphics, strings and icons.

2. Make its Signed Apk(Using Release Keystore)

To generate keystores for signing Android apps at the command line, use:

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

A debug keystore which is used to sign an Android app during development needs a specific alias and password combination as dictated by Google. To create a debug keystore, use:

```
$ keytool -genkey -v -keystoredebug.keystore -storepass android -alias androiddebugkey -keypass android -keyalg RSA -keysize 2048 -validity 10000
```

Keystore name: “debug.keystore”

Keystore password: “android”

Key alias: “androiddebugkey”

Key password: “android”

To generate signed bundle/apk using Android Application Wizard

- Go to Build→Generate Signed Bundle/APK→APK→ and the go through the steps to finish the signing process.

3. Login to your Gmail Account and visit this link: https://play.google.com/apps/publish/

4. Create a Merchant Account

Note: It will charge you once in a lifetime fee i.e 25$. Just do it to start uploading your first app.

5. Mandatory Fields needed in App Listing: Title, Short Desc, Full Desc, App Screenshots(JPEG or 24-bit PNG (no alpha))(Min-2,Max-8)(Min-320px,Max-3840px), Hi-res icon(512 x 512)(32-bit PNG (with alpha)), Feature Graphic(1024 w x 500 h)(JPG or 24-bit PNG (no alpha)), App Type, Category, Content Rating, Developer/Company Email, Privacy Policy Url, And some other details

How to Publish an App on Google Play?

With over 1 billion monthly active users, Google Play is arguably one of the largest platforms for distributing, promoting, and selling Android apps.

For newbies and first-timers, publishing an app on Google Play can seem like an intimidating process. Like any other app store, the platform comes with its own set of rules, regulations, and procedures, and you need to understand how it works in order to avoid any future issues.

Here is a step-by-step guide on how to successfully publish your app on the Google Play Store.

Step 1: Create a Developer Account

Before you can publish any app on Google Play, you need to create a Developer Account. You can easily sign up for one using your existing Google Account.The signup process is fairly straightforward, and you’ll need to pay a one-time registration fee of $25. After you’ve reviewed and accepted the Developer Distribution Agreement, you can proceed to make the payment using your credit or debit card.To finish the sign up process, fill out all your necessary account details, including your Developer Name, which will be visible to your customers on Google Play. You can always add more details later.Also, do remember that it can take up to 48 hours for your registration to be fully processed.

Step 2: Plan to Sell? Link Your Merchant Account

If you want to publish a paid app or plan to sell in-app purchases, you need to create a payments center profile, i.e. a merchant account. Here’s how you can do that:

1. Sign in to your Play Console
2. Click on Download Reports – Financial
3. Select ‘Set up a merchant account now’
4. Fill out your business information

Once you create the profile, it will be automatically linked to your developer account.A merchant account will let you manage your app sales and monthly payouts, as well as analyze your sales reports right in your Play Console.

Step 3: Create an App

Now that you have set up your Play Console, you can finally add your app. Here’s how to do that:

1. Navigate to the ‘All applications’ tab in the menu
2. Click on 'Create Application'

3. Select your app’s default language from the drop-down menu
4. Type in a title for your app
5. Click on “Create”

The title of your app will show on Google Play after you’ve published. Don’t worry too much about it at this stage; you can always change the name later.

After you’ve created your app, you’ll be taken to the store entry page. Here, you will need to fill out all the details for your app’s store listing.

Step 4: Prepare Store Listing

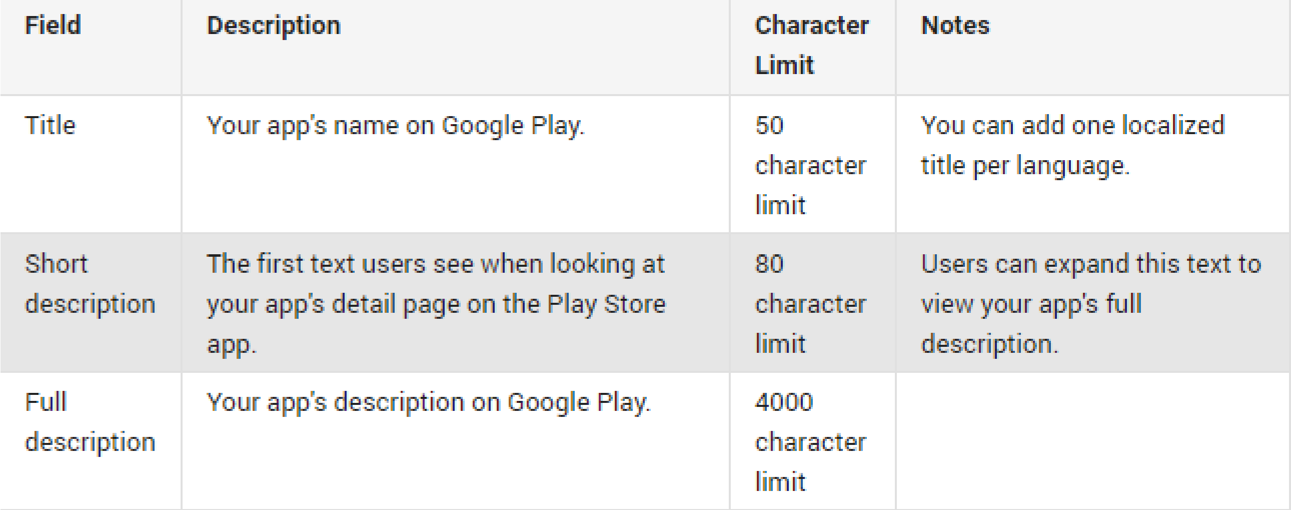
Before you can publish your app, you need to prepare its store listing. These are all the details that will show up to customers on your app’s listing on Google Play.

Note: You don’t necessarily have to complete this step before moving on to the next one. You can always save a draft and revisit it later when you’re ready to publish.

The information required for your store listing is divided into several categories:

Product Details

There are three fields here that you need to fill out:



Field	Description	Character Limit	Notes
Title	Your app's name on Google Play.	50 character limit	You can add one localized title per language.
Short description	The first text users see when looking at your app's detail page on the Play Store app.	80 character limit	Users can expand this text to view your app's full description.
Full description	Your app's description on Google Play.	4000 character limit	

Your app’s title and description should be written with a great user experience in mind.Use the right keywords, but don’t overdo it. Make sure your app doesn’t come across as spamor promotional, or it will risk getting suspended on the Play Store.

Graphic Assets

Under graphic assets, you can add screenshots, images, videos, promotional graphics, and icons that showcase your app’s features and functionality.Some parts under graphic assets are mandatory, like screenshots, a feature graphic, and a high-resolution icon. Others are optional, but you can add them to make your app look more attractive to users.There are specific requirements for each graphic asset that you upload, such as the file format and dimensions. You can read more about each requirement here.

Languages & Translations

You can also add translations of your app’s information in the store listing details, along with in-language screenshots and other localized images.There’s also an option for users to view automated translations of your app’s information using Google Translate, in case you don’t add your own translations.

Categorization

This part requires you to select the appropriate type and category your app belongs to. From the drop-down menu, you can pick either app or game for the application type.There are various categories for each type of app available on the Play Store. Pick the one your app fits into best. In order to rate your content, you'll need to upload an APK first. You can skip this step for later.

Contact Details

This part requires you to enter contact details to offer your customers access to support regarding your app.

You can add multiple contact channels here, like an email, website, and phone number, but providing a contact email is mandatory for publishing an app.

Privacy Policy

For apps that request access to sensitive user data or permissions, you need to enter a comprehensive privacy policy that effectively discloses how your app collects, uses, and shares that data.You must add a URL linking to your privacy policy in your store listing and within your app. Make sure the link is active and relevant to your app.

You're now done with the store listing. Go ahead and click on 'Save Draft' to save your details. You can always skip some steps and come back to them later before you publish your app.

Step 5: Upload APK to an App Release

Now that you have prepared the ground to finally upload your app, it's time to dig out your APK file.

The Android Package Kit (or APK, for short) is the file format used by the Android operating system to distribute and install apps. Simply put, your APK file contains all the elements needed for your app to actually work on a device.Google offers you multiple ways to upload and release your APK. Before you upload the file, however, you need to create an app release.

To create a release, select the app you created in Step 3. Then, from the menu on the left side, navigate to 'Release management' -> 'App releases.'

Here, you need to select the type of release you want to upload your first app version to. You can choose between an internal test, a closed test, an open test, and a production release.The first three releases allow you to test out your app among a select group of users before you make it go live for everyone to access.

This is a safer option because you can analyze the test results and optimize or fix your app accordingly if you need to before rolling it out to all users.However, if you create a production release, your uploaded app version will become accessible to everyone in the countries you choose to distribute it in.

Once you've picked an option, click on 'Create release.'Next, follow the on-screen instructions to add your APK files, and name and describe your release.After you're done, press Save.

Step 6: Provide an Appropriate Content Rating

If you don't assign a rating to your app, it will be listed as 'Unrated'. Apps that are 'Unrated' may get removed from Google Play.To rate your app, you need to fill out a content rating questionnaire. You can access it when you select your app in the Play Console, and navigate to 'Store presence' – 'Content rating' on the left menu.Make sure you enter accurate information. Misrepresentation of your app's content can lead to suspension or removal from the Play Store.An appropriate content rating will also help you get to the right audience, which will eventually improve your engagement rates.

Step 7: Set Up Pricing & Distribution

Before you can fill out the details required in this step, you need to determine your app's monetization strategy.Once you know how your app is going to make money, you can go ahead and set up your app as free or paid.Remember, you can always change your app from paid to free later, but you cannot change a free app to paid. For that, you'll need to create a new app and set its price.

You can also choose the countries you wish to distribute your app in, and opt-in to distribute to specific Android devices and programs too.

Step 8: Rollout Release to Publish Your App

You're almost done. The final step involves reviewing and rolling out your release after making sure you've taken care of everything else.Before you review and rollout your release, make sure the store listing, content rating, and pricing and distribution sections of your app each have a green check mark next to them.

Once you're sure you've filled out those details, select your app and navigate to 'Release management' – 'App releases.' Press 'Edit release' next to your desired release, and review it.Next, click on 'Review' to be taken to the 'Review and rollout release' screen. Here, you can see if there are any issues or warnings you might have missed out on.Finally, select 'Confirm rollout.' This will also publish your app to all users in your target countries on Google Play.

8.2. Distributing Applications

One of the advantages of Android's open ecosystem is the freedom to publish and distribute your applications however, and wherever, you choose. The most common and popular distribution channel is Google Play; however, you are free to distribute your applications using alternative markets, your own website, social media, or any other distribution channel.

In addition to Google Play, there are several alternatives of varying reach, including OEM and carrier pre-installs, the Amazon App Store, and carrier-specific stores.

When distributing your application, it's important to note that application package names are usedas unique identifiers for each application. As a result, each application — including variations thatyou plan to distribute separately — must each have a unique package name. Note that the filenameof your APK does not have to be unique — it will be discarded during the installation process (onlythe package name is used).

8.3. MonetizingApplication

As an open ecosystem, Android enables you to monetize your applications using whatever mechanism you choose.

If you choose to distribute and monetize your applications using Google Play, three options typically are available:

- ✓ **Paid applications** — Charge users an upfront fee before they download and install your application.
- ✓ **Free applications with In-App Billing (IAB)** — Make the download and installation of the application free, but charge within the application for virtual goods, upgrades, and other value-adds.
- ✓ **Advertising-supported applications** — Distribute the application for free, and monetize it by displaying advertising.

Although paid applications and advertising-supported applications are the traditional mechanisms for monetizing mobile applications, IAB has emerged as an extremely effective alternative. If you choose to charge for your applications on Google Play, either through upfront charges or IAB,the revenue is split between you and Google Play in the form of a transaction fee. The revenue split is set at 70 percent for the developer and 30% for Google play.

In order to use either approach, you must first create a Google Checkout Merchant Account, youcan do this from your Android publisher account. Your application listings will then include the option to set a price for the application and the items sold using IAB.

In each case you are the application distributor and merchant of record, so you are responsible for any legal or taxation obligations associated with the sale of your application, subject to the terms described in the DDA.

You can also monetize your application using in-app advertising. The specific process required to setup advertising within your application will vary depending on the ads provider you choose. The general process could be described as follows:

a) Create a publisher account.

b) Download and install the associated ads SDK.

c) Update your Fragment or Activity layouts to include an add banner.

It’s important to ensure that any ads included within your application are as unobtrusive as possible and don’t detract significantly from the user experience of your application. It’s also important toensure that your user interaction model doesn’t encourage accidental clicks on the ad banner. In many cases, developers have chosen to offer a paid alternative (either using up-front paymentor IAB) to allow users to eliminate ad banners from their applications.

8.4. Application Marketing and Promotion Strategies

The first step in effectively marketing and promoting your application is ensuring that you providethe full set of high-quality assets for your Google Play listing. Several promotional opportunities are available within Google Play.However, with more than 2.6million other applications available, it’s important that you consideralternative avenues for marketing and promotion rather than simply launching your application.

While your marketing and promotion strategies will vary widely depending on your goals andbudget, the following list details some of the most effective techniques to consider:

- ✓ **Offline cross promotion** — If you have a significant offline presence (such as a stores orbranches), or a large media presence (such as within newspapers, magazines, or on TV),cross promoting your application through those channels can be a particularly effective wayto increase awareness and help to ensure users trust the download. Traditional advertisingtechniques such as TV and newspaper advertisements can be extremely effective in raisingawareness of your application.
- ✓ **Online cross promotion** — If you have a significant web presence, promoting your applicationthrough direct links to Google Play can be an effective way to drive downloads. Ifyourapplication provides a better user experience than your mobile website, you can detectbrowservisitors from Android devices and direct them to Google Play to download yournativeapp.
- ✓ **Third-party promotion** — Distributing a promotional video on YouTube and leveragingsocial networks, blogs, press releases, and online review sites can help provide positive wordof mouth.
- ✓ **Online advertising** — Online advertising using in-app ad networks (such as AdMob) or traditionalsearch-based advertising (such as Google AdWords) can drive significant impressionsanddownloads for your application.

Application Launch Strategies

Ratings and reviews can have a significant impact on your application's ranking in category lists andwithin Google Play search results. As a result, it can be difficult to recover from a poor launch. Thefollowing list describes some of the strategies you can use to ensure a successful launch:

- ✓ **Iterate on features not quality-** A poorly implemented but feature-rich application willreceive worse reviews than a well-polished application that doesn't do everything. If you areusing an agile approach of releasing early and often, ensure each release is of the same highquality, adding new features as part of each release. Similarly, each release should be morepolished and stable than the last.
- ✓ **Create high quality Google Play assets -** The first impression your application makesis through its appearance in Google Play. Maximize the likelihood of that impressionresulting in an installation by creating assets that represent the quality of yourapplication.
- ✓ **Be honest and descriptive-** Disappointed users who find your application is not as it wasdescribed are likely to uninstall it, rate it poorly, and leave negative comments.

Promotion Within Google Play

In addition to the effect of reviews, downloads, and installs will have on your Google Play listing,there are several automated and curated lists that are used by the Google Play editorial team tohighlight high quality applications.Additionally, a small number of applications are chosen as "featured" applications, receiving priorityplacementinGooglePlay.

Featuredapplicationstypicallyreceivea significant boostindownloadnumbers,makingfeaturinghighlyprizedgoal.While the criteria used to determine which applications become featured is not publicly available,therearecertaincriteriathathavegenerallybecomeassociatedwithfeaturedapplications,including:

- ✓ **High quality and innovative-** The featured applications in Google Play act as a showcasefor the platform. As a result, the easiest way to be featured is by creating a high qualityapplication that is useful and innovative.
- ✓ **A high degree of fit-and-finish -** The listings for featured applications include all the requisite promotional assets, while the applications themselves have few bugs and a highquality user interface.
- ✓ **Broad device and platform support-** Featured applications typically support a broad rangeof device types and platform versions, including both handsets and tablets.
- ✓ **Use of newly released features-** Applications that leverage hardware features and softwareAPIs from new Android platform releases offer an opportunity for the Google Play team tohighlight those new features to reviewers and early adopters.
- ✓ **Consistency with the platform user experience-** Featured applications provide a compellinguser experience that is consistent with the UI and interaction models offered by the Androidplatform.

Analytics and Referral Tracking

Mobile application analytics packages, such as Google Analytics and Flurry, can be effective tools for you to better understand who is using your application and how they are using it. Understanding this information can help you make objective decisions on where to focus your development resources. While the statistics provided by the Android Developer Console offer valuable insight into your users' language, country, and handsets, using detailed analytics can provide a much richer source of information, from which you can discover bugs, prioritize your feature list, and decide where best to allocate your development resources.

Broadly speaking, you can track three types of data within your application:

- ✓ **User analytics-**Understand the geographic locations (and language settings) of your users,as well as the speed of their Internet connections, their screen sizes and resolutions, and theorientation of their displays. Use this information to prioritize your translation efforts andoptimize your layout and assets for different screen sizes and resolutions.
- ✓ **Application usage patterns-** The first step in integrating analytics is to record each Activityas you would a web site. This will help you understand the way your application is beingused, and will help you to optimize your workflows in the same way you would a web site. Taken one step further, you can record any action — which options were changed, which menu items or Action Bar actions were selected, which popup menus were displayed, if a Widget was added, and which buttons were pressed. Using this information, you can determine exactly how your application is being used, allowing you to better understand how well the assumptions you made during design match actual usage.When building games, you can use the same process to gain insight into players' progress though the game. You can track how far people progress before quitting, identify levels that are more difficult (or easier) than you expected, and then modify your game accordingly.

- ✓ **Exception tracking-** In addition to printing an error into the log output, post each unique exception thrown using your analytics. Not only does this alert you to which exceptions are being thrown, you will also gain insight into their context. Specifically, you’ll be able to see if there are particular devices, locations, or usage patterns that lead to particular exceptions.

While it’s important to track as much analytic information as possible, care must be taken when transmitting that data back to the analytics server. Every time a new data connection is created, the wireless radio may be activated and it will continue to draw power for up to 20 seconds on a typical 3G wireless radio.

As a result, it’s important to bundle the analytics information you collect and queue it for transfer the next time your application transfers data, rather than transmitting it as it’s collected.