

Play Console Help Center - Full Articles (Part 1 of 3)

Articles 1-14 of 27: Create/set up your app through Control when app changes are reviewed and published.
Source: support.google.com/googleplay/android-developer

ARTICLE 1: Create and set up your app

Source: <https://support.google.com/googleplay/android-developer/answer/9859152>

After you've created your Google Play developer account, you can create apps and set them up using Play Console.

Create your app

- 1 Open Play Console.
- 2 Select Home > Create app.
- 3 Select a default language and add the name of your app as you want it to appear on Google Play. You can change this later.
- 4 Specify whether your application is an app or a game. You can change this later. Note: If your game is free, it will appear on both Google Play and Google Play Games on PC by default. If your game is paid, you must release it on Google Play Games on PC yourself.
- 5 Specify whether your application is free or paid.
- 6 Add an email address that Play Store users can use to contact you about this application.
- 7 In the "Declarations" section: Acknowledge the "Developer Program Policies" and "US export laws" declarations; Accept the Play App Signing Terms of Service.
- 8 Select Create app.

Set up your app

After you create your app, you can start setting it up. Your app's dashboard will guide you through all the most important steps to get your app available on Google Play.

You'll start by providing details about your app's content, and entering information for your Google Play store listing. After that, you can move onto app release; this guides you through pre-release management, testing, and promotion to build pre-release excitement and awareness. The final step is launching your app on Google Play, making it available to billions of users.

Developers with personal accounts created after November 13, 2023, must meet specific testing requirements before they can make their app available on Google Play.

To start setting up your app, select Dashboard on the left menu.

Manage your app and app bundles

Google Play uses Android App Bundles to generate and deliver APKs that are optimized for each device configuration, providing users with more efficient apps. This means you only need to build, sign, and upload a single app bundle to support optimized APKs for a wide variety of device configurations. Google Play then manages and serves your app's distribution APKs for you.

Package names for app files are unique and permanent, so please name them carefully. Package names can't be deleted or re-used in the future.

Your release can have one of three statuses:

- Draft: APKs that have not been served to users yet
- Active: APKs currently being served to users
- Archived: APKs that were once active but are no longer being served to users

Find your APK files

To see your app bundles and APKs:

- 1 Open Play Console and go to the App bundle explorer page (Test and release > App bundle explorer).
- 2 The App bundle explorer page features a version filter on the top right of the page, which you can use in conjunction with the three tabs (Details, Downloads, and Delivery) to explore different versions and configurations of your app's APKs on different devices. Note: This version filter is the functional equivalent of the "Artifact library" on the old version of Play Console.

Maximum size limit

Apps on Google Play have a size limit, which is based on the maximum compressed size of your APKs at the time of download across all supported devices.

After you upload an app bundle, Play Console uses gzip to estimate what your app's download size will be. Google Play uses the best possible advanced compression tools meaning a user's actual download size will often be smaller than the estimate you see in Play Console.

The maximum size limits are:

- 200MB: Maximum compressed download size of APKs for one device generated from app bundles. The app bundle itself can be many times larger than this size.
- 100MB: Maximum compressed download size of APK for apps published with APKs (only applicable to apps created before August 2021).

Configure Play App Signing

Android requires that all apps be digitally signed with a certificate before they can be installed. When you create your first release you can configure Play App Signing to decide whether your app should use a Google-generated app signing key or an app signing key that you choose.

App version requirements for Play Console

Every app bundle and APK has a versionCode in the manifest file that's incremented with each update of your app.

To upload your app to Play Console, the greatest possible value for versionCode is 2100000000. If the versionCode of your app exceeds this value, Play Console will prevent you from submitting a new app bundle.

When selecting the versionCode for your app bundle, keep in mind that you'll need to increase the versionCode for every update and still stay below the maximum.

Note: Keep in mind the Android MAXINT differs from Play Console upload requirements.

Meet target API level requirements

When you upload an app bundle, it needs to meet Google Play's target API level requirements.

Set up your store listing and settings

Your app's store listing is displayed on Google Play and includes details that help users learn more about your app. Your store listing is shared across tracks, including testing tracks.

Product details

- 1 Open Play Console and go to the Main store listing page.
- 2 Fill in the fields under "App Details."

Field	Description	Character limit	Notes
App name	Your app's name on Google Play.	30 character limit	You can add one localized name 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	

*Character limits apply to both full-width and half-width characters — the numbers listed above are the maximum limits regardless of what type of characters you are using.

Note: Repetitive or irrelevant use of keywords in the app name, description, or promotional description can create an unpleasant user experience and result in an app being suspended on Google Play.

Preview assets

Learn more about adding preview assets (including a short description, screenshots, and videos) to showcase your app.

Note: By making your app available on Google Play, you grant Google license to use your app's assets (including your app icon, screenshots, and videos) for promotional purposes. You're enabling Google Play to showcase your app in the best possible light. Google may use these assets to highlight your app on the Play Store, in promotional campaigns, and across other Google services.

Languages and translations

Add and manage translations

When you upload an app, the default language is English (United States, en-US). You can add translations of your app's information, along with in-language screenshots and other graphic assets.

Localized images and videos

To market your app in different languages more effectively, you can add localized graphic assets to your app's Main store listing page. Users will see the localized graphic assets on Google Play if their language preferences match the languages that you've added.

Automated translations

If you don't add your own translations, users can view an automated translation of your app's Google Play store listing page using Google Translate or the default language for your app. For automated translations, there will be a note explaining that the translation has been done automatically, along with an option to view the app's default language. Automated translations aren't supported for Armenian, Raeto-romance, Tagalog, and Zulu.

Categorization and tags

You can choose a category and add tags to your apps or game in Play Console. Categories and tags help users to search for and discover the most relevant apps in the Play Store.

Contact details

When you provide an email address, website, or phone number for your app, your contact information is available to users on your app's store listing. A contact email address is required, and Google highly recommends including a website where users can contact you.

To add your support information:

- 1 Open Play Console and go to the Store settings page (Grow users > Store presence > Store settings).
- 2 Scroll down to "Contact Details."
- 3 Add your support email address (required), a phone number, and a website URL.

Next steps

- Prepare your app for review.
- Discover best practices for creating a compelling store listing.
- Publish your app using standard or managed publishing.
- Optimize your store listing with experiments.

Related content

- Learn more about publishing your app in the Academy for App Success.

ARTICLE 2: App testing requirements for new personal developer accounts

Source: <https://support.google.com/googleplay/android-developer/answer/14151465>

In November 2023, Google changed the requirements for publishing new apps on Google Play, to help developers to test their app, identify issues, get feedback, and ensure that everything is ready before they launch. These changes require developers with personal accounts created after November 13, 2023, to meet specific testing requirements before they can make their app available on Google Play.

Introducing testing requirements for newly-created personal accounts

Testing is an integral part of the app development process. By running tests against your app consistently, you can verify your app's correctness, functional behavior, and usability before you release it publicly. This minimizes the impact of any technical or user experience issues, and helps you release the best version of your app.

Developers with personal accounts created after November 13, 2023, will need to test their apps before those apps are eligible to be published for distribution on Google Play. Certain features in Play Console, such as Production (Test and release > Production) and Pre-registration (Test and release > Testing > Pre-registration), will be disabled until developers meet these requirements.

Overview of testing requirements

If you have a newly created personal developer account, you must run a closed test for your app with a minimum of 12 testers who have been opted-in for at least the last 14 days continuously. When you meet these criteria, you can apply for production access on the Dashboard in Play Console so that you can ultimately distribute your app on Google Play. When you apply, you must answer some questions to help Google understand your app, its testing process, and its production readiness.

Understanding different testing tracks and their requirements

Play Console provides different types of testing tracks so that you can gradually ramp up testing and improve your app to the point that it's ready to reach billions of users on Google Play.

- **Internal testing:** Before you've finished setting up your app, you can quickly distribute builds to a small group of your own trusted testers. This can help you to identify issues and get early feedback. Builds are normally available to testers within seconds of being added in Play Console. Internal testing is optional, but recommended as a starting point.
- **Closed testing:** With closed testing, you can share your app with a wide group of users that you control. This allows you to fix issues and ensure that your app complies with Google Play policy before you launch. You must run a closed test before you can apply to publish your app to production. At least 12 testers must be opted-in to your closed test when you apply for production access. They must have been opted-in for the last 14 days continuously.
- **Open testing:** Allows you to surface your app's test version on Google Play. If you run an open test, anyone can join your testing program and submit private feedback to you. Open testing is available when you have production access.
- **Production:** Where you make your app available to billions of users on Google Play. Before you can apply to publish your app to production, you need to run a closed test which meets the criteria.

Summary of testing requirements per track

Track types	Purpose	Requirements to access this track
Internal testing	To quickly distribute builds to a small group of your own trusted testers to identify issues and get early feedback.	None.
Closed testing	To share your app with a wide group of users that you control so you can fix issues and ensure compliance before launch.	Must have finished setting up your app.
Open testing	To surface your app's test version on Google Play — anyone can join your test and submit private feedback.	Must have gained access to production.
Production	To make your app available to billions of users on Google Play.	Must run a closed test with at least 12 opted-in testers for 14 days, then apply for production access.

Guidance and best practices for closed testing

Recruiting testers

The most common way to recruit testers is to use personal and professional networks — friends, family, colleagues, or classmates. You can reach out to communities where target users are likely to exist and actively recruit them, or post about your app on social media and ask followers to sign up for testing.

If possible, recruit a diverse group of testers representative of your app's future users to identify bugs and usability issues specific to certain types of users or devices.

Engaging with testers

Provide testers with clear instructions on how to test your app and report bugs. Let testers know what type of feedback you're looking for, and encourage them to use as many features as possible.

Include a feedback channel (email, website, or message forum). Testers can also provide private feedback through Google Play.

Important: Emphasize to testers that they need to remain opted in to your closed test for at least 14 days continuously.

Gathering and viewing user feedback

If you have an app in testing, you can access and reply to user feedback in Play Console. Feedback from users is only visible to you and can't be seen on Google Play.

- 1 Open Play Console and go to the Testing feedback page (Ratings and reviews > Testing feedback).
- 2 Decide how you want to browse your feedback: Filter (by date, language, reply state, app version, device, etc.) or Search.

Tip: Keep a record of feedback you receive — you'll need to summarize your testing feedback when you apply for production access.

Acting on user feedback

Be responsive to testers' feedback and fix bugs they find. This helps improve user experience, increases the likelihood of a successful production access application, and helps avoid negative reviews later.

Advanced testing

Explore advanced testing resources such as testing apps on Android and the fundamentals of testing on the Android Developers site. Play Console also provides a pre-launch report to proactively identify issues before your app reaches users.

Running closed tests

Learn how to set up and run a closed test.

Apply for access to production

Once you've met the closed testing requirements, you can apply for production access:

- 1 Go to the Dashboard.
- 2 Click Apply for production.

You must then answer questions divided into three sections: About your closed test, About your app/game, and Production readiness.

Part 1: Tell us about your closed test

This helps Google ensure apps have been satisfactorily tested before publishing, protecting users from low-quality apps, malware, and fraud.

- 1 Let Google know how easy you found it to recruit testers.
- 2 Provide information about tester engagement (whether testers used all features; whether usage was consistent with expected production use).
- 3 Summarize the feedback received and how you collected it.
- 4 Click Next. Important: If you click Discard or quit without finishing, your changes won't be saved.

Part 2: Tell us about your app/game

This helps Google learn more about your app; answers aren't shown on Google Play and won't affect features, display, or eligibility for developer programs.

- 1 Let Google know who is the intended audience of your app or game, as specifically as possible.
- 2 For apps: describe how your app provides value to users. For games: describe what makes your game stand out.
- 3 Let Google know how many installs you expect in the first year (rough estimate is fine).
- 4 Click Next.

Part 3: Tell us about your production readiness

- 1 Let Google know what changes you made based on what you learned during your closed test.
- 2 Describe how you decided your app or game was ready for production.
- 3 Click Apply.

After you apply for access to production

After completing your request, Google will review your submission and email the account owner with an update, usually within 7 days or less (occasionally longer).

If your app isn't ready to be published, you may be required to continue testing — for example, not having 12 testers opted-in, or testers not being engaged.

If successful, you can access Production and make your app available to billions of users on Google Play, and use Open testing. Google recommends testing extensively before publishing, and routinely testing future updates.

Frequently asked questions

What do you mean when you say testers must be opted-in for the last 14 days continuously before I can apply for production?

This means Google won't count testers who opted in, tested for less than 14 days, and then opted out. Even if they opt back in for a total of 14 days, these days must be consecutive to count toward the criteria of 12 opted-in testers who tested for 14 consecutive days.

Are there any more best practices for testing that I might find helpful?

Continue closed testing while fixing user-reported issues and bugs before going to production, to minimize low quality reviews and ratings. Consider inviting closed testers to a messaging group so feedback can be seen by others. Test overall user experience, not just crashes and bugs.

Are there any other Play Console features I should learn about to help me succeed on Google Play?

Learn more about Play Console's features on the Google Play site, and take free online training on Google Play Academy.

ARTICLE 3: Device verification requirements for new developer accounts

Source: <https://support.google.com/googleplay/android-developer/answer/14316361>

Starting in early 2024, developers with new personal accounts will be required to verify that they have access to a real Android mobile device using the Play Console mobile app before they can make their app available on Google Play.

This is an important step in Google's ongoing efforts to improve user trust and safety for the billions of people that use Google Play. By requiring developers to verify that they have access to an Android device, Google can help ensure that they are able to properly test their apps before making them available to users, improving app quality and keeping users safe.

Complete device verifications

- 1 Log into your Play Console account (web version) as the account owner (the Google account used to create the developer account).
- 2 Go to the Home page.
- 3 Find the "Verify that you have access to an Android mobile device" task and click View details.
- 4 Scan the QR code to launch or install the Play Console mobile app on the device you wish to use.
- 5 Open the Play Console mobile app and log in as the account owner.
- 6 Select your developer account.
- 7 Tap Verify and follow the on-screen instructions.
- 8 After completing device verification, the mobile device verification task will no longer be shown on the Home page in Play Console.

Frequently asked questions

Why can't I see my developer account in the Play Console mobile app?

This may be because you didn't sign in using the correct account. Make sure to sign in using the same Google Account that you used to create your Google Play developer account.

What sort of device can I use to complete device verifications?

You can use any non-rooted physical Android mobile device that runs at least the Android 10 operating system.

I have multiple devices. Does it matter which one I use for verification?

You can use any non-rooted physical Android mobile device that runs at least the Android 10 operating system.

What is my device information used for?

Google wants to ensure that people who develop apps for Google Play users have the basic tools they need to succeed. Some information may also be used to help with abuse prevention efforts.

Can I use a device that I have used to verify a different developer account?

Yes, you can use the same device to verify multiple accounts.

How long does verification take?

Device verifications using the Play Console mobile app should take less than a minute.

Does it have to be the device with the same mobile number I provided earlier in the sign-up process?

No, the phone number of the device used to verify is not used or collected as part of device verifications.

Will I be required to keep this device so it can be verified again in the future?

No. Google may ask you to verify in the future, but you will not have to use the same device.

ARTICLE 4: Set up your app on the app dashboard

Source: <https://support.google.com/googleplay/android-developer/answer/9859454>

After you've created your app, you can start setting it up. Your app's dashboard will guide you through all the most important steps to get your app available on Google Play.

Using your dashboard

To start setting up your app, select Dashboard on the left menu. Under your app's details at the top of the page, you'll find different categories and tasks relating to app setup and release. You must complete the mandatory tasks before you can launch your app on Google Play.

When you complete a task, you'll see a green tick mark and strikethrough text. The progress bar at the top of the section will also be updated. As you complete more tasks, your dashboard will populate with performance data and key insights by set time periods.

After launching your app, you can visit your dashboard to get an overview of key metrics, trends, and insights for your app.

Initial setup

App content

Start by providing some information about your app and setting up your Google Play store listing. Providing this information helps ensure that your app is safe for its intended users, is compliant with Google Play policies, and satisfies legal requirements.

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.
- Support your app's users by providing contact information.
- Create a compelling store listing by adding screenshots and a video that showcase your app.

Release your app

Test your app

Test your app with specific groups of testers initially, before opening testing up to a bigger set of Google Play users. Users can provide constructive feedback that helps you make improvements without affecting your public rating.

Important: Developers with personal accounts created after November 13, 2023, must meet specific testing requirements before they can access production to release their app on Google Play, or use other features such as pre-registration or open testing.

Testing tasks include:

- Setting up an internal or closed test.
- Preparing and rolling out a release to the test track.
- Inviting testers via email or Google Groups, and gathering private feedback.
- Setting up an open testing track, and moving on to open testing so anyone can join your tests on Google Play and provide feedback without affecting your Google Play rating.

Pre-registration

With pre-registration, you can publish your store listing before you launch so users can register their interest. When you launch, pre-registered users receive a notification to download your app, or the app is auto-installed.

Release your app (to production)

To launch your app on Google Play, you need to roll out a release to its production track.

Post-launch

After launching your app, you can visit your dashboard to get an overview of key metrics, trends, and insights for your app. Metrics displayed may change over time as more usage data is available. You can customize your dashboard to display the data most important to you.

Related content

- [Create and set up your app](#)
- [Update or unpublish your app](#)
- [Publish your app](#)
- [Control when app changes are published with managed publishing](#)

ARTICLE 5: Inspect app versions on the Latest releases and bundles page

Source: <https://support.google.com/googleplay/android-developer/answer/9844279>

Starting August 2021, new apps are required to publish with the Android App Bundle on Google Play. New apps larger than 200MB can use either Play Asset Delivery or Play Feature Delivery.

As of June 30, 2023, Google Play no longer supports TV app updates using APKs. All TV app updates must be published with Android App Bundles (AAB).

The Android App Bundle is the publishing format on Google Play. Publishing using app bundles helps reduce the size of your app, simplify releases, and enable advanced distribution features.

Using the Latest releases and bundles page in Play Console, you can easily manage your app bundles and versions in one place, and access useful metadata, downloads, and insight into what Google Play generates for asset delivery.

How app bundles work and their benefits

Google Play uses app bundles to build and deliver APKs optimized for each device configuration. You only need to build, sign, and upload a single app bundle to support optimized APKs for a wide variety of device configurations. Google Play manages and serves your app's distribution APKs.

App bundle features and benefits

- Publish smaller apps, facilitating faster installs and smaller size-on-disk, which can lead to fewer uninstalls.
- Simplify release management, removing the complexity of publishing and managing multiple APKs.
- Use Play Feature Delivery to modularize your app and include feature modules, delivered in different ways:
- Install-time delivery: Feature modules delivered at install time.
- Conditional delivery: Feature modules delivered at install time based on conditions such as user country, device features, and minimum SDK version.
- On demand delivery: Install and uninstall feature modules as needed.
- Instant experiences: Feature modules can be instant-enabled to offer instant experiences from links and the Try Now button on your store listing without requiring install first.
- Use Play Asset Delivery to deliver large asset packs, customizable in different ways:
- Install-time delivery: Asset packs delivered with the install and available at launch.
- Fast follow delivery: Asset packs delivered automatically after install without blocking app open.
- On demand delivery: Asset packs downloaded as needed when the app is running.
- If your app has its own in-app language picker, you can use the additional languages API to let users access and download additional language resources on-demand.

Using the Latest releases and app bundles page

The Latest app bundles page features a version filter on the top right, used with three tabs to explore different versions and configurations of your app's distribution APKs on different devices (equivalent to the old "Artifact library").

- Details: Review granular details for each version of your app.

- Downloads: Generate install links for each version, download device-specific APKs for testing or pre-installing, download a universal APK, manage other assets.
- Delivery: Review what Google Play generates from your app bundle and under what conditions distribution artifacts are delivered.

Prerequisites and recommendations

- To use app bundles, you must enroll in Play App Signing.
- To use advanced distribution features, review documentation on Play Feature Delivery and Play Asset Delivery.

Review your APK details

- 1 Open the Latest releases and bundles page (Test and release > Latest releases and bundles).
- 2 On the Devices tab, select the version filter near the top right of the page.
- 3 On the "Choose a version" table, select the right arrow on the version you want to view.
- 4 Review the version details. Optional: Under "Supported Android devices," select View device catalog to view and manage compatible devices.

Share an install link

- 1 Open the Latest releases and bundles page.
- 2 Select the version filter.
- 3 Select the right arrow on the version you want to view.
- 4 Select the Downloads tab.
- 5 In the "Internal app sharing link" section, select Copy shareable link.
- 6 Share the link. Tip: Select Manage access to visit the Internal app sharing page and quickly share links with your team.

Download device-specific APKs

You can download device-specific APKs and pre-install APKs to share with OEMs to pre-install on their devices.

- 1 Open the All app versions page from Latest releases and app bundles (Test and release > Latest releases and app bundles > All app versions).
- 2 Select the version filter.
- 3 Select the right arrow on the version you want to view.
- 4 Select the Downloads tab.
- 5 In the "Device-specific APKs" table, select the download icon next to the APK you want.

Download a signed, universal APK

A signed, universal APK is a single, installable APK signed with the same app signing key used by Play App Signing. You can distribute it on other app stores and channels, so wherever you distribute your app, it's signed with the same key.

- 1 Open the All app versions page.
- 2 Select the version filter.

- 3 Select the right arrow on the version.
- 4 Select the Downloads tab.
- 5 In the "Assets" table, select the download icon next to "Signed, universal APK."

Download asset and deobfuscation files

- 1 Open the All app versions page (Test and release > Latest releases and app bundles > All app versions > App bundle explorer).
- 2 Select the version filter.
- 3 Select the right arrow on the version.
- 4 Select the Downloads tab.
- 5 In the "Assets" table, select the download icon next to the files you want.

View delivery information for feature modules and asset packs

- 1 Open the All app versions page.
- 2 Select the version filter.
- 3 Select the right arrow on the version.
- 4 Select the Delivery tab.
- 5 View delivery information: Modules table (always available, includes base module) and Asset packs table (only if asset packs are associated with your app), listing name, delivery conditions, and download size.
- 6 Click a table row to view everything generated from your app bundle, including detailed delivery conditions, generated split APKs, and generated standalone APKs. Note: Standalone APKs always include the base module and any install-time feature modules or asset packs, plus any on-demand modules with fusing enabled.

Related content

- Learn how to roll out your app to a specific track.
- Become an Android App Bundle expert.

ARTICLE 6: Prepare and roll out a release

Source: <https://support.google.com/googleplay/android-developer/answer/9859348>

With a release, you can manage your app's Android App Bundle (or APK for apps created before August 2021) and then roll out your app to a specific track.

Step 1: Create a release

A release is a combination of one or more app versions that you'll prepare to launch an app or roll out an app update. You can create a release on three different testing tracks or to production:

- Open testing: Available to testers on Google Play. Users can join tests from your store listing.
- Closed testing: Available to a limited number of testers you choose, who test a pre-release version and submit feedback.
- Internal testing: Available to up to 100 testers you choose.
- Production: Available to all Google Play users in your chosen countries.

Important:

- You must have the "Release apps to testing tracks" permission to create a new release.
- Developers with personal accounts created after November 13, 2023, must meet specific testing requirements before making their app available on Google Play.
- You cannot create a new release when you have outstanding releases. Roll out any staged releases to 100%, or remove changes on the Publishing overview page and discard any unpublished releases first.

To start your release:

- 1 Open Play Console and go to the track you want to start your release on: Open testing, Closed testing (to create a release on an existing closed testing track, select Manage track; to create a new track, click Create track), Internal testing, or Production.
- 2 Near the top right of the page, click Create new release. Note: If Create new release is disabled, you may have outstanding setup tasks listed on the Dashboard page.

If you want to edit an existing release, go to the appropriate release page and click Edit release.

Step 2: Prepare your app's release

- 1 Follow the on-screen instructions:
 - If this is your first release, follow instructions to configure Play App Signing.
 - Add your app bundles (apps created before August 2021 can add app bundles or APKs for app updates).
 - Optional: If this is the first release, you can change your app signing key via "App integrity" section > Change app signing key. Before changing: internal/closed track users who already installed will no longer receive updates and need to reinstall; you cannot use previously uploaded app versions and must re-upload.
 - Optional: Add Play Games Sidekick (beta) to your App bundle.
 - Name your release and enter release notes.
- 1 Select Save as draft to save changes.
- 2 Select Next when finished preparing.

Note: Once you publish an app to an open track, its key is fixed.

Prepare (section details)

App integrity: Shows the status of Play App Signing.

App bundles: Upload new app bundles or add from library. Click the three dots icon to: upload a ReTrace mapping file (.txt), upload native debug symbols (.zip), upload an expansion file (.obb), or remove an app bundle. Apps created before August 2021 can use app bundles or APKs.

Included: View details about app bundles from your previous release that will be included in this release. Clicking Remove removes the app bundle from this release (findable again in App bundle explorer).

Not included: View details about app bundles from your previous release that will not be included. Clicking Include adds the app bundle to this release.

Declare permissions for your app (optional): Permission requests are evaluated during the release process after adding app bundles or APKs. High-risk or sensitive permissions (e.g., SMS or Call Log) may require completing the Permissions Declaration Form and receiving approval.

Release name: Only for use in Play Console, not visible to users. Auto-populated with the version name of the first app bundle/APK added. Add a meaningful name like a build version ("3.2.5-RC2") or internal code name ("Banana").

What's new in this release?

- Overview: Let users know about recent updates in this release. Release notes shouldn't be used for promotional purposes or to solicit actions.
- Add release notes and manage translations: Add description between relevant language tags (shown per supported language). To change supported languages, add translations first. Format example:

``

`The release notes description can take up multiple lines.`

``

Note: Up to 500 Unicode characters per language.

- Copy from previous release: Select Copy from a previous release to copy notes and translations into the text box for further editing (replaces existing entered notes).

Step 3: Review and roll out your release

Prerequisite: Before rolling out, make sure you've set up your app's store listing, prepared your app for review on the App content page, and set up your app's prices.

Important: When preparing your release, you may have the option to Save or Publish depending on whether changes need review. Publishing pushes changes live immediately. Saving adds changes to "Changes ready to send for review" on the Publishing overview page, where you decide when to send for review.

Once ready to roll out:

- 1 Open Play Console and go to the appropriate track (Open testing or Closed testing).
- 2 Select the Releases tab, then Edit under the release you want to roll out.

- 3 Review your draft release, make necessary changes, and select Next. You'll see the "Preview and confirm" screen to check for issues.
- 4 If you see "Errors summary" at the top, click Show more to review details and resolve problems. Note: You can view recommended/required resolutions. You can't publish until errors are resolved; warnings/minor issues can still be published but addressing them first is recommended.
- 5 If updating an existing app, select a rollout percentage (first release won't show this option). For staged rollout targeting specific countries, see the staged rollouts article.
- 6 Select Start rollout. If rolling out your app's first production release, clicking Start rollout to production also publishes your app to all Google Play users in the countries you selected.

Step 4: Review release details

Once created, you'll see this information for the latest app release rolled out to each track under "Latest releases" on your Releases overview page:

- Release: Name to identify release in Play Console only.
- Track: The track the release has been rolled out to.
- Release status: Current status.
- Last updated: Date/timestamp of last rollout event.
- Countries/regions: Number of countries/regions the last rollout is available to.

You can view more in-depth info via the release's Release details page, including:

- Release overview: Metrics on installs, updates, performance issues, and rating vs. previous releases.
- App bundles and APKs: List of new, retained, and deactivated app bundles/APKs.
- Release notes: List of previous release notes.
- Rollout history: Timeline of when the release was halted, resumed, or served to a new percentage of users.

Optional: Discard a release

Process varies by release state:

- Draft: Click Discard draft release near the top-right. You lose changes made in this release.
- Ready to send for review: Click Discard release on the release summary; removed from Publishing overview and not included in changes to send for review.
- In review or Ready to publish: First remove changes that are in review/ready to publish on Publishing overview, then click Discard release.
- Rejected: Find your rejected release and click Discard release.

Note: You can only discard the latest release on a track, including draft and recently rejected releases.

Track and manage releases on your Releases overview page

The Releases overview page (under "Release" in the left menu) helps you monitor availability of your apps across tracks, view countries/regions, and select individual releases for details.

Track and manage app changes on the Publishing overview page

Use the Publishing overview page to control when certain changes to your app are sent for review and published.

Related content

- [Distribute app releases to specific countries.](#)
- [Set up an open, closed, or internal test.](#)
- [Learn about releasing updates progressively with staged rollouts in the Academy for App Success.](#)

ARTICLE 7: About your submission activity

Source: <https://support.google.com/googleplay/android-developer/answer/17118609>

Every time you submit a group of changes to Google for review, a unique submission is created. Your submission activity provides a log of what specific modifications were included in each submission, when they were received, how it was submitted, and its current state.

Starting May 1, 2026, Google will display your submission records, maintaining up to one year of history from that point forward.

Using your submission activity

This page is a destination where you can view and monitor publishing activity. Use it to get a better understanding of the overall review process and see exactly when your submission has changed states. It also provides a comprehensive history of past submissions, and a submission ID that you can reference when seeking support.

Track your submission status

Status	Description
In review	Google is actively evaluating the changes in your submission. This status remains active until all necessary reviews are complete.
Ready to publish	Your changes have passed all reviews. You must manually publish when you're ready.
Published	Your submission is now live on Google Play.
Rejected	Your changes did not comply with Google Play policies and were not published. If an update is rejected, your last approved version remains live.
Cancelled	Your submission was withdrawn. This happens if you remove the changes manually, or if a newer submission supersedes an older one that was still in review, or if the app is blocked or suspended.

Review submission details

In the submission activity page, select the submission you want to review. This opens a summary of the submission details including:

- Submission ID: A unique identifier used for tracking and support resolutions.
- Source: Indicates whether the submission originated from the Play Console interface or the Publishing API.
- Status update timestamp: Records history across the publishing lifecycle, marking when the submission was sent for review, approved, canceled, or published.

Policy transparency

Latest rejected submissions include a link to the Policy status page, where you can see details of the issue and get instructions on how to fix them.

ARTICLE 8: Halting a fully rolled-out release

Source: <https://support.google.com/googleplay/android-developer/answer/16285429>

You can halt an app release that is rolled out to 100% of your users. If you discover a critical issue with your app, the halt rollout feature minimizes disruption for users by preventing new and existing users from installing, or updating to the affected version of your app. When you halt a full release, the previous version of your app will become available to new users, and anyone not using the halted version of your app.

To halt a fully rolled-out release

- 1 Go to the Releases tab on your track page.
- 2 Select Manage rollout > Halt rollout.

Key capabilities

- You can stop a fully rolled-out release on any track, excluding internal test tracks, directly from the Play Console web or mobile UI, or via the Publishing API. The halted version will no longer be served to users.
- When you halt a release, a previously live, fully rolled-out version of your app will automatically take its place and become available to new and eligible users.
- You can resume a halted release to any percentage of users, allowing you to gradually reintroduce a release after fixing the issue.

Considerations

- If the release you want to halt has been available for a significant length of time, or is being used by a large percentage of your users, halting it might not be the most effective solution as most users may have already updated to the halted version.
- You cannot halt your first release on a track. If your current 100% release is the very first version of your app on a given track, there will be no previous release available to revert to.
- If your previous release has a policy violation, it cannot be used in place of your halted release.

By using the Halt a fully-rolled out release feature, you can gain more control over your app releases and quickly mitigate issues that arise after a full rollout, ensuring a better experience for your users.

ARTICLE 9: Detect app issues early with pre-review checks

Source: <https://support.google.com/googleplay/android-developer/answer/1480773>

Pre-review checks help you catch potential problems with changes you plan to make to your app, before you send them for review. Some of the problems may be critical and must be fixed before you can send your changes for review.

Overview

You can find pre-review checks on the Publishing overview page, in the "Changes not yet sent for review" section. This section is only visible if you have made changes to your app since it was last sent for review.

Pre-review checks automatically start running whenever you make changes to your app. You can view their status on the Publishing overview page:

- Some checks take longer to run than others, so you may see an estimated time until they finish.
- You don't need to wait for the checks to finish to send your changes for review. You can send changes at any time, and Google will finish these checks before sending your changes to be reviewed.

Issues will be visible as soon as they are discovered. Click View issues to learn about each issue.

Managing detected issues

Google checks for a variety of different issues, providing information on each along with links to help fix them or learn more. You can fix many issues directly in Play Console, while others might require technical changes to your app (a new app version), or changes outside Play Console (e.g., to your developer website).

Some issues must be fixed before you can proceed. Others can be deferred by selecting Proceed anyway. To proceed with known issues, you may be asked to disclose why by selecting one or more reasons, such as "I already have plans to resolve this issue" or "This issue is incorrect." This is logged in the Activity log and used to help improve the reliability of checks. Other times, you may be asked to confirm you've fixed something outside of Play Console before proceeding.

Your response won't directly affect your app review outcome, but if the issue makes your app non-functional, it might be rejected during review for failing to comply with Google Play's app quality policies (such as the Minimum Functionality policy). Other issues still present during review are also likely to result in rejection.

If you proceed anyway without fixing detected issues, they may reappear in the future.

Required permissions to proceed with issues

Only users with one of the following permissions can proceed with known issues:

- Release to production, exclude devices and use Play app signing
- Release apps to testing tracks
- Manage testing tracks and edit tester lists
- Manage store presence
- Admin (all permissions)

These are the same permissions used for other features on the Publishing overview page, including managed publishing.

Frequently asked questions

Why are you introducing these new checks?

To provide early feedback on critical issues with planned changes, ensuring apps work great for users, and to help avoid common mistakes that can result in changes being rejected during review.

Are these checks new?

Google has always run checks to ensure apps comply with the Developer Program Policy and perform well on user devices. Google is now surfacing a number of these upfront in Play Console, along with newer quality checks such as app compatibility issues on newer Android versions.

Are pre-review checks the same as the existing app review process?

No. These checks run before your app is in the review process.

What pre-review checks are run?

A variety of checks across your app, including ensuring updates contain all mandatory declarations within the App content page, and detecting high crash rates on certain devices or Android versions.

How long do the checks take?

All checks take no longer than 15 minutes to return but are usually much shorter. You can select Send for review while checks are still running if no issues have been found yet — Google finishes running checks and automatically sends your changes for review if no issues are found.

How do these checks work over the API?

Currently, most checks run only on the Play Console website, with a subset running over the API. Checks that return almost immediately run over the API; longer-running checks do not.

Can I simply click Proceed anyway for all issues?

No, some checks are required for publishing before you can proceed, such as missing app content declarations. Some issues are also checked during the review process, so if the issue persists, your app is likely to be rejected.

What do you do with the responses selected after I proceed with known issues?

These responses help Google understand if a check was right or wrong, improving check reliability. Your response won't directly affect your review outcome, but non-functional issues might still be rejected for policy violations (like the Minimum Functionality policy).

If I pass all pre-review checks, can I still fail the app review process?

Yes — it's possible to pass all pre-review checks and still fail the more comprehensive app review process.

What happens if I send changes for review before the checks are finished?

The checks continue running, and your changes are sent for review automatically once complete, as long as no critical issues are detected.

Are all detected issues accurate?

Google strives for accuracy, but some issues might be incorrect due to their technical nature. You can proceed in these instances and select an appropriate reason as feedback.

Related content

- Learn more about launching your app or game at Google Play Academy.
- Discover how to control when app changes are published with managed publishing.
- Get more information on how to update or unpublish an app.

ARTICLE 10: Publish your app

Source: <https://support.google.com/googleplay/android-developer/answer/9859751>

Whether you're publishing an app for the first time or making an update, your app's publishing status helps you understand its availability on Google Play. You can see your app's latest publishing status under your app's name and package name when you select your app in Play Console.

Note: For certain developer accounts, Google will take more time to thoroughly review your app to help better protect users. This may result in review times of up to seven days or longer in exceptional cases.

Publishing status

There are three types of publishing status:

- App status: Helps you understand your app's availability on Google Play and who it's available to (testers, all Google Play users, etc).
- Update status: Helps you understand the availability of your latest update.
- Item status: Helps you understand the availability of a specific part of an update, such as a particular release, a content rating, or a store listing experiment.

App status

Status	Description
Draft	Your app has no presence on Google Play because either you haven't published it yet or it was rejected during review.
Internal testing	Your app is available to internal testers via URL only — not discoverable on Google Play.
Closed testing	Your app is discoverable on Google Play but only selected testers can install and use it.
Open testing	Your app is discoverable on Google Play and any user can become a tester. You can set a limit on the number of testers.
Pre-registration	Your app is available for users to pre-register on Google Play. When you launch, pre-registered customers receive a notification to download the app.
Production	Your app is available for download to Google Play users in your chosen country or region.
No active releases	Either you did not roll out updates on any tracks, or the updates were rejected.
Unpublished	You've chosen to unpublish your app from Google Play. It's discoverable for existing users only; updates are only available for existing users.
Removed by Google	Google has removed your app indefinitely — not discoverable, requiring a policy-compliant update to reinstate.
Suspended by Google	Google has removed your app indefinitely — not discoverable, requiring a successful appeal to reinstate.

Update status

Type	Description
No pending changes	Your app has no changes available to submit.
In review	Google is reviewing your update.
Update rejected	One or more changes are not compliant with Google Play policy or the Developer Distribution Agreement. Fix and resubmit, or submit an appeal.
App rejected	Same as Update rejected but applies only to apps in Draft attempting to publish for the first time.
Changes not yet sent for review	You've made changes that haven't yet been reviewed. Every change requiring review is added to the "Changes not yet sent for review" section on the Publishing overview page; changes are not automatically sent for review.
Previous	A valid update made in the past, kept as a record — helpful for managed publishing.

Item status

Status	Description
Draft	This item has never been submitted for review.
In review	This item is part of an update being reviewed.
Update rejected / App rejected (Draft to Published)	Part of a non-compliant update. Fix and resubmit, or appeal.
App rejected	Part of a non-compliant update; applies only to apps in Draft attempting to publish.
Available	The most recent version of this item is available on your app and Google Play.
Previous	A valid item submitted in the past but superseded or made obsolete by a later submission.

Publish a draft app

When ready to publish a draft app, roll out a release. At the end of the release process, clicking Release will also publish your app.

Problems publishing a draft app?

If you see "Errors summary" at the top of your release's review summary page, click Show more for details. You can't publish until errors are resolved. Warnings/minor issues can still allow publishing, but addressing them first is recommended.

Publish an app update

Use standard publishing or managed publishing:

- **Standard publishing:** Updates are processed and published as soon as possible (default). Certain apps may be subject to extended reviews, up to 7 days or longer in exceptional cases.

- Managed publishing: Updates processed as usual; after approval, you control exactly when changes are published.

Important: To publish updates, decide which permissions you need with your account owner:

- Release to production, exclude devices, and use Play App Signing
- Release apps to testing tracks
- Manage testing tracks and edit tester lists
- Publish Google Play games services projects
- Create and publish private apps to your organization

Related content

- Learn more about launching your app or game in the Academy for App Success.
- Discover how to control when app changes are published with managed publishing.
- Learn how to make your app unavailable to new users in Update or unpublish an app.

ARTICLE 11: Helpful tips to get your app published on Google Play

Source: <https://support.google.com/googleplay/android-developer/answer/15191715>

Google is here to help make your app publishing experience on Google Play as smooth as possible. Whether you're a seasoned developer or just starting out, these tips should help you avoid the most common pitfalls that developers often encounter when publishing an app for the first time.

Be clear and comprehensive with your store listing information

Include a clear and comprehensive description in your store listing. This makes your app easier for users to find, while helping Google quickly understand your app during review. Even if your app functionality seems obvious, write more than a single word or sentence — be descriptive and clearly explain what your app does, why it's valuable, and what users can expect. Misleading, non-descriptive, irrelevant, excessive, or inappropriate store listing information may be rejected per the Metadata policy.

Make sure your app delivers what it promises

Your store listing information, including screenshots and images, should accurately reflect your app's functionality. If reviewers notice a discrepancy between what's promised and what's delivered (e.g., a described feature is absent), the app may be rejected. Don't include misleading information or images.

Provide login information to help us review your app

If your app or parts of it are restricted based on login credentials or other sign-in details, provide an active demo account, valid login information, and any other resources needed for review, per Play Console Requirements. Otherwise, the app can't be reviewed and may be rejected.

Ensure that your app works properly and try test tracks

Apps must install and load properly. Policy requires a stable, engaging, responsive user experience — apps that load correctly, respond quickly, and don't crash. New developers are required to thoroughly test their apps with a variety of users before submission. Per the Broken Functionality policy, apps that don't function as expected will be rejected. Use test tracks — pre-release channels for distributing to selected or wider tester groups before public launch — to catch bugs and refine your app. Catching issues early helps prevent multiple rejections, which may lead to suspension.

If your app gets rejected, fix the issue and resubmit

If rejected, fix the issue and resubmit an updated version. Complete comprehensive testing to confirm the issue is resolved before resubmitting. Review the complete Developer Program Policies in the Developer Policy Center, take free e-learning policy courses via Google Play Academy, or visit the Google Play Developer Help Community forum.

Quick checklist for a smooth app publishing experience

- Describe your app clearly: Detail its functionality and benefits in your store listing.
- Deliver what you promise: Your app must deliver what the description says it will.
- Provide login details: If required, provide details to the review team.
- Test thoroughly: Submit a polished, bug-free app. Use test tracks to catch issues.
- Resubmit or reach out: If rejected, revise and resubmit.

ARTICLE 12: Best Practices for White Label Developers

Source: <https://support.google.com/googleplay/android-developer/answer/15884185>

If you're a white-label app developer creating template-based apps for third-party clients, this best practices guide helps streamline your publishing process on Google Play. It focuses on four key areas:

- Account management: Understanding account structure and trade-offs of each approach.
- Quality store presence: Crafting unique and compelling store listings that comply with policies and ensure clarity for users.
- Policy compliance: Understanding and complying with policies to prevent account suspensions or bans.
- App review and publishing: Streamlining the review process and managing releases effectively.

Account Management

White-label app developers can typically use two approaches, though Google highly recommends decentralized account management.

Decentralized account management (recommended): Each client (or group of clients) has a separate developer account. Clients generally manage their own app updates and content. Two management styles:

- Unmanaged accounts: Clients retain full control and responsibility for managing their own accounts independently.
- Managed accounts: The white-label developer retains administrative access and oversees the client's accounts — typical when individualized developer accounts are needed but the client still wants technical maintenance support (e.g., financial institutions with regulatory requirements).

Benefits: Isolates risks from the white-label company and allows fully independent client branding under a unique developer name.

Centralized account management: All apps are managed under a single developer account by the white-label developer.

While this may appear to simplify management, Google strongly advises using the decentralized model instead, since policy issues with one app can negatively impact all other apps in the same account (including account suspension or termination). Also, developer account information shown on Google Play will be the same for all apps under this model — if clients want to be fully branded away from the white-label developer, decentralized is preferred.

Summary: Google recommends the decentralized model for white-label app developers because it:

- Enables clients to manage their own apps and publish apps under their own unique developer name.
- Allows isolation of the impact of potential policy violations.
- Empowers you to provide comprehensive guidance and support to clients.

Google strongly recommends very limited use of the centralized model, because centralized management of multiple apps for different organizations carries inherent risks, and policy violations on a single account can lead to removal of all associated apps, impacting partners, customers, or clients.

Repeated or severe policy violations can result in an account ban, removing all associated apps and preventing future submissions. Complying with Google Play policies is critical for long-term success in white-label app publishing regardless of account management approach.

Building a quality store presence

White-label app developers often repurpose metadata across multiple apps for efficiency. To ensure a positive user experience and comply with policies, make each store listing unique and avoid repetitive or misleading content:

- **Ensure store listings are unique:** Each app — even a branded template app — should have its own compelling store listing with a unique description, icons, graphics, and relevant screenshots. This prevents a "sea of identical looking apps." Per the Repetitive Content policy, apps should provide value through unique content or services. Don't use graphic assets identical or similar enough to existing products/services to mislead users. For locale-specific apps, include the locale in the app icon (e.g., "App1-New York", "App2-Los Angeles"). Common violations include: using the exact same descriptions for multiple apps; using the same store listing screenshots for multiple apps; repurposing the same featured graphics for multiple apps.
- **Describe your app accurately:** Ensure descriptions accurately describe functionality and content, per the Deceptive Behavior policy. Only highlight features actually available in your app — avoid mentioning unreleased features (update the description after launching new features). A common violation: using screenshots showing functionalities/features not available in the app.
- **Make sure screenshots match the app:** Screenshots must accurately reflect the app's functionality. If a screenshot shows something that can't be found or reproduced during review, it may be rejected for being misleading or violating the Metadata or Broken Functionality policy.
- **Show clear value proposition:** Explain clearly what your app does and why it's different from other versions (e.g., regional variations).
- **Use concise descriptions:** Keep descriptions concise, well-formatted, and avoid unnecessary details.
- **Avoid repetition:** Don't repeat the short description in the full description or excessively repeat keywords (e.g., avoid: "Car racing, car driving, race cars, car races, race track, driving, drive, race, cars, vehicles, automobiles, trucks.>").
- **Be transparent:** Clearly state if an app is designed for a specific user base.

Additional recommendations

- Use templates for pre-publish questions: Create your own documents/templates to answer declarations accurately and consistently in Play Console.
- Review additional best practices: See other helpful tips and best practices for your store listing.

Policy compliance tips

Policy violations can have serious consequences, including account bans:

- **Submit fully functional apps:** Ensure your app is fully functional before submitting for review — non-working areas may result in rejection. Use test tracks for unfinished apps.
- **Provide login credentials:** Include an active demo account, login credentials, sign-in details, and any resources needed for review, per Play Console Requirements. Otherwise the app may be rejected.
- For unmanaged accounts using the decentralized model, implement basic checks before publishing to ensure basic quality standards (e.g., verifying the app description is more than just a copy of the app title).

App reviews and publishing

- **Avoid changes during review:** Refrain from making changes to your app while under review — consider a "code freeze" during this period.

- Check your policy status: Open Play Console > select an app > Policy status on the left menu. "No issues found" means no enforcement actions have been taken. If rejected, the last successfully published version remains available. If removed, it won't be available until a compliant update is submitted. If suspended, it's no longer available and you can select Appeal.
- Plan for review time: Build extra time into your release schedule; reviews typically complete within 7 days but may take longer in exceptional cases.
- Consider submitting updates early using Managed Publishing: Updates are processed as usual, but you control exactly when changes are published.
- Document policy resolutions: If publishing frequently, document common policy issues, resolutions, and process changes to prevent similar issues in future releases.

By following these tips, white-label app developers can navigate the complexities of app publishing, maintain positive account standing, and ensure smooth, successful app launches for clients.

ARTICLE 13: Update or unpublish your app

Source: <https://support.google.com/googleplay/android-developer/answer/9859350>

When you're ready to make changes to your app, this article describes what you need to do to make sure that existing users receive your update.

Prepare your updates

- The package name of the updated Android App Bundle must be the same as the current version.
- The version code must be greater than the current version.
- The updated app bundle must be signed with the same signature as the current version.

Confirm an app bundle's certification

To confirm your app bundle uses the same certification as the previous version, run the following command on both versions and compare results:

```
`$ jarsigner -verify -verbose -certs my_application.aab`
```

If results are identical, you're using the same upload key and are ready to continue. If different, you'll need to re-sign the app bundle with the correct upload key.

Tip: Use the App bundle explorer to easily manage your app bundles in one place.

Upload your app bundle

Once your updated artifact is ready, you can create a new release.

Delivery of updates

After submitting an update, you'll see "In review" under "Update status" on your app's Dashboard. Once published, the update starts being distributed to existing users.

Once available, users can download the update from your app's store listing page or from their "My apps" page on the Google Play app. If a user has automatic updates turned on, the update downloads and installs automatically. App updates can take some time to be delivered to existing users.

Unpublish an app

When you unpublish an app, existing users can still use your app and receive app updates, but new users won't find and download it on Google Play.

Before you can unpublish an app, ensure that:

- You have accepted the latest Developer Distribution Agreement.
- Your app has no errors that need to be addressed, such as failing to fill in the content rating questionnaire or provide details about your app's target audience and content.
- Managed publishing is not active for the app you want to unpublish.

To unpublish your app:

- 1 Open Play Console.
- 2 Select an app.

- 3 Go to Test and release > Setup > Advanced settings.
- 4 On the App Availability tab, select Unpublish.

Updating system apps

Users will see system apps (including preloaded apps) in the "My Apps" section of Google Play as soon as an app with the same package name is uploaded to Play Console (even if the app is unpublished).

Google Play can manage updates to preloaded apps, provided:

- The preloaded app is in the system partition.
- The preloaded app is free.
- The preloaded app is signed with the same signature as the app published on Google Play.
- The package name of the preloaded and updated app is the same.
- The version code of the updated app is greater than that of the preloaded app.

Tip: Use the Play Developer API to download system APKs generated from app bundles you upload to Google Play.

If you need to upload a system app and encounter an error, contact support.

Related content

- Learn more about updating your app in the Academy for App Success.

ARTICLE 14: Control when app changes are reviewed and published

Source: <https://support.google.com/googleplay/android-developer/answer/9859654>

In February 2023, Google made changes to the publishing workflow to make it easier to understand which changes you're sending for review, and to better control when you send certain changes for review.

You can use the Publishing overview page to control when certain changes to your app are sent for review and published. You can see changes made to your app, and send certain changes for review when ready. You can also turn managed publishing on and off — when on, you can see which submitted changes are in review and which are ready for publishing on the Publishing overview page.

See an overview of your changes

The Publishing overview page provides an overview of changes made to your app, and helps you control when certain changes are sent for review or published. Changes listed may include:

- Country targeting changes affecting production and testing tracks.
- Changes to App content that help review your app, such as the content rating questionnaire or target audience changes.
- Any time you sync/unsync a track's country targeting with production.
- Synced tracks affected by changes to your production track's country targeting.
- Updates to your store listings.
- Changes to your store settings.

Each change is listed in a table, with a high-level description and a link to the change itself.

Send changes for review

The "Changes not yet sent for review" section provides flexibility on when you send changes for review. Every change needing review is added here, listed with a description and link. Items are not sent for review until you click Send for review.

If ready to send some changes but not others, click Save for later on changes you don't want to review yet. You can view changes saved for later and undo to return them to Publishing overview. Once you send other changes for review, saved changes automatically return to Publishing overview.

Changes publish automatically as soon as reviewed and approved, unless managed publishing is on. The "Changes in review" section shows changes in the review process. If changes haven't been published yet, or are in "Ready to publish," you can move them back to "Changes not yet sent for review" to revert unwanted changes before they publish.

Note: In some cases, removing app changes from "Changes in review" or "Changes ready to publish" may still subject them to review, which could affect publishing plans.

Use managed publishing

Managed publishing is useful for tracking app changes and their review statuses, or pushing an update live at a specific time — helpful for coordinating an ad campaign, launch event, or new version release with store listing or distribution changes. Also useful for apps subject to extended review times (e.g., new developer accounts).

Before you start: Important notes about managed publishing

Prerequisites:

- Need one of these permissions to turn managed publishing on/off: Release to production/exclude devices/use Play App Signing; Release apps to testing tracks; Manage testing tracks and edit tester lists; Manage store presence; Admin (all permissions — gives full access to all other permissions).
- Your app must already be available to use Managed publishing — can't use it when publishing for the first time.

Testing:

- If using managed publishing for a product launch, Google strongly recommends publishing to a closed testing track first.

Planning for reviews and submitting updates:

- All app changes need to be processed before publishing — this can take a few hours or up to seven days (or longer in exceptional cases). Tip: build a buffer of at least a week between submitting your app and going live.
- Depending on update status, changes may not be sent for review automatically (e.g., if a previous update was rejected and you made changes to resolve it). You must go to Publishing overview and click Send for review.

Turning managed publishing on/off:

- Can be turned on or off at any time, including while a change is being reviewed and processed.

Changes that are not held back by managed publishing

Managed publishing holds back most changes until you actively decide to publish, including:

- Full and staged roll-outs of releases (with exceptions listed below)
- Launching and updating pre-registration
- Store listing changes, including custom store listings and live experiments
- App content changes
- Changes to app category
- Managed Play settings
- Changes to tester configuration for a track (e.g., new email list/Google group)

Exceptions (not held back), including but not limited to:

- Increasing an existing staged roll-out to 100%
- Updating the "Release notes" section
- Changes to device exclusion rules
- Changes to email list/Google group membership used by a testing track
- Unpublishing your app
- Changes to the In-app products page
- Price changes
- Stopping store listing experiments

Publish an app update with managed publishing

Step 1: Turn on managed publishing

- 1 Open Play Console and go to Publishing overview.
- 2 In "Managed publishing status," go to Turn on managed publishing.
- 3 Click Save.

After turning on, you'll see a green check mark and "Managed publishing is turned on" until you turn it off. Make and submit changes as normal — they won't be published until reviewed, approved, and published from the Managed publishing page.

Step 2: Track and review your changes

- 1 After making changes, select Publishing overview on the left menu.
- 2 In "Changes in review," review the table: "Item changed" column shows item name and relevant Play Console area (e.g., "Main store listing" or "App content"); "Description" gives a short summary; select the right arrow icon to go to the relevant section.

As changes are reviewed and approved, they populate "Changes ready to publish."

Step 3: Publish your app update

When "Changes in review" is empty and all changes are in "Changes ready to publish," you can publish your update:

- 1 Select Publishing overview.
- 2 Check all changes are approved and listed under Changes ready to publish.
- 3 Select Publish changes. After confirming, your update publishes within a few minutes.

Important: Once you select Review and publish and confirm, your changes become available and visible to Google Play users.

Frequently asked questions

Will submitting additional changes to my app lead to longer review times?

The SLA/turnaround time is counted from the last submitted change. Submitting a change while others are in review may push your app to the back of the review queue — Review and publish will be disabled until review is complete.

How does managed publishing work with internal testing? Will the update be covered by managed publishing?

Internal test tracks are not covered by Managed publishing. In most cases, uploading an app bundle to an Internal test track makes changes immediately available. Updates on internal test tracks aren't subject to reviews but may be subject to retroactive reviews after going live. Two caveats:

- If your app's first release roll-out is on an Internal test track, the submission must be reviewed before publishing (doesn't apply to internal test apps that aren't fully configured).
- If your previous submission was rejected, the next submission must be reviewed before publishing — uploading to an Internal test track won't make changes available until review is complete and approved.

What's the difference between "Changes not yet sent for review" and managed publishing? Will managed publishing still work as intended?

"Changes not yet sent for review" lets you decide when to send changes for approval review. "Managed publishing" lets you decide when to publish approved changes. Yes, managed publishing still works as intended.

Can we have a comprehensive list of what changes are and are not included in the "Changes not yet sent for review" feature?

The changes included are the same as those included in managed publishing.

Related content

- [Learn more about choosing when app updates are published with Managed publishing on Play Academy.](#)