More like an introduction into the subject

Apple App Store

In June this year, at WWDC, Apple announced 100 billion app downloads
Apple's App Store has passed 100 billion app downloads

But, before being downloaded, you need to be there. And good news is, they want you to be there.
This is how the App Store Review Guidelines starts.
With that in mind let's have a look at the etiquette needed to be respected to get there

App Store Review Guidelines

Introduction

We're pleased that you want to invest your talents and time to develop applications for iOS. It has been a rewarding experience - both professionally and financially - for hundreds of thousands of developers and we want to help you join this successful group. We have published our App Store Review Guidelines in the hope that they will help you steer clear of issues as you develop your App and speed you through the approval process when you submit it.

We view Apps different than books or songs, which we do not curate. If you want to criticize a religion, write a book. If you want to describe sex, write a book or a song, or create a medical App. It can get complicated, but we have decided to not allow certain kinds of content in the App Store. It may help to keep some of our broader themes in mind:

- We have lots of kids downloading lots of Apps. Parental controls work great to protect kids, but you have to do your part too. So know that we're keeping an eye out for the kids.
- We have over a million Apps in the App Store. If your App doesn't do something useful, unique or provide some form of lasting entertainment, or if your app is plain creepy, it may not be accepted.
- If your App looks like it was cobbled together in a few days, or you're trying to get your first practice App into the store to impress your friends, please brace yourself for rejection. We have lots of serious developers who don't want their quality Apps to be surrounded by amateur hour.
- We will reject Apps for any content or behavior that we believe is over the line. What line, you ask? Well, as a Supreme Court Justice once said, "I'll know it when I see it". And we think that you will also know it when you cross it.
- If your App is rejected, we have a Review Board that you can appeal to. If you run to the press and trash us, it never helps.

- If you attempt to cheat the system (for example, by trying to trick the review process, steal data from users, copy another developer's work, or manipulate the ratings) your Apps will be removed from the store and you will be expelled from the developer program.

- This is a living document, and new Apps presenting new questions may result in new rules at any time. Perhaps your App will trigger this.

Lastly, we love this stuff too, and honor what you do. We're really trying our best to create the best platform in the world for you to express your talents and make a living too. If it sounds like we're control freaks, well, maybe it's because we're so committed to our users and making sure they have a quality experience with our products. Just like almost all of you are, too.

Table of Contents

1. Terms and conditions

1.1 -- As a developer of Apps for the App Store you are bound by the terms of the Program License Agreement (PLA), Human Interface Guidelines (HIG), and any other licenses or contracts between you and Apple. The following rules and examples are intended to assist you in gaining acceptance for your App in the App Store, not to amend or remove provisions from any other agreement.

2. Functionality

2.1 Apps that crash will be rejected

2.2 Apps that exhibit bugs will be rejected

2.3 Apps that do not perform as advertised by the developer will be rejected

2.4 Apps that include undocumented or hidden features inconsistent with the description of the App will be rejected

2.5 Apps that use non-public APIs will be rejected

2.6 Apps that read or write data outside its designated container area will be rejected

2.7 Apps that download code in any way or form will be rejected

2.8 Apps that install or launch other executable code will be rejected

2.9 Apps that are "demo", "trial", or "test" versions will be rejected. Beta Apps may only be submitted through TestFlight and must follow the TestFlight guidelines

2.10 iPhone Apps must also run on iPad without modification, at iPhone resolution, and at 2X iPhone 3GS resolution

2.11 Apps that duplicate Apps already in the App Store may be rejected, particularly if there are many of them, such as fart, burp, flashlight, and Kama Sutra Apps

2.12 Apps that are not very useful, unique, are simply web sites bundled as Apps, or do not provide any lasting entertainment value may be rejected

2.13 Apps that are primarily marketing materials or advertisements will be rejected

2.14 Apps that are intended to provide trick or fake functionality that are not clearly marked as such will be rejected

2.15 Apps larger than 100MB in size will not download over cellular networks (this is automatically prohibited by the App Store)

2.16 Multitasking Apps may only use background services for their intended purposes: VoIP, audio playback, location, task completion, local notifications, etc.

2.17 Apps that browse the web must use the iOS WebKit framework and WebKit Javascript

2.18 Apps that encourage excessive consumption of alcohol or illegal substances, or encourage minors to consume alcohol or smoke cigarettes, will be rejected

2.19 Apps that provide incorrect diagnostic or other inaccurate device data will be rejected

2.20 Developers "spamming" the App Store with many versions of similar Apps will be removed from the iOS Developer Program

2.21 Apps that are simply a song or movie should be submitted to the iTunes store. Apps that are simply a book should be submitted to the iBooks Store

2.22 Apps that arbitrarily restrict which users may use the App, such as by location or carrier, may be rejected

2.23 Apps must follow the iOS Data Storage Guidelines or they will be rejected

2.24 Apps that are offered in Newsstand must comply with schedules 1, 2 and 3 of the Program License Agreement or they will be rejected

2.25 Apps that display Apps other than your own for purchase or promotion in a manner similar to or confusing with the App Store will be rejected

2.26 Apps may display and recommend apps other than your own only if the collection is designed for a specific approved need (e.g. health management, aviation, accessibility, etc.) or provides significant added value for a specific group of customers, or they will be rejected

3. Metadata (name, descriptions, ratings, rankings, etc.)

3.1 Apps or metadata that mentions the name of any other mobile platform will be rejected

3.2 Apps with placeholder text will be rejected

3.3 Apps with names, descriptions, screenshots, or previews not relevant to the content and functionality of the App will be rejected

3.4 App names in iTunes Connect and as displayed on a device should be similar, so as not to cause confusion

3.5 Small and large App icons should be similar, so as to not to cause confusion

3.6 Apps with App icons, screenshots, and previews that do not adhere to the 4+ age rating will be rejected

3.7 Apps with Category and Genre selections that are not appropriate for the App content will be rejected

3.8 Developers are responsible for assigning appropriate ratings to their Apps. Inappropriate ratings may be changed/deleted by Apple

3.9 Developers are responsible for assigning appropriate keywords for their Apps. Inappropriate keywords may be changed/deleted by Apple

3.10 Developers who attempt to manipulate or cheat the user reviews or chart ranking in the App Store with fake or paid reviews, or any other inappropriate methods will be removed from the iOS Developer Program

3.11 Apps that recommend that users restart their iOS device prior to installation or launch may be rejected

3.12 Apps should have all included URLs fully functional when you submit it for review, such as support and privacy policy URLs

3.13 Apps with screenshots, previews, and marketing text that do not clearly identify supplemental content or items that must be purchased separately (e.g. using IAP) will be rejected

3.14 App previews may only use video screen captures of the app, voice-overs, and textual and design overlays, or the app will be rejected

3.15 Apps with previews that display personal information of a real person without permission will be rejected

3.16 App previews may only include music that is licensed for that purpose in all selected territories

3.17 App previews that include content played or streamed via the app (e.g. iTunes playlist, YouTube streaming video) that is not licensed for use in the preview will be rejected

4. Location

4.1 Apps that do not notify and obtain user consent before collecting, transmitting, or using location data will be rejected

4.2 Apps that use location-based APIs for automatic or autonomous control of vehicles, aircraft, or other devices will be rejected

4.3 Apps that use location-based APIs for emergency services will be rejected

4.4 Location data can only be used when directly relevant to the features and services provided by the App to the user or to support approved advertising uses

5. Push Notifications

5.1 Apps that provide Push Notifications without using the Apple Push Notification (APN) API will be rejected

5.2 Apps that use the APN service without obtaining a Push Application ID from Apple will be rejected

5.3 Apps that send Push Notifications without first obtaining user consent, as well as apps that require Push Notifications to function, will be rejected

5.4 Apps that send sensitive personal or confidential information using Push Notifications will be rejected

5.5 Apps that use Push Notifications to send unsolicited messages, or for the purpose of phishing or spamming will be rejected

5.6 Apps cannot use Push Notifications to send advertising, promotions, or direct marketing of any kind

5.7 Apps cannot charge users for use of Push Notifications

5.8 Apps that excessively use the network capacity or bandwidth of the APN service or unduly burden a device with Push Notifications will be rejected

5.9 Apps that transmit viruses, files, computer code, or programs that may harm or disrupt the normal operation of the APN service will be rejected

6. Game Center

6.1 Apps that display any Player ID to end users or any third party will be rejected

6.2 Apps that use Player IDs for any use other than as approved by the Game Center terms will be rejected

6.3 Developers that attempt to reverse lookup, trace, relate, associate, mine, harvest, or otherwise exploit Player IDs, aliases, or other information obtained through Game Center will be removed from the iOS Developer Program

6.4 Game Center information, such as Leaderboard scores, may only be used in Apps approved for use with Game Center

6.5 Apps that use the Game Center service to send unsolicited messages, or for the purpose of phishing or spamming will be rejected

6.6 Apps that excessively use the network capacity or bandwidth of Game Center will be rejected

6.7 Apps that transmit viruses, files, computer code, or programs that may harm or disrupt the normal operation of the Game Center service will be rejected

7. Advertising

7.1 Apps that artificially increase the number of impressions or click-throughs of ads will be rejected

7.2 Apps that contain empty iAd banners will be rejected

7.3 Apps that are designed predominantly for the display of ads will be rejected

8. Content and Intellectual Property Rights

8.1 Apps must comply with all terms and conditions explained in the Guidelines for Using Apple Trademarks and Copyrights and the Apple Trademark List

8.2 Apps that suggest or infer that Apple is a source or supplier of the App, or that Apple endorses any particular representation regarding quality or functionality will be rejected

8.3 Apps that appear confusingly similar to an existing Apple product, interface, or advertising theme will be rejected
8.4 Apps that misspell Apple product names in their App name (i.e., GPS for Iphone, iTune) will be rejected
8.5 Apps may not use protected third party material such as trademarks, copyrights, patents or violate 3rd party terms of use. Authorization to use such material must be provided upon request
8.6 Apps that include the ability to download music or video content from third party sources (e.g. YouTube, SoundCloud, Vimeo, etc) without explicit authorization from those sources will be rejected

9. Media content

9.1 Apps that do not use the MediaPlayer framework to access media in the Music Library will be rejected
9.2 App user interfaces that mimic any iPod or iTunes interface will be rejected
9.3 Audio streaming content over a cellular network may not use more than 5MB over 5 minutes
9.4 Video streaming content over a cellular network longer than 10 minutes must use HTTP Live Streaming and include a baseline 192 kbps or lower HTTP Live stream

10. User interface

10.1 Apps must comply with all terms and conditions explained in the Apple iOS Human Interface Guidelines
10.2 Apps that look similar to Apps bundled on iOS or Watch OS devices, including the App Store, iTunes Store, and iBooks Store, will be rejected
10.3 Apps that do not use system provided items, such as buttons and icons, correctly and as described in the Apple iOS Human Interface Guidelines may be rejected
10.4 Apps that create alternate desktop/home screen environments or simulate multi-App widget experiences will be rejected
10.5 Apps that alter the functions of standard switches, such as the Volume Up/Down and Ring/Silent switches, will be rejected
10.6 Apple and our customers place a high value on simple, refined, creative, well thought through interfaces. They take more work but are worth it. Apple sets a high bar. If your user interface is complex or less than very good, it may be rejected
10.7 Watch Apps whose primary function is telling time will be rejected

11. Purchasing and currencies

11.1 Apps that unlock or enable additional features or functionality with mechanisms other than the App Store will be rejected
11.2 Apps utilizing a system other than the In-App Purchase API (IAP) to purchase content, functionality, or services in an App will be rejected
11.3 Apps using IAP to purchase physical goods or goods and services used outside of the App will be rejected
11.4 Apps that use IAP to purchase credits or other currencies must consume those credits within the App
11.5 Apps that use IAP to purchase credits or other currencies that expire will be rejected
11.6 Content subscriptions using IAP must last a minimum of 7 days and be available to the user from all of their iOS devices

11.7 Apps that use IAP to purchase items must assign the correct Purchasability type

11.8 Apps that use IAP to purchase access to built-in capabilities provided by iOS, such as the camera or the gyroscope, will be rejected

11.9 Apps containing content or services that expire after a limited time will be rejected, except for specific approved content (e.g. films, television programs, music, books)

11.10 Insurance Apps must be free, in legal-compliance in the regions distributed, and cannot use IAP

11.11 In general, the more expensive your App, the more thoroughly we will review it

11.12 Apps offering subscriptions must do so using IAP, Apple will share the same 70/30 revenue split with developers for these purchases, as set forth in the Program License Agreement

11.13 Apps that link to external mechanisms for purchases or subscriptions to be used in the App, such as a "buy" button that goes to a web site to purchase a digital book, will be rejected

11.14 Apps can read or play approved content (specifically magazines, newspapers, books, audio, music, video and cloud storage) that is subscribed to or purchased outside of the App, as long as there is no button or external link in the App to purchase the approved content. Apple will only receive a portion of revenues for content purchased inside the App

11.15 Apps may only use auto-renewing subscriptions for periodicals (newspapers, magazines), business Apps (enterprise, productivity, professional creative, cloud storage), and media Apps (video, audio, voice), or the App will be rejected

11.16 Apps may enable additional approved features or functionality when used in combination with specific approved physical products (such as a toy) as long as the additional features and functionality are either completely dependent on such hardware (for example an App that is used to control a telescope) or also available through the App without the physical products, such as by way of reward for achievement or by use of IAP

11.17 Apps may facilitate transmission of approved virtual currencies provided that they do so in compliance with all state and federal laws for the territories in which the app functions

12. Scraping and aggregation

12.1 Apps that scrape any information from Apple sites (for example from apple.com, iTunes Store, App Store, iTunes Connect, Apple Developer Programs, etc.) or create rankings using content from Apple sites and services will be rejected

12.2 Apps may use approved Apple RSS feeds such as the iTunes Store RSS feed

12.3 Apps that are simply web clippings, content aggregators, or a collection of links, may be rejected

13. Damage or injury

13.1 Apps that encourage users to use an Apple Device in a way that may cause damage to the device will be rejected

13.2 Apps that rapidly drain the device's battery or generate excessive heat will be rejected

13.3 Apps whose use may result in physical harm may be rejected

14. Personal attacks

14.1 Any App that is defamatory, offensive, mean-spirited, or likely to place the targeted individual or group in harm's way will be rejected

14.2 Professional political satirists and humorists are exempt from the ban on offensive or mean-spirited commentary

14.3 Apps that display user generated content must include a method for filtering objectionable material, a mechanism for users to flag offensive content, and the ability to block abusive users from the service

15. Violence

15.1 Apps portraying realistic images of people or animals being killed or maimed, shot, stabbed, tortured or injured will be rejected

15.2 Apps that depict violence or abuse of children will be rejected

15.3 "Enemies" within the context of a game cannot solely target a specific race, culture, a real government or corporation, or any other real entity

15.4 Apps involving realistic depictions of weapons in such a way as to encourage illegal or reckless use of such weapons will be rejected

15.5 Apps that include games of Russian roulette will be rejected

16. Objectionable content

16.1 Apps that present excessively objectionable or crude content will be rejected

16.2 Apps that are primarily designed to upset or disgust users will be rejected

17. Privacy

17.1 Apps cannot transmit data about a user without obtaining the user's prior permission and providing the user with access to information about how and where the data will be used

17.2 Apps that require users to share personal information, such as email address and date of birth, in order to function will be rejected

17.3 Apps may ask for date of birth (or use other age-gating mechanisms) only for the purpose of complying with applicable children's privacy statutes, but must include some useful functionality or entertainment value regardless of the user's age

17.4 Apps that collect, transmit, or have the capability to share personal information (e.g. name, address, email, location, photos, videos, drawings, the ability to chat, other personal data, or persistent identifiers used in combination with any of the above) from a minor must comply with applicable children's privacy statutes, and must include a privacy policy

17.5 Apps that include account registration or access a user’s existing account must include a privacy policy or they will be rejected

18. Pornography

18.1 Apps containing pornographic material, defined by Webster's Dictionary as "explicit descriptions or displays of sexual organs or activities intended to stimulate erotic rather than aesthetic or emotional feelings", will be rejected

18.2 Apps that contain user generated content that is frequently pornographic (e.g. "Chat Roulette" Apps) will be rejected

19. Religion, culture, and ethnicity

19.1 Apps containing references or commentary about a religious, cultural or ethnic group that are defamatory, offensive, mean-spirited or likely to expose the targeted group to harm or violence will be rejected

19.2 Apps may contain or quote religious text provided the quotes or translations are accurate and not misleading. Commentary should be educational or informative rather than inflammatory

20. Contests, sweepstakes, lotteries, raffles, and gambling

20.1 Sweepstakes and contests must be sponsored by the developer/company of the App

20.2 Official rules for sweepstakes and contests must be presented in the App and make it clear that Apple is not a sponsor or involved in the activity in any manner

20.3 It must be permissible by law for the developer to run a lottery App, and a lottery App must have all of the following characteristics: consideration, chance, and a prize

20.4 Apps that allow a user to directly purchase a raffle ticket in the App will be rejected

20.5 Apps that offer real money gaming (e.g. sports betting, poker, casino games, horse racing) or lotteries must have necessary licensing and permissions in the locations where the App is used, must be restricted to those locations, and must be free on the App Store

20.6 Apps that use IAP to purchase credit or currency to use in conjunction with real money gaming will be rejected

21. Charities and contributions

21.1 Apps that include the ability to make donations to recognized charitable organizations must be free

21.2 The collection of charitable donations must be done via a web site in Safari or an SMS

22. Legal requirements

22.1 Apps must comply with all legal requirements in any location where they are made available to users. It is the developer's obligation to understand and conform to all local laws

22.2 Apps that contain false, fraudulent or misleading representations or use names or icons similar to other Apps will be rejected

22.3 Apps that solicit, promote, or encourage criminal or clearly reckless behavior will be rejected

22.4 Apps that enable illegal file sharing will be rejected

22.5 Apps that are designed for use as illegal gambling aids, including card counters, will be rejected

22.6 Apps that enable anonymous or prank phone calls or SMS/MMS messaging will be rejected

22.7 Developers who create Apps that surreptitiously attempt to discover user passwords or other private user data will be removed from the iOS Developer Program

22.8 Apps that contain DUI checkpoints that are not published by law enforcement agencies, or encourage and enable drunk driving, will be rejected

22.9 Apps that calculate medicinal dosages must be submitted by the manufacturer of those medications or recognized institutions such as hospitals, insurance companies, and universities

22.10 Apps that use iTunes music previews in an unauthorized manner will be rejected

23. Wallet

23.1 Wallet passes can be used to make or receive payments, transmit offers, or offer identification (such as movie tickets, airline tickets, coupons and reward offers). Other uses may result in the rejection of the App and the revocation of Wallet credentials

23.2 Passes must include valid contact information from the issuer of the pass or the App will be rejected and Wallet credentials may be revoked

23.3 Passes must be signed by the entity that will be distributing the pass under its own name, trademark, or brand or the App will be rejected and Wallet credentials may be revoked

24. Kids Category

24.1 Apps in the Kids Category must include a privacy policy and must comply with applicable children's privacy statutes

24.2 Apps in the Kids Category may not include behavioral advertising (e.g. the advertiser may not serve ads based on the user's activity within the App), and any contextual ads presented in the App must be appropriate for kids

24.3 Apps in the Kids Category must get parental permission or use a parental gate before allowing the user to link out of the app or engage in commerce

24.4 Apps in the Kids Category must be made specifically for kids ages 5 and under, ages 6-8, or ages 9-11

25. Extensions

25.1 Apps hosting extensions must comply with the App Extension Programming Guide

25.2 Apps hosting extensions must provide some functionality (help screens, additional settings) or they will be rejected

25.3 Apps hosting extensions that include marketing, advertising, or in-app purchases in their extension view will be rejected

25.4 Keyboard extensions must provide a method for progressing to the next keyboard

25.5 Keyboard extensions must remain functional with no network access or they will be rejected

25.6 Keyboard extensions must provide Number and Decimal keyboard types as described in the App Extension Programming Guide or they will be rejected

25.7 Apps offering Keyboard extensions must have a primary category of Utilities and a privacy policy or they will be rejected

25.8 Apps offering Keyboard extensions may only collect user activity to enhance the functionality of their keyboard extension on the iOS device or they may be rejected

26. HomeKit

26.1 Apps using the HomeKit framework must have a primary purpose of providing home automation services

26.2 Apps using the HomeKit framework must indicate this usage in their marketing text and they must provide a privacy policy or they will be rejected

26.3 Apps must not use data gathered from the HomeKit APIs for advertising or other use-based data mining

26.4 Apps using data gathered from the HomeKit API for purposes other than improving the user experience or hardware/software performance in providing home automation functionality will be rejected

27. HealthKit and Human Subject Research

27.1 Apps using the HealthKit framework or conducting human subject research for health purposes, such as through the use of ResearchKit, must comply with applicable law for each Territory in which the App is made available, as well as Sections 3.3.28 and 3.3.39 of the iOS Developer Program License Agreement

27.2 Apps that write false or inaccurate data into HealthKit will be rejected

27.3 Apps using the HealthKit framework that store users' health information in iCloud will be rejected

27.4 Apps may not use or disclose to third parties user data gathered from the HealthKit API or from health-related human subject research for advertising or other use-based data mining purposes other than improving health, or for the purpose of health research

27.5 Apps that share user data acquired via the HealthKit API with third parties without user consent will be rejected

27.6 Apps using the HealthKit framework must indicate integration with the Health app in their marketing text and must clearly identify the HealthKit functionality in the app's user interface

27.7 Apps using the HealthKit framework or conducting human subject research must provide a privacy policy or they will be rejected

27.8 Apps that provide diagnoses, treatment advice, or control hardware designed to diagnose or treat medical conditions that do not provide written regulatory approval upon request will be rejected

27.9 Apps conducting health-related human subject research must obtain consent from participants or, in the case of minors, their parent or guardian. Such consent must include the (a) nature, purpose, and duration of the research; (b) procedures, risks, and benefits to the participant; (c) information about confidentiality and handling of data (including any sharing with third parties); (d) a point of contact for participant questions; and (e) the withdrawal process

27.10 Apps conducting health-related human subject research must secure approval from an independent ethics review board. Proof of such approval must be provided upon request.

28. TestFlight

28.1 Apps may only use TestFlight to beta test apps intended for public distribution and must comply with the full App Review Guidelines

28.2 Apps using TestFlight must be submitted for review whenever a build contains material changes to content or functionality

28.3 Apps using TestFlight may not be distributed to testers in exchange for compensation of any kind

29. Apple Pay

29.1 Apps using Apple Pay must provide all material purchase information to the user prior to sale of any good or service or they will be rejected; Apps using Apple Pay to offer recurring payments must, at a minimum, disclose the length of the renewal term and the fact that it will continue until canceled, what will be provided during each period, the charges that will be billed to the customer, and how to cancel.

29.2 Apps using Apple Pay must use Apple Pay branding and user interface elements correctly and as described in the Apple Pay Identity Guidelines or they will be rejected

29.3 Apps using Apple Pay as a purchasing mechanism may not offer goods or services that violate the law of any territory in which the good or service will be delivered and may not be used for any illegal purpose

29.4 Apps using Apple Pay must provide a privacy policy or they will be rejected

29.5 Apps using Apple Pay may only share user data acquired via Apple Pay with third parties when provided to facilitate or improve delivery of goods and services or to comply with legal requirements

Living document

This document represents our best efforts to share how we review Apps submitted to the App Store, and we hope it is a helpful guide as you develop and submit your Apps. It is a living document that will evolve as we are presented with new Apps and situations, and we'll update it periodically to reflect these changes.

Thank you for developing for iOS. Even though this document is a formidable list of what not to do, please also keep in mind the much shorter list of what you must do. Above all else, join us in trying to surprise and delight users. Show them their world in innovative ways, and let them interact with it like never before. In our experience, users really respond to polish, both in functionality and user interface. Go the extra mile. Give them more than they expect. And take them places where they have never been before. We are ready to help.

In addition Apple has a store for desktop applications, with the following guidelines

Mac App Store Review Guidelines

Introduction

The Mac App Store brings the phenomenal success and user experience of the App Store to the Mac, and has revolutionized the way desktop Apps are developed and distributed. It is the best place for users to discover and purchase innovative and exciting new Mac Apps.

We are thrilled with the success our developers have had in bringing their Apps to the tens of millions of Mac users in over 150 countries around the world and want to help you bring your new Apps to the Mac App Store so we can continue to surprise and delight our customers.

To ensure that all Apps are reliable, perform as advertised, and are free of offensive material, we review each App with the Mac App Store Guidelines in mind. These guidelines are designed to prepare your Apps so they will sail through our approval process.

Table of Contents

10. Personal attacks
11. Violence
12. Objectionable content
13. Privacy
14. Pornography
15. Religion, culture, and ethnicity
16. Contests, sweepstakes, lotteries, and raffles
17. Charities and contributions
18. Legal requirements
19. Push Notifications
20. Game Center
21. Safari Extensions

The development of the app needs to take in consideration these guidelines; to avoid rejection and Apple reminds us by publishing the most common mistakes that determine an app rejection

Common App Rejections

Crashes and Bugs

You should submit your app for review only when it is complete and ready to be published. Make sure to thoroughly test your app on devices and fix all bugs before submitting.

Broken Links

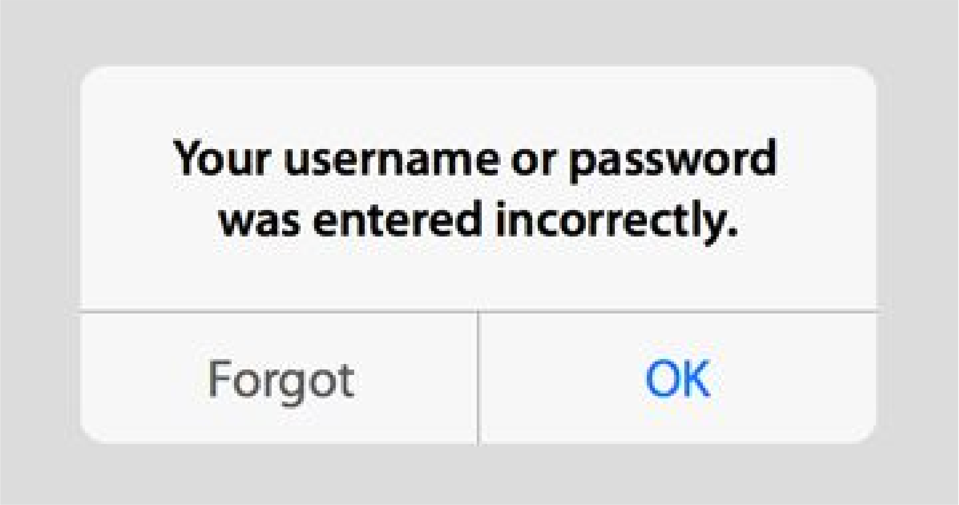
All links in your app must be functional. A link to user support with up-to-date contact information is required for all apps, and if you're offering auto-renewable or free subscriptions or your app is in the Kids Category, you must also provide a link to your privacy policy.

Placeholder Content

Finalize all images and text in your app before sending it in for review. Apps that are still in progress and contain placeholder content are not ready to be distributed and cannot be approved.

Incomplete Information

Enter all of the details needed to review your app in the App Review Information section of iTunes Connect. If some features require signing in, provide a valid demo account username and password. If there are special configurations to set, include the specifics. If features require an environment that is hard to replicate or require specific hardware, be prepared to provide a demo video or the hardware. Also, please make sure your contact information is complete and up-to-date.



Your username or password was entered incorrectly.

Inaccurate Descriptions

Your app description and screenshots should clearly and accurately convey your app's functionality. This helps users understand your app and makes for a positive App Store experience.

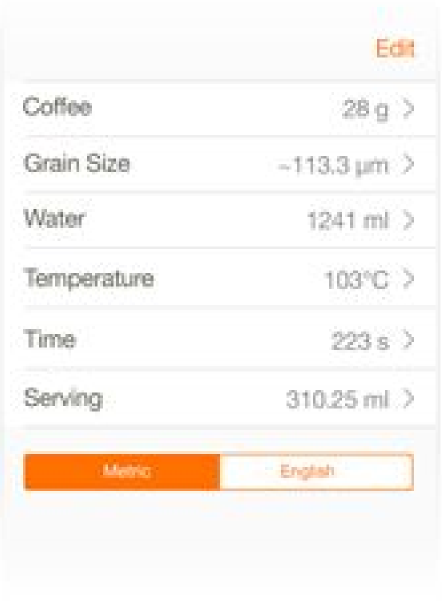
Misleading Users

Your app must perform as advertised and should not give users the impression the app is something it is not. If your app appears to promise certain features and functionalities, it needs to deliver.

Substandard User Interface

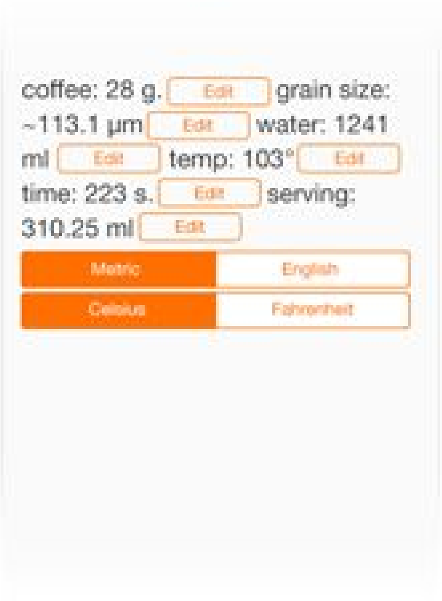
Apple places a high value on clean, refined, and user-friendly interfaces. Make sure your UI meets these requirements by planning your design carefully and following our design guides and UI Design Dos and Don'ts.

✓



Coffee	28 g >
Grain Size	~113.3 µm >
Water	1241 ml >
Temperature	103°C >
Time	223 s >
Serving	310.25 ml >

✗



coffee: 28 g. grain size: ~113.1 µm water: 1241 ml temp: 103° time: 223 s. serving: 310.25 ml

Advertisements

When submitting your app for review, you'll be asked whether your app uses the Advertising Identifier (IDFA) to serve advertisements. If you indicate that your app uses the IDFA, but it does not have ad functionality or does not display ads properly, your app may be rejected. Make sure to test your app on an iOS device to verify that ads work correctly. Similarly, if you indicate that your app does not use the IDFA, but it does, your app will be put into the "Invalid Binary" status.

Web clippings, content aggregators, or a collections of links

Your app should be engaging and useful, and make the most of the features unique to iOS. Websites served in an iOS app web content that is not formatted for iOS, and limited web interactions do not make a quality app.

Repeated Submission of Similar Apps

Submitting several apps that are essentially the same ties up the App Review process and risks the rejection of your app Improve your review experience — and the experience of your future users — by thoughtfully combining your apps into one.

Not enough lasting value

If your app doesn't offer much functionality or content, or only applies to a small niche market, it may not be approved. Before creating your app, take a look at the apps in your category on the App Store and consider how you can provide an even better user experience.

Top 10 reasons for app rejections during the

7-day period ending June 17, 2015.

- 14% More information needed
- 10% Guideline 2.2: Apps that exhibit bugs will be rejected
- 5% Guideline 10.6: Apple and our customers place a high value on simple, refined, creative, well thought through interfaces. They take more work but are worth it. Apple sets a high bar. If your user interface is complex or less than very good, it may be rejected
- 3% Guideline 3.8: Developers are responsible for assigning appropriate ratings to their Apps. Inappropriate ratings may be changed/deleted by Apple
- 3% Guideline 3.1: Apps or metadata that mentions the name of any other mobile platform will be rejected
- 3% Guideline 2.1: Apps that crash will be rejected
- 3% Guideline 3.3: Apps with names, descriptions, screenshots, or previews not relevant to the content and functionality of the App will be rejected
- 3% Guideline 22.2: Apps that contain false, fraudulent or misleading representations or use names or icons similar to other Apps will be rejected
- 2% Guideline 3.4: App names in iTunes Connect and as displayed on a device should be similar, so as not to cause confusion
- 2% 17.2: Apps that require users to share personal information, such as email address and date of birth, in order to function will be rejected

Aside from the defects that should not be present in an app, there are guidelines on how to design an app to meet the standards

Designing Great Apps

Introducing the 2015 Apple Design Award winners.

Read about the developers and their apps, which reflect the best in design, innovation, and technology on iOS, OS X, and watchOS. See this year’s winners

Watch the latest from WWDC 2015.

Gain valuable insights on best practices in typography, animation, and navigation. Learn critical prototyping techniques to help you quickly create the best visual and interaction design. Watch the session videos

- Designing for Future Hardware
- Designing for Apple Watch
- Designing with Animation
- Introducing the New System Font
- Apple Watch Design Tips and Tricks

San Francisco Fonts

Get your apps ready with San Francisco, the new system font for iOS 9, OS X El Capitan, and watchOS 2. Download the San Francisco fonts

Human Interface Guidelines

- Apple Watch Human Interface Guidelines
- OS X Human Interface Guidelines
- iOS Human Interface Guidelines
- UI Design Dos and Don't

And i will present here the do’s and dont’s, given we will have a future presentation dedicated for design

UI Design Do’s and Don’ts

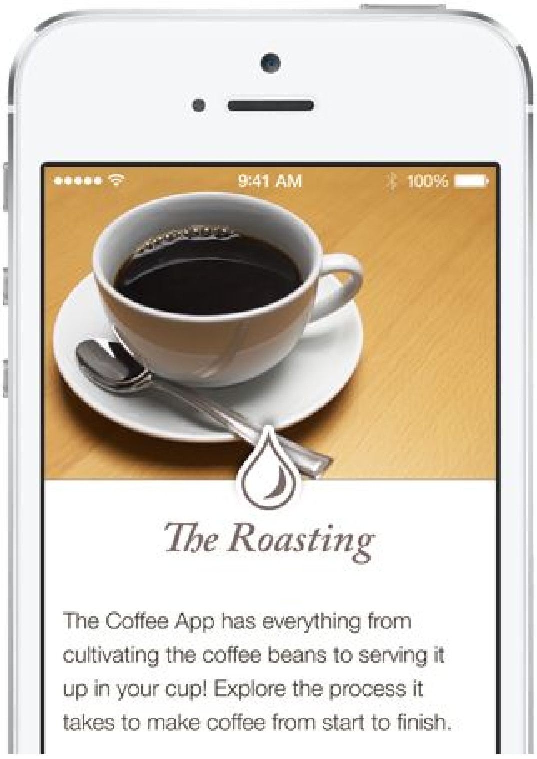
iOS raises the bar for excellence in user interface design and offers great opportunities for you to deliver engaging and unique user experiences. Consider these common design concepts before you start coding to enhance the usability and appeal of your apps. To see more recommendations for great interface design, read the iOS Human Interface Guidelines.

Interactivity Readability Graphics Clarity

Formatting Content

Create a layout that fits the screen of an iOS device. Users should see primary content without zooming or scrolling horizontally. Learn more

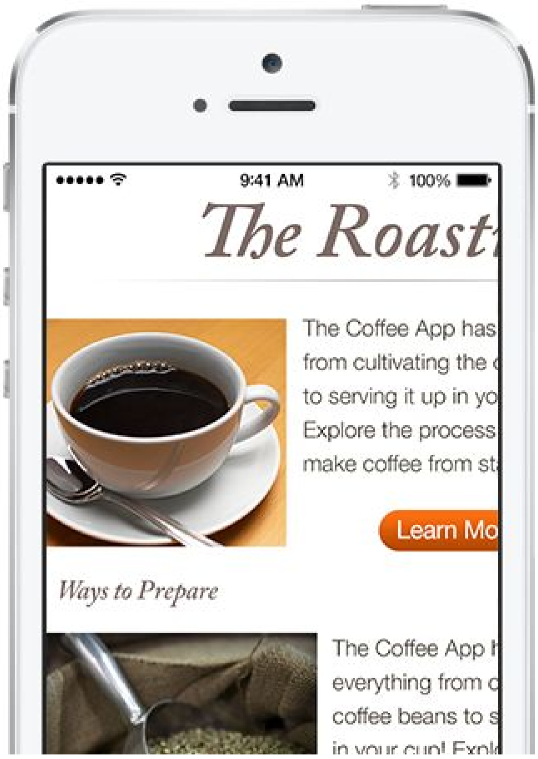
✓



The Roasting

The Coffee App has everything from cultivating the coffee beans to serving it up in your cup! Explore the process it takes to make coffee from start to finish.

✗



The Roast

The Coffee App has from cultivating the to serving it up in yo Explore the process make coffee from st

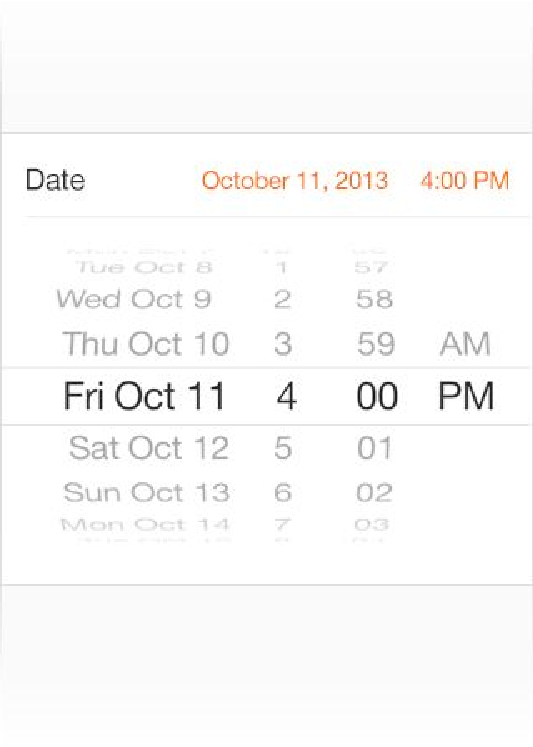
Ways to Prepare

The Coffee App h everything from c coffee beans to s in your cup! Expl

Touch Controls

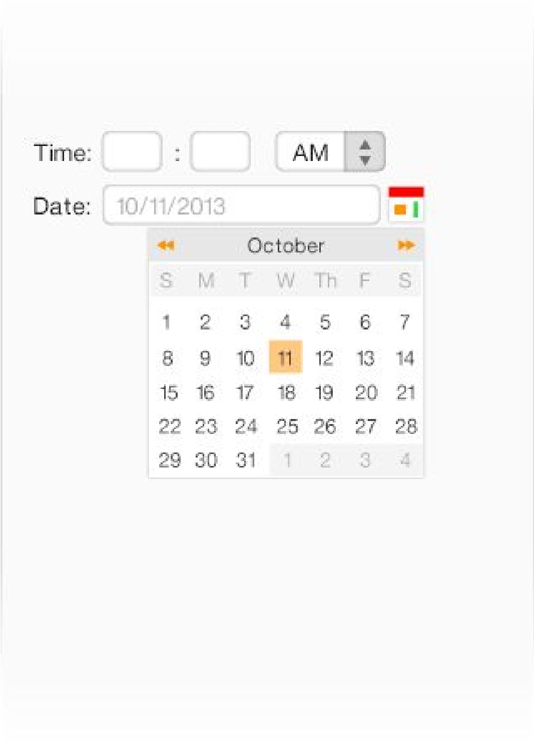
Use UI elements that are designed for touch gestures to make interaction with your app feel easy and natural. Learn more

✓



Date October 11, 2013 4:00 PM

✗



Time: : AM

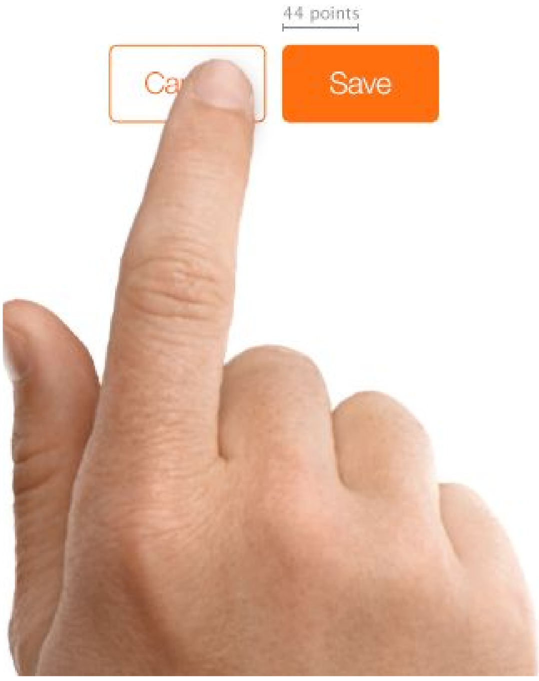
Date: 10/11/2013

Hit Targets

Create controls that measure at least 44 points x 44 points so they can be accurately tapped with a finger. Learn more

✓

✗

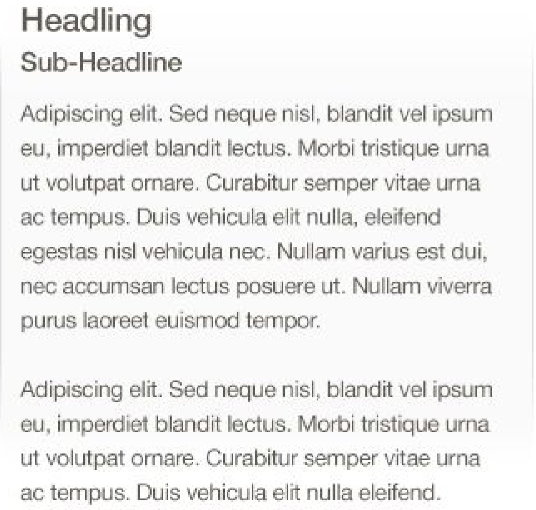


Text Size

Text should be at least 11 points so it's legible at a typical viewing distance without zooming. Learn more

✓

✗

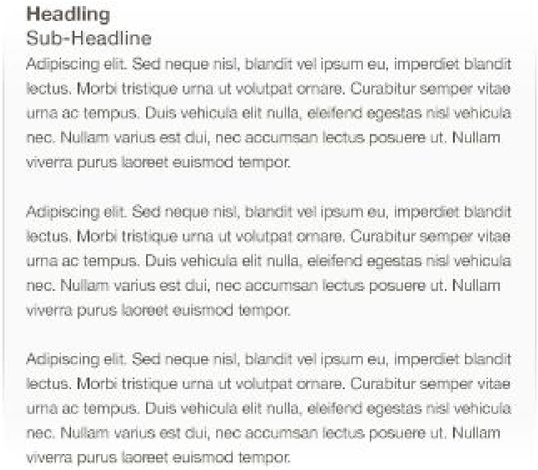


Headling

Sub-Headline

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla eleifend.



Headling

Sub-Headline

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

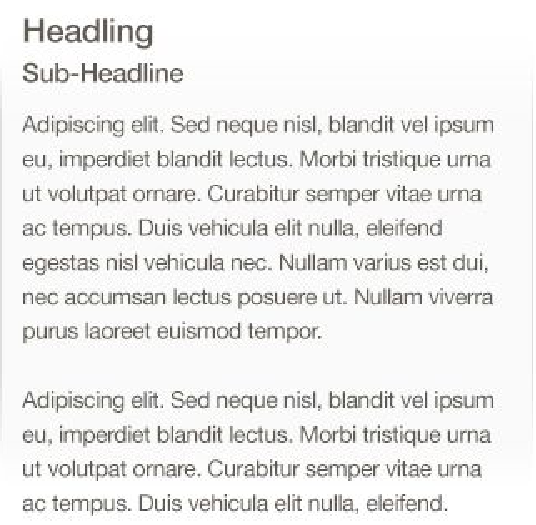
Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

Contrast

Make sure there is ample contrast between the font color and the background so text is legible. Learn more

✓

✗

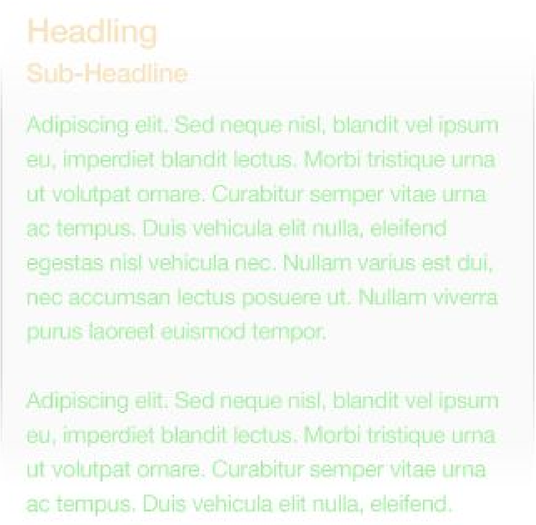


Headling

Sub-Headline

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend.



Headling

Sub-Headline

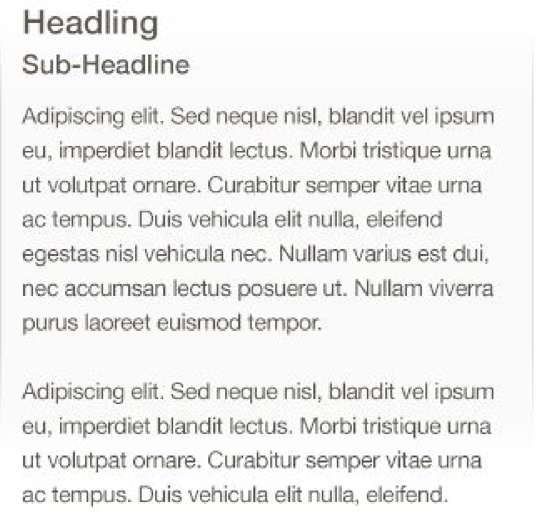
Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend.

Spacing

Don't let text overlap. Improve legibility by increasing line height or letter spacing. Learn more

✓ ✗

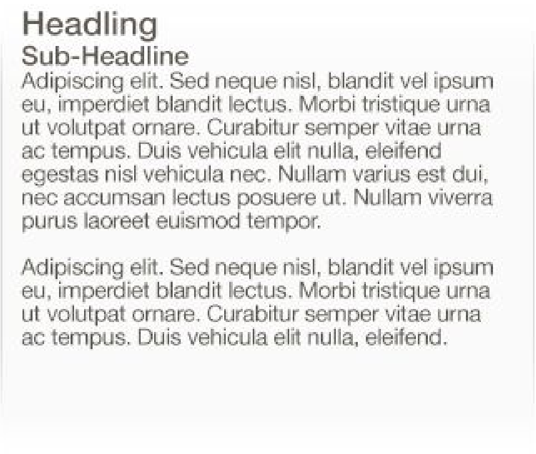


Headling

Sub-Headline

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend.



Headling

Sub-Headline

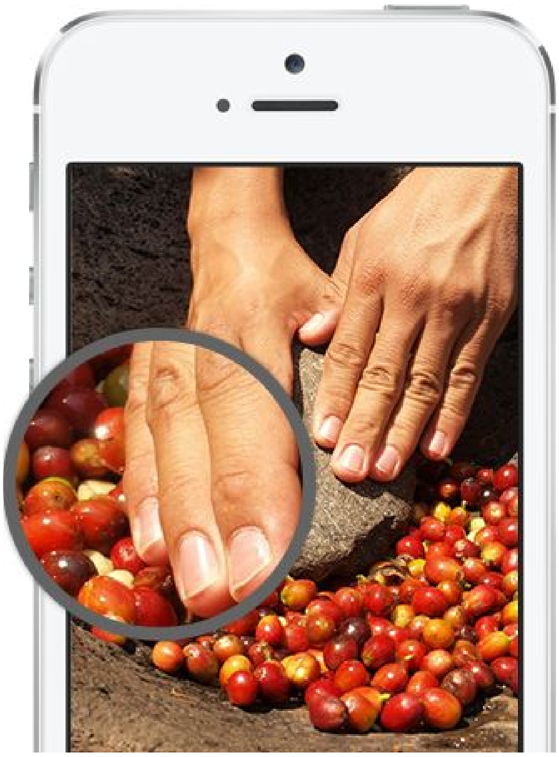
Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend egestas nisl vehicula nec. Nullam varius est dui, nec accumsan lectus posuere ut. Nullam viverra purus laoreet euismod tempor.

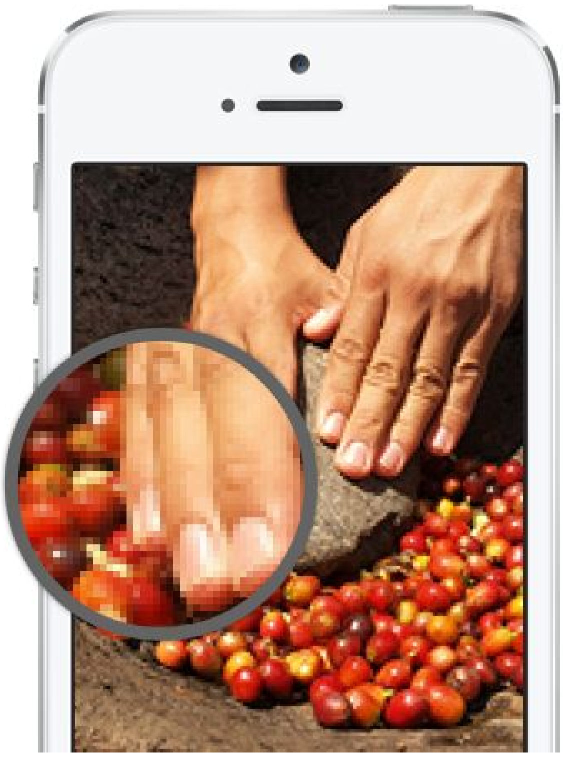
Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus. Duis vehicula elit nulla, eleifend.

High Resolution

Provide high-resolution versions of all image assets. Images that are not @2x and @3x will appear blurry on the Retina display. Learn more

✓ ✗



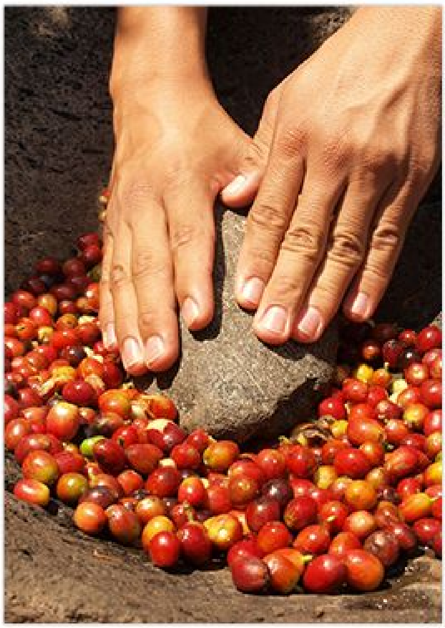


Distortion

Always display images at their intended aspect ratio to avoid distortion. Learn more.

✓

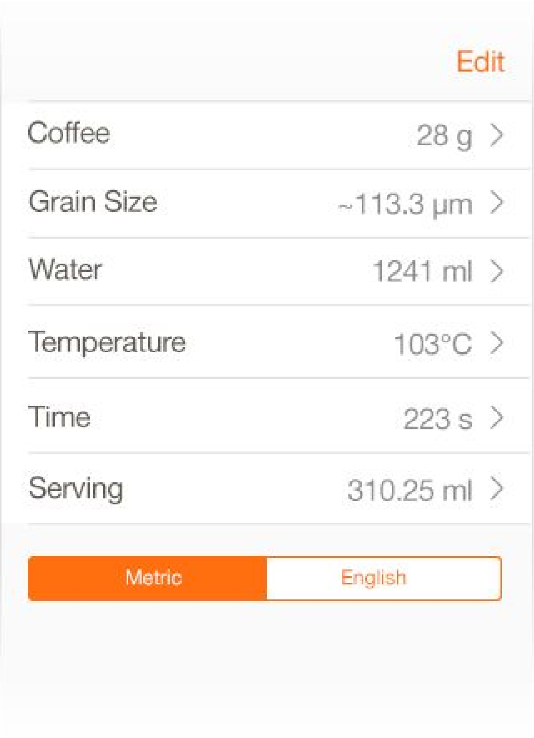
✗



Organization

Create an easy-to-read layout that puts controls close to the content they modify. Learn more

✓ ✗



Edit

Coffee 28 g >

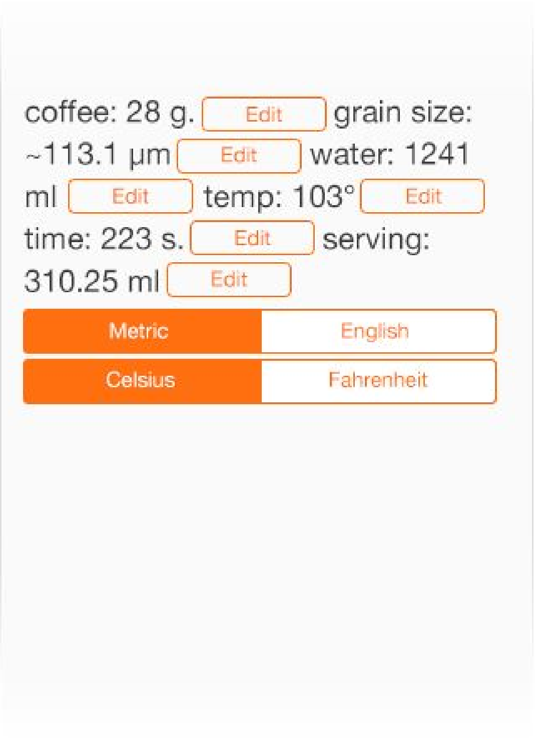
Grain Size ~113.3 µm >

Water 1241 ml >

Temperature 103°C >

Time 223 s >

Serving 310.25 ml >

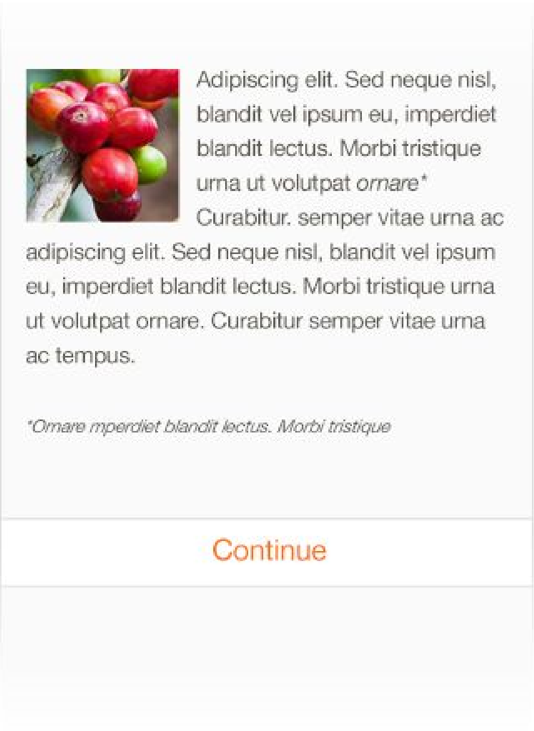


coffee: 28 g. Edit grain size: ~113.1 µm Edit water: 1241 ml Edit temp: 103° Edit time: 223 s. Edit serving: 310.25 ml Edit

Alignment

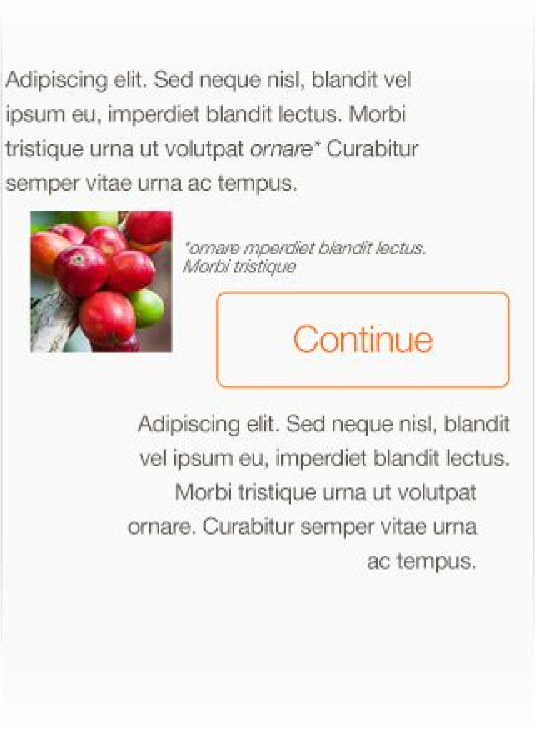
Align text, images, and buttons to show users how information is related. Learn more

✓ ✗



Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat _ornare_* Curabitur. semper vitae urna ac adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus.

*Ornare mperdiet blandit lectus. Morbi tristique



Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat _ornare_* Curabitur semper vitae urna ac tempus.

*ornare mperdiet blandit lectus. Morbi tristique

Adipiscing elit. Sed neque nisl, blandit vel ipsum eu, imperdiet blandit lectus. Morbi tristique urna ut volutpat ornare. Curabitur semper vitae urna ac tempus.

Additional Design Resources

View video presentations and guides for designing great apps. Learn more

Google Play

Since March 2015 Google has introduced an app approval process

Announced by Eunice Kim, Product Manager for Google Play , and then published by tech media App Submissions On Google Play Now Reviewed By Staff, Will Include Age-Based Ratings the approval is based on these guidelines

Google Play Developer Programme Policies

The policies listed below play an important role in maintaining a positive experience for everyone using Google Play. Defined terms used here have the same meaning as in the Developer Distribution Agreement. Make sure that you check back from time to time, as these policies may change.

Content Policies

Our content policies apply to any content to which your app displays or links, including any ads it shows to users and any user-generated content to which it hosts or links. Furthermore, they apply to any content from your developer account which is publicly displayed in Google Play, including your developer name and the landing page of your listed developer website. In addition to complying with these policies, the content of your app must be rated in accordance with our Content Rating Guidelines.

- **Sexually Explicit Material:** Apps that contain or promote pornography are prohibited; this includes sexually explicit or erotic content, icons, titles or descriptions. Google has a zero-tolerance policy against child sexual abuse imagery. If we become aware of content with child sexual abuse imagery, we will report it to the appropriate authorities and delete the Google Accounts of those involved with the distribution.
- **Violence and Bullying:** Depictions of gratuitous violence are not allowed. Apps should not contain materials that threaten, harass or bully other users.
- **Hate Speech:** We don't allow content advocating against groups of people based on their race or ethnic origin, religion, disability, gender, age, nationality, veteran status, sexual orientation or gender identity.
- **Sensitive Events:** We don't allow content which may be deemed as capitalising on or lacking reasonable sensitivity towards a natural disaster, atrocity, conflict, death or other tragic event.
- **Impersonation or Deceptive Behaviour:** Don't pretend to be someone else, and don't give any impression that your app is authorised by or produced by another company or organisation if that is not the case. Products must provide accurate disclosure of their functionality and should perform as reasonably expected by the user.
    - Products or the ads they contain must not mimic functionality or warnings from the operating system or that other apps.
    - Products must not contain false or misleading information or claims in any content, title, icon, description or screenshots.
    - Developers must not divert users or provide links to any other site that mimics or passes itself off as another app or service.
    - Apps must not have names or icons that appear confusingly similar to existing products or to apps supplied with the device (such as Camera, Gallery or Messaging).

- **Intellectual Property:** Don’t infringe on the intellectual property rights of others, (including patent, trademark, trade secret, copyright and other proprietary rights), or encourage or induce infringement of intellectual property rights. We will respond to clear notices of alleged copyright infringement. For more information or to file a DMCA request, please visit our copyright procedures.
- **Personal and Confidential Information:** We don't allow unauthorised publishing or disclosure of people's private and confidential information, such as credit card numbers, government identification numbers, driver's and other licence numbers, non-public contacts or any other information that is not publicly accessible.
- **Illegal Activities:** Keep it legal. Don't engage in unlawful activities on this product, such as the sale of prescription drugs without a prescription.
- **Gambling:** We don’t allow content or services that facilitate online gambling, including, but not limited to, online casinos, sports betting and lotteries or games of skill that offer prizes of cash or other value.
- **Dangerous Products:** We don't allow content that harms, interferes with the operation of, or accesses in an unauthorised manner, networks, servers, application programming interfaces (APIs) or other infrastructure.
    - Don't transmit or link to viruses, worms, defects, Trojan horses, malware or any other items that may introduce security vulnerabilities to or harm user devices, apps or personal data.
    - Apps that collect information (such as the user's location or behaviour) without the user's knowledge (spyware) are prohibited.
    - Malicious scripts and password phishing scams are also prohibited on Google Play, as are apps that cause users to unknowingly download or install apps from sources outside of Google Play.
    - An app downloaded from Google Play may not modify, replace or update its own APK binary code using any method other than Google Play's update mechanism.
- **System Interference:**
    - An app downloaded from Google Play (or its components or derivative elements) must not make changes to the user’s device outside of the app without the user’s knowledge and consent.
    - This includes behaviour such as replacing or reordering the default presentation of apps, widgets or the settings on the device. If an app makes such changes with the user’s knowledge and consent, it must be clear to the user which app has made the change and the user must be able to reverse the change easily, or by uninstalling the app altogether.
    - Apps and their ads must not modify or add browser settings or bookmarks, add homescreen shortcuts, or icons on the user’s device as a service to third parties or for advertising purposes.
    - Apps and their ads must not display advertisements through system level notifications on the user’s device, unless the notifications derive from an integral feature provided by the installed app (e.g. an airline app that notifies users of special deals, or a game that notifies users of in-game promotions).
    - Apps must not encourage, incentivise or mislead users into removing or disabling third-party apps except as part of a security service provided by the app.

Network Usage and Terms

Apps must not create unpredictable network usage that has an adverse impact on a user's service charges or an Authorised Operator's network. Apps also may not knowingly violate an Authorised Operator's terms of service for allowed usage or any Google terms of service.

Spam and Placement in the Store

Developers are important partners in maintaining a great user experience on Google Play.

- Do not post repetitive content.
- Do not use irrelevant, misleading or excessive keywords in apps descriptions, titles or metadata.
- Developers must not attempt to change the placement of any Product in the Store, or manipulate any product ratings or reviews by unauthorised means such as fraudulent installs, paid or fake reviews or ratings, or by offering incentives to rate products.
- Apps that are created by an automated tool or wizard service must not be submitted to Google Play by the operator of that service on behalf of other persons.
- Do not post an app where the primary functionality is to:
    - Drive affiliate traffic to a website or
    - Provide a web view of a website not owned or administered by you (unless you have permission from the website owner/administrator to do so)
- Do not send SMS, email or other messages on behalf of the user without providing the user with the ability to confirm content and intended recipient.

App Promotion

Apps published on Google Play may not directly or indirectly engage in or benefit from the following behaviour:

- Promotion via deceptive ads on websites, apps or other properties, including simulated system, service or app notifications or alerts.
- Promotion or install tactics which cause redirection to Google Play or the download of the app without informed user action.
- Unsolicited promotion via SMS services.

It is your responsibility to ensure that no ad network or affiliate uses such methods to direct users to pages that make your app available for download.

Paid and Free Apps

In-store purchases**:** Developers charging for apps and downloads from Google Play must use Google Play's payment system.

In-app purchases:

- Developers offering products within a game downloaded from Google Play or providing access to game content must use Google Play In-app Billing as the method of payment.
- Developers offering products within another category of app downloaded from Google Play must use Google Play In-app Billing as the method of payment, except:
    - where payment is solely for physical products; or
    - where payment is for digital content that may be consumed outside of the app itself (e.g. buying songs that can be played on other music players).

A list of examples describing appropriate use cases for Google Play In-app Billing is available in the Google Play Developer Help Centre.

- In-app virtual currencies must only be used within the app where they were first purchased.

- Developers must not mislead users about the apps that they are selling nor about any in-app services, goods, content or functionality that they are selling. If your product description on Google Play refers to in-app features to which a specific or additional charge applies, your description must clearly notify users that payment is required to access those features.

Subscriptions and Cancellations

Google's subscription cancellation policy is that a user will not receive a refund for the current billing period when cancelling a subscription, but will continue to receive issues and updates of the relevant subscription content (if any) for the remainder of the billing period, regardless of the cancellation.

You (as the content or access provider) may implement a more flexible refund policy with your users directly, and it is your responsibility to notify your users of those policies and to ensure that the policies comply with applicable law.

Ad Policy

The policy below covers all ads that are implemented in, and bundled, with apps. These rules are important in maintaining a positive experience for everyone using Android apps from Google Play. Make sure that you check back from time to time, as these policies may change.

1. **Developer Terms apply to the entire user experience of your app**
2. Please be aware that Google's Developer Distribution Agreement and Developer Programme Policies (together, "Developer Terms") apply to each app as well as any ads or third-party libraries bundled or made available through the app. Offer your users a consistent, policy-compliant and well-communicated user experience.
3. Ads are considered part of your app for purposes of content review and compliance with the Developer Terms. Therefore, all of the policies referrred to above also apply. Please take care to use advertising which does not violate the Developer Terms.
4. Ads which are inconsistent with the app's content rating also violate our Developer Terms.
5. **Ads Context**
6. Ads must not simulate or impersonate the user interface of any app, or notification and warning elements of an operating system. It must be clear to the user which app each ad is associated with or implemented in.
7. **Ad Walls and Interstitial Ads**
8. Interstitial ads may only be displayed inside of the app with which they came. Forcing the user to click on ads or submit personal information for advertising purposes in order to fully use an app is prohibited. A prominent and accessible target must be made available to users in any interstitial ad so that they may dismiss the ad without penalty or inadvertent click-through.
9. **Interfering with Apps and Third-party Ads**
10. Ads associated with your app must not interfere with other apps or their ads.
11. **Usage of Android advertising ID**
12. Google Play Services version 4.0 introduced new APIs and an ID for use by advertising and analytics providers. Terms for the use of this ID are below.

- **Usage**. The Android advertising identifier must only be used for advertising and user analytics. The status of the "Opt-out of Interest-based Advertising" setting must be verified on each access of the ID.
- **Association with personally identifiable information or other identifiers**. The advertising identifier must not be connected to personally identifiable information or associated with any persistent device identifier (for example: SSAID, MAC address, IMEI, etc.) without the explicit consent of the user.

- **Respecting users' selections**. Upon reset, a new advertising identifier must not be connected to a previous advertising identifier or data derived from a previous advertising identifier without the explicit consent of the user. Furthermore, you must abide by a user's "Opt out of Interest-based Advertising" setting. If a user has enabled this setting, you may not use the advertising identifier for creating user profiles for advertising purposes or for targeting users with interest-based advertising. Allowed activities include contextual advertising, frequency capping, conversion tracking, reporting and security and fraud detection.
- **Transparency to users**. The collection and use of the advertising identifier and commitment to these terms must be disclosed to users in a legally adequate privacy notification.
- **Abiding by the terms of use**. The advertising identifier may only be used in accordance with these terms, including by any party that you may share it with in the course of your business. Beginning 1 August 2014, all updates and new apps uploaded to the Play Store must use the advertising ID (when available on a device) in lieu of any other device identifiers for any advertising purposes.

Policy Enforcement

In the event that your app is removed from Google Play, you will receive an email notification to that effect. If you have any questions or concerns regarding a removal or a rating/comment from a user, you may contact us at http://support.google.com/googleplay/android-developer. Serious or repeated violations of the Developer Distribution Agreement or this Content Policy will result in account termination. Repeated infringement of intellectual property rights, including copyright, will also result in account termination. For more information on Google's copyright policies, please see here.

"The approval process also makes it easier for developers to check on the status of their apps and receive more detail about why their apps were not approved through Google's Developer Console."
Google Launches App Approval Process: What Android Developers Must Know

And that is here: Upload & distribute apps

Apparently the review process is faster, and why we learn from App Submissions On Google Play Now Reviewed By Staff, Will Include Age-Based Ratings
"The reason why Google's app review team is able to process app submissions so quickly is because the system also includes an automated element. Before app reviewers are presented with the applications, Google uses software to pre-analyze the app for things like viruses and malware as well as other content violations. For example, its image analysis systems are capable of automatically detecting apps that include sexual content, as well as those that infringe on other applications' copyright."

A,B,C,D,E,F, ok... Google provides support for conformance with these guidelines in the form of a Launch Checklist , which is quite comprehensive and is worth reading It starts with detailing the Publishing Process, Understand the Publishing Process and along with that how to Preparing for Release, and steps presented there and if not included in the development process, it will benefit from checking against And I will itemize them here:

Preparing for Release

Gathering Materials and Resources

To begin preparing your application for release you need to gather several supporting items. At a minimum this includes cryptographic keys for signing your application and an application icon. You might also want to include an end-user license agreement.

Cryptographic keys

The Android system requires that each installed application be digitally signed with a certificate that is owned by the application's developer (that is, a certificate for which the developer holds the private key). The Android system uses the certificate as a means of identifying the author of an application and establishing trust relationships between applications. The certificate that you use for signing does not need to be signed by a certificate authority; the Android system allows you to sign your applications with a self-signed certificate. To learn about certificate requirements, see Signing Your Applications.

Important: Your application must be signed with a cryptographic key whose validity period ends after 22 October 2033.
You may also have to obtain other release keys if your application accesses a service or uses a third-party library that requires you to use a key that is based on your private key. For example, if your application uses the MapView class, which is part of the Google Maps external library, you will need to register your application with the Google Maps service and obtain a Maps API key. For information about getting a Maps API key, see Obtaining a Maps API key.

Application Icon

Be sure you have an application icon and that it meets the recommended icon guidelines. Your application's icon helps users identify your application on a device's Home screen and in the Launcher window. It also appears in Manage Applications, My Downloads, and elsewhere. In addition, publishing services such as Google Play display your icon to users.

Note: If you are releasing your application on Google Play, you need to create a high resolution version of your icon. See Graphic Assets for your Application for more information.

End-user License Agreement

Consider preparing an End User License Agreement (EULA) for your application. A EULA can help protect your person, organization, and intellectual property, and we recommend that you provide one with your application.

Miscellaneous Materials

You might also have to prepare promotional and marketing materials to publicize your application. For example, if you are releasing your application on Google Play you will need to prepare some promotional text and you will need to create screenshots of your application. For more information, see Graphic Assets for your Application

Configuring Your Application for Release

After you gather all of your supporting materials you can start configuring your application for release. This section provides a summary of the configuration changes we recommend that you make to your source code, resource files, and application manifest prior to releasing your application. Although most of the configuration changes listed in this section are optional, they are considered good coding practices and we encourage you to implement them. In some cases, you may have already made these configuration changes as part of your development process.

Choose a good package name
Make sure you choose a package name that is suitable over the life of your application. You cannot change the package name after you distribute your application to users. You can set the package name in application's manifest file. For more information, see the package attribute documentation.

Turn off logging and debugging
Make sure you deactivate logging and disable the debugging option before you build your application for release. You can deactivate logging by removing calls to Log methods in your source files. You can disable debugging by removing theandroid:debuggable attribute from the <application> tag in your manifest file, or by setting the android:debuggableattribute to false in your manifest file. Also, remove any log files or static test files that were created in your project.
Also, you should remove all Debug tracing calls that you added to your code, such as startMethodTracing() andstopMethodTracing() method calls.
Important: Ensure that you disable debugging for your app if using WebView to display paid for content or if using JavaScript interfaces, since debugging allows users to inject scripts and extract content using Chrome DevTools. To disable debugging, use the WebView.setWebContentsDebuggingEnabled() method.

Clean up your project directories
Clean up your project and make sure it conforms to the directory structure described in Android Projects. Leaving stray or orphaned files in your project can prevent your application from compiling and cause your application to behave unpredictably. At a minimum you should do the following cleanup tasks:

- Review the contents of your jni/, lib/, and src/ directories. The jni/ directory should contain only source files associated with the Android NDK, such as .c, .cpp, .h, and .mk files. The lib/ directory should contain only third-party library files or private library files, including prebuilt shared and static libraries (for example, .so files). Thesrc/ directory should contain only the source files for your application (.java and .aidl files). The src/ directory should not contain any .jar files.
- Check your project for private or proprietary data files that your application does not use and remove them. For example, look in your project's res/ directory for old drawable files, layout files, and values files that you are no longer using and delete them.
- Check your lib/ directory for test libraries and remove them if they are no longer being used by your application.
- Review the contents of your assets/ directory and your res/raw/ directory for raw asset files and static files that you need to update or remove prior to release.

Review and update your manifest and Gradle build settings
Verify that the following manifest and build files items are set correctly:

- <uses-permission> element

  You should specify only those permissions that are relevant and required for your application.
- android:icon and android:label attributes

  You must specify values for these attributes, which are located in the <application> element.
- android:versionCode and android:versionName attributes.

  We recommend that you specify values for these attributes, which are located in the <manifest> element. For more information see Versioning your Application.

There are several additional manifest or build file elements that you can set if you are releasing your application on Google Play. For example, the android:minSdkVersion and android:targetSdkVersion attributes, which are located in the<uses-sdk> element. For more information about these and other Google Play settings, see Filters on Google Play.

Address compatibility issues

Android provides several tools and techniques to make your application compatible with a wide range of devices. To make your application available to the largest number of users, consider doing the following:

- **Add support for multiple screen configurations**

  Make sure you meet the best practices for supporting multiple screens. By supporting multiple screen configurations you can create an application that functions properly and looks good on any of the screen sizes supported by Android.

- **Optimize your application for Android tablet devices.**

  If your application is designed for devices older than Android 3.0, make it compatible with Android 3.0 devices by following the guidelines and best practices described in Optimizing Apps for Android 3.0 .

- **Consider using the Support Library**

  If your application is designed for devices running Android 3.x, make your application compatible with older versions of Android by adding the Support Library to your application project. The Support Library provides static support libraries that you can add to your Android application, which enables you to use APIs that are either not available on older platform versions or use utility APIs that are not part of the framework APIs.

Update URLs for servers and services
If your application accesses remote servers or services, make sure you are using the production URL or path for the server or service and not a test URL or path.

Implement Licensing (if you are releasing on Google Play)
If you are releasing a paid application through Google Play, consider adding support for Google Play Licensing. Licensing lets you control access to your application based on whether the current user has purchased it. Using Google Play Licensing is optional even if you are releasing your app through Google Play.
For more information about Google Play Licensing Service and how to use it in your application, see Application Licensing.

Building Your Application for Release

After you finish configuring your application you can build it into a release-ready `.apk` file that is signed and optimized. The JDK includes the tools for signing the `.apk` file (Keytool and Jarsigner); the Android SDK includes the tools for compiling and optimizing the `.apk` file. If you are using Android Studio or you are using the Gradle build system from the command line, you can automate the entire build process. For more information about configuring Gradle builds, see Configuring Gradle Builds.

Building with Android Studio

You can use the Gradle build system, integrated with Android Studio to build a release-ready `.apk` file that is signed with your private key and optimized. To learn how to setup and run builds from Android Studio, see Building and Running from Android Studio.

The build process assumes that you have a certificate and private key suitable for signing your application. If you do not have a suitable certificate and private key, Android Studio can help you generate one. For more information about the signing process, see Signing Your Applications.

Preparing External Servers and Resources

If your application relies on a remote server, make sure the server is secure and that it is configured for production use. This is particularly important if you are implementing in-app billing in your application and you are performing the signature verification step on a remote server.

Also, if your application fetches content from a remote server or a real-time service (such as a content feed), be sure the content you are providing is up to date and production-ready.

Testing Your Application for Release

Testing the release version of your application helps ensure that your application runs properly under realistic device and network conditions. Ideally, you should test your application on at least one handset-sized device and one tablet-sized device to verify that your user interface elements are sized correctly and that your application's performance and battery efficiency are acceptable.

As a starting point for testing, see What to Test. This article provides a summary of common Android situations that you should consider when you are testing. When you are done testing and you are satisfied that the release version of your application behaves correctly, you can release your application to users. For more information, see Releasing Your Application to Users. If you are publishing your application on Google Play, see Launch Checklist for Google Play.

The steps include links with information about application icon and graphic assets, app signing and I will enumerate them here, due to their importance in having the app approved

http://developer.android.com/guide/topics/manifest/manifest-element.html#package
http://android-developers.blogspot.ca/2011/10/android-market-featured-image.html
https://support.google.com/googleplay/android-developer/answer/1078870
http://developer.android.com/tools/publishing/app-signing.html#studio
http://developer.android.com/tools/testing/what_to_test.html

Among the tools provided to help and sustain the development
http://developer.android.com/tools/help/emulator.html , with valuable info on how to configure various features Graphics Acceleration, Virtual Machine acceleration, SD card emulation
http://developer.android.com/distribute/tools/promote/device-art.html
http://developer.android.com/tools/help/emulator.html#netspeed

Localization Checklist

Another important stage in preparing for release is app localization, and I will highlight the guidelines provided by google in Localization Checklist

1. Identify target languages and locales
2. Design for localization
3. Manage strings for localization
4. Translate UI strings and other resources
5. Test your localized app
6. Prepare for international launch
7. Support international users after launch

These checklists are a good checkpoint before release, because the road to release is development, and development that delivers quality. And, once again google provides directions

Core App Quality

Tablet App Quality

Wear App Quality

TV App Quality

Auto App Quality

With Core App Quality extending the following topics

Visual Design and User Interaction

These criteria ensure that your app provides standard Android visual design and interaction patterns where appropriate, for a consistent and intuitive user experience.

Area	ID	Description	Tests
Standard design	UX-B1	App follows Android Design guidelines and uses common UI patterns and icons: a. App does not redefine the expected function of a system icon (such as the Back button). b. App does not replace a system icon with a completely different icon if it triggers the standard UI behavior. c. If the app provides a customized version of a standard system icon, the icon strongly resembles the system icon and triggers the standard system behavior. d. App does not redefine or misuse Android UI patterns, such that icons or behaviors could be misleading or confusing to users.	CR-all
Navigation	UX-N1	App supports standard system Back button navigation and does not make use of any custom, on-screen "Back button" prompts.	CR-3
	UX-N2	All dialogs are dismissable using the Back button.	CR-3
	UX-N3	Pressing the Home button at any point navigates to the Home screen of the device.	CR-1
Notifications	UX-S1	Notifications follow Android Design guidelines. In particular:	CR-11

		a. Multiple notifications are stacked into a single notification object, where possible. b. Notifications are persistent only if related to ongoing events (such as music playback or a phone call). c. Notifications do not contain advertising or content unrelated to the core function of the app, unless the user has opted in.	
	UX-S2	App uses notifications only to: a. Indicate a change in context relating to the user personally (such as an incoming message), or b. Expose information/controls relating to an ongoing event (such as music playback or a phone call).	CR-11

Other documentation resources of interest being:

Design
Up and running with material design
Material Design -- Animation
Material Design -- Style
Material Design -- Layout
Material Design -- Components
Material Design -- Patterns
Material Design -- Usability

(Another approach is flat design, The future of flat design, but will be discussed in future meetups)

Navigation with Back and Up

Navigation Within Your App
Navigation into Your App via Home Screen Widgets and Notifications
Navigation Between Apps

Action Bar

Material Design
General Organization
Adapting to Rotation and Different Screen Sizes
Layout Considerations for Split Action Bars
Action Buttons
Contextual Action Bars

Action Bar Checklist

Notifications

Anatomy of a Notification -- layouts, actions
Heads-up Notification
Guidelines

Functionality

These criteria ensure that your app provides expected functional behavior, with the appropriate level of permissions.

Area	ID	Description	Tests
Permissions	FN-P1	App requests only the *absolute minimum* permissions that it needs to support core functionality.	CR-11
	FN-P2	App does not request permissions to access sensitive data (such as Contacts or the System Log) or services that can cost the user money (such as the Dialer or SMS), unless related to a core capability of the app.	
Install location	FN-L1	App functions normally when installed on SD card (if supported by app). Supporting installation to SD card is recommended for most large apps (10MB+). See the App Install Location developer guide for information about which types of apps should support installation to SD card.	SD-1
Audio	FN-A1	Audio does not play when the screen is off, unless this is a core feature (for example, the app is a music player).	CR-7
	FN-A2	Audio does not play behind the lock screen, unless this is a core feature.	CR-8
	FN-A3	Audio does not play on the home screen or over another app, unless this is a core feature.	CR-1, CR-2
	FN-A4	Audio resumes when the app returns to the foreground, or indicates to the user that playback is in a paused state.	CR-1, CR-8
UI and Graphics	FN-U1	App supports both landscape and portrait orientations (if possible). Orientations expose largely the same features and actions and preserve functional parity. Minor changes in content or views are acceptable.	CR-5

	FN-U2	App uses the whole screen in both orientations and does not letterbox to account for orientation changes. Minor letterboxing to compensate for small variations in screen geometry is acceptable.	CR-5
	FN-U3	App correctly handles rapid transitions between display orientations without rendering problems.	CR-5
User/app state	FN-S1	App should not leave any services running when the app is in the background, unless related to a core capability of the app. For example, the app should not leave services running to maintain a network connection for notifications, to maintain a Bluetooth connection, or to keep the GPS powered-on.	CR-6
	FN-S2	App correctly preserves and restores user or app state. App preserves user or app state when leaving the foreground and prevents accidental data loss due to back-navigation and other state changes. When returning to the foreground, the app must restore the preserved state and any significant stateful transaction that was pending, such as changes to editable fields, game progress, menus, videos, and other sections of the app or game. a. When the app is resumed from the Recents app switcher, the app returns the user to the exact state in which it was last used. b. When the app is resumed after the device wakes from sleep (locked) state, the app returns the user to the exact state in which it was last used. c. When the app is relaunched from Home or All Apps, the app restores the app state as closely as possible to the previous state. d. On Back keypresses, the app gives the user the option of saving any app or user state that would otherwise be lost on back-navigation.	CR-1, CR-3, CR-5

Performance and Stability

These criteria ensure that apps provide the performance, stability, and responsiveness expected by users.

Area	ID	Description	Tests
Stability	PS-S1	App does not crash, force close, freeze, or otherwise function abnormally on any targeted device.	CR-all, SD-1,HA-1

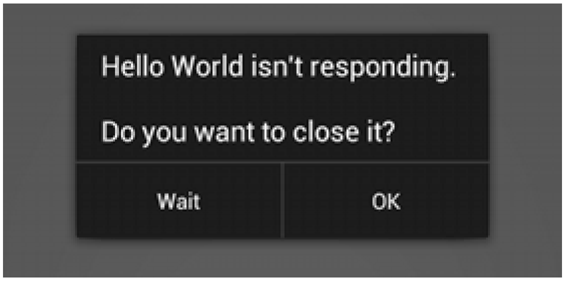
Performance	PS-P1	App loads quickly or provides onscreen feedback to the user (a progress indicator or similar cue) if the app takes longer than two seconds to load.	CR-all, SD-1
	PS-P2	With StrictMode enabled (see StrictMode Testing, below), no red flashes (performance warnings from StrictMode) are visible when exercising the app, including during game play, animations and UI transitions, and any other part of the app.	PM-1
Media	PS-M1	Music and video playback is smooth, without crackle, stutter, or other artifacts, during normal app usage and load.	CR-all, SD-1,H A-1
Visual quality	PS-V1	App displays graphics, text, images, and other UI elements without noticeable distortion, blurring, or pixelation. a. App provides high-quality graphics for all targeted screen sizes and form factors, including for larger-screen devices such as tablets. b. No aliasing at the edges of menus, buttons, and other UI elements is visible.	CR-all
	PS-V2	App displays text and text blocks in an acceptable manner. a. Composition is acceptable in all supported form factors, including for	

StrictMode API for Built-In Performance Monitoring

and the famous Google 20% time”: contradicted somehow by Marissa Mayer
The 'Dirty Little Secret' About Google's 20% Time, According To Marissa Mayer

Keeping Your App Responsive

Because “It's possible to write code that wins every performance test in the world, but still feels sluggish, hang or freeze for significant periods, or take too long to process input. The worst thing that can happen to your app's responsiveness is an "Application Not Responding" (ANR) dialog.”



Hello World isn't responding.

Do you want to close it?

Multithreading For Performance

And criteria that determines app readiness

Google Play

These criteria ensure that your apps are ready to publish on Google Play.

Area	ID	Description	Tests
Policies	GP-P1	App strictly adheres to the terms of the Google Play Developer Content Policy and does not offer inappropriate content, does not use intellectual property or brand of others, and so on.	GP-all
	GP-P2	App maturity level is set appropriately, based on the Content Rating Guidelines. Especially, note that apps that request permission to use the device location cannot be given the maturity level "Everyone".	GP-1
App Details Page	GP-D1	App feature graphic follows the guidelines outlined in this blog post. Make sure that: a. The app listing includes a high-quality feature graphic. b. The feature graphic does not contain device images, screenshots, or small text that will be illegible when scaled down and displayed on the smallest screen size that your app is targeting. c. The feature graphic does not resemble an advertisement.	GP-1, GP-2
	GP-D2	App screenshots and videos do not show or reference non-Android devices.	GP-1
	GP-D3	App screenshots or videos do not represent the content and experience of your app in a misleading way.	
User Support	GP-X1	Common user-reported bugs in the Reviews tab of the Google Play page are addressed if they are reproducible and occur on many different devices. If a bug occurs on only a few devices, you should still address it if those devices are particularly popular or new.	GP-1

Setting Up a Test Environment

To assess the quality of your app, you need to set up a suitable hardware or emulator environment for testing.

The ideal test environment would include a small number of actual hardware devices that represent key form factors and hardware/software combinations currently available to consumers. It's not necessary to test on *every* device that's on the market — rather, you should focus on a small number of representative devices, even using one or two devices per form factor.

If you are not able to obtain actual hardware devices for testing, you should set up emulated devices (AVDs) to represent the most common form factors and hardware/software combinations.

To go beyond basic testing, you can add more devices, more form factors, or new hardware/software combinations to your test environment. You can also increase the number or complexity of tests and quality criteria.

Test Procedures

These test procedures help you discover various types of quality issues in your app. You can combine the tests or integrate groups of tests together in your own test plans. See the sections above for references that associate specific criteria with specific tests.

Type	Test	Description
Core Suite	CR-0	Navigate to all parts of the app — all screens, dialogs, settings, and all user flows. a. If the application allows for editing or content creation, game play, or media playback, make sure to enter those flows to create or modify content. b. While exercising the app, introduce transient changes in network connectivity, battery function, GPS or location availability, system load, and so on.
	CR-1	From each app screen, press the device's Home key, then re-launch the app from the All Apps screen.
	CR-2	From each app screen, switch to another running app and then return to the app under test using the Recents app switcher.
	CR-3	From each app screen (and dialogs), press the Back button.
	CR-5	From each app screen, rotate the device between landscape and portrait orientation at least three times.
	CR-6	Switch to another app to send the test app into the background. Go to Settings and check whether the test app has any services running while in the background. In Android 4.0 and higher, go to the Apps screen and find the app in the "Running" tab. In earlier versions, use "Manage Applications" to check for running services.
	CR-7	Press the power button to put the device to sleep, then press the power button again to awaken the screen.
	CR-8	Set the device to lock when the power button is pressed. Press the power button to put the device to sleep, then press the power button again to awaken the screen, then unlock the device.
	CR-9	For devices that have slide-out keyboards, slide the keyboard in and out at least once. For devices that have keyboard docks, attach the device to the keyboard dock.

	CR-10	For devices that have an external display port, plug-in the external display.
	CR-11	Trigger and observe in the notications drawer all types of notifications that the app can display. Expand notifications where applicable (Android 4.1 and higher), and tap all actions offered.
	CR-12	Examine the permissions requested by the app by going to Settings > App Info.
Install on SD Card	SD-1	Repeat *Core Suite* with app installed to device SD card (if supported by app). To move the app to SD card, you can use Settings > App Info > Move to SD Card.
Hardware acceleration	HA-1	Repeat *Core Suite* with hardware acceleration enabled. To force-enable hardware acceleration (where supported by device), add `hardware-accelerated="true"` to the `<application>` in the app manifest and recompile.
Performance Monitoring	PM-1	Repeat *Core Suite* with StrictMode profiling enabled as described below. Pay close attention to garbage collection and its impact on the user experience.
Google Play	GP-1	Sign into the Developer Console to review your developer profile, app description, screenshots, feature graphic, maturity settings, and user feedback.
	GP-2	Download your feature graphic and screenshots and scale them down to match the display sizes on the devices and form factors you are targeting.
	GP-3	Review all graphical assets, media, text, code libraries, and other content packaged in the app or expansion file download.
Payments	GP-4	Navigate to all screens of your app and enter all in-app purchase flows.

Testing with StrictMode

For performance testing, we recommend enabling StrictMode in your app and using it to catch operations on the main thread and other threads that could affect performance, network accesses, file reads/writes, and so on.

You can set up a monitoring policy per thread using StrictMode.ThreadPolicy.Builder and enable all supported monitoring in the ThreadPolicy using detectAll().

Make sure to enable **visual notification** of policy violations for the ThreadPolicy using penaltyFlashScreen().

Tablet App Quality

1. *Test for Basic Tablet App Quality*
2. *Optimize Layouts for Larger Screens*
3. *Take Advantage of Extra Screen Area*
4. *Use Assets Designed for Tablet Screens*

Table 1*. Raw asset sizes for icon types.*

Density	*Launcher*	*Action Bar*	*Small/Contextual*	*Notification*
mdpi	*48x48 px*	*32x32 px*	*16x16 px*	*24x24 px*
hdpi	*72x72 px*	*48x48 px*	*24x24 px*	*36x36 px*
tvdpi	*(use hdpi)*	*(use hdpi)*	*(use hdpi)*	*(use hdpi)*
xhdpi	*96x96 px*	*64x64 px*	*32x32 px*	*48x48 px*
xxhdpi	*144x144 px*	*96x96 px*	*48x48 px*	*72x72 px*

5. *Adjust Font Sizes and Touch Targets*
6. *Adjust Sizes of Home Screen Widgets*
7. *Full Feature Set for Tablet Users*
8. *Target Android Versions Properly*
9. *Declare Hardware Feature Dependencies Properly*
10. *Declare Support for Tablet Screens*
11. *Showcase Your Tablet UI in Google Play*
12. *Follow Best Practices for Publishing in Google Play regarding app optimization, app filtering (not filter out for tablets), distribute as single apk*

And the corresponding tests

Setting Up a Test Environment for Tablets

Assess the quality of your app on tablets — both for core app quality and tablet app quality — with a suitable hardware or emulator environment for testing.

Compared to the recommended test environment for testing against the core app quality criteria, include mid-size tablets and tablets with more or fewer hardware/software features.

Table 1. A typical tablet test environment might include one or two devices from each row in the table below, with one of the listed platform versions, screen configurations, and hardware feature configurations.

Type	Size	Density	Version	AVD Skin
7-inch tablet	large or -sw600	hdpi, tvdpi	Android 4.0+ (API level 14 and higher)	WXGA800-7in
10-inch tablet	xlarge or -sw800	mdpi, hdpi, xhdpi	Android 3.2+ (API level 13 and higher)	WXGA800

Wear App Quality

Functionality

These criteria ensure that your app is configured correctly and provides the expected functional behavior.

Type	ID	Description
General	WR-GL	Handheld app includes either notifications with wearable-specific functionality or a wearable app that runs directly on the Wear device. (Learn how)

	WR-VF	App has Wear functionality that is visible to the user.
	WR-BF	Wear functionality works as expected or as described in the app's Google Play Store listing.
Packaging	WR-PK	Wearable apps that are dependent on a handheld app for functionality are packaged inside that handheld app. (Learn how)
Notifications	WR-FW	Notifications with wearable-specific functionality use a `RemoteInput` or`WearableExtender`. (Learn how)
	WR-FR	Notifications for messaging apps allow users to reply via voice input or quick responses. (Learn how)
	WR-FG	Similar notifications are grouped together in a stack. (Learn how)
Gestures	WR-GP	Full-screen activities use long press for the sole purpose of prompting to quit. (Learn how)
	WR-GD	If the app disables the swipe-to-dismiss gesture in a full-screen activity, it responds to the long-press-to-dismiss gesture in that activity. (Learn how)
Watch Face	WR-WF	Apps that include watch faces use the official Watch Face API. (Learn how)

Visual Design and User Interaction

These criteria ensure that your app follows critical design and interaction patterns to provide a consistent, intuitive, and enjoyable user experience on wearables.

Type	ID	Description
Layout	WR-LS	App user interface is formatted appropriately for square displays. App content fits within the physical display area and no text or controls are cut off by the screen edges. (Learn how)
	WR-LR	App user interface is formatted appropriately for round displays including devices with an inset (or "chin") at the bottom of the screen. App content

		fits within the physical display area and no text or controls are cut off by the screen edges. (Learn how)
	WR-TC	App text is large and glanceable with a suggested minimum size of 16sp. (Learn how)
Launcher	WR-LN	App launcher string is the app name, not a command phrase. (Learn how)
Notifications	WR-NC	App displays confirmation animations when appropriate. (Learn how)
	WR-NR	Notification cards have the app icon visible at the top right edge. The one exception is if the notification card has single-action controls, for example a media playback card. (Learn how)
	WR-WI	Notification actions have a white icon, action title, and transparent background. (Learn how)
	WR-PB	Notification photo backgrounds are used only to convey information, not to brand a card. (Learn how)
	WR-PR	Notification photo backgrounds have a resolution of at least 400x400. (Learn how)
Google Play	WR-GS	App includes at least one Wear screenshot in its Play Store Listing. (Learn how)

TV App Quality

Visual Design and User Interaction

These criteria ensure that your app follows critical design and interaction patterns to ensure a consistent, intuitive, and enjoyable user experience on TV devices.

Type	Test	Description
Launcher	TV-LM	App displays a launcher icon in the Android TV Launcher after installation. (Learn how)

	TV-LB	App displays a 320px x 180px full-size banner as its launcher icon in the Android TV Launcher. (Learn how)
	TV-BN	App launch banner contains the name of the app. (Learn how)
	TV-LG	If the app is a game, it appears in the Games row in the Android TV Launcher. (Learn how)
Layout	TV-LO	All app interfaces are presented in landscape orientation. (Learn how)
	TV-TC	App displays core text at 16sp or higher in size. (Learn how)
	TV-TA	App displays all text at 12sp or higher in size. (Learn how)
	TV-OV	App does not display any text or functionality that is partially cut off by the edges of the screen. (Learn how)
	TV-TR	App does not partially obscure other apps. App fills the entire screen and has a non-transparent background.
Navigation	TV-DP	App functionality is navigable using 5-way D-pad controls, unless the app requires a gamepad controller as specified in TV-GP. (Learn how)
	TV-DK	If the app requires a game controller, as specified in TV-GP, all functionality is navigable using standard Android game controller keys. (Learn how)
	TV-DM	App does not depend on a remote controller having a Menu button to access user interface controls. (Learn how)

Functionality

These criteria ensure that your app is configured correctly and provides expected functional behavior.

Type	Test	Description

Manifest	TV-ML	App manifest sets an intent type of `ACTION_MAIN` with category`CATEGORY_LEANBACK_LAUNCHER`. (Learn how)
	TV-MT	App manifest sets the hardware feature `android.hardware.touchscreen` to not required. (Learn how)
Game Controllers	TV-GP	If the app uses a game controller as it's primary input method, it declares the appropriate requirement with the `<uses-feature>` manifest tag. (Learn how)
	TV-GC	If the app provides visual instructions for using game controllers, the instructions should be free of branding and show a compatible button layout. (Learn how)
Advertising	TV-AP	App allows interaction with advertising using D-pad controls. (Learn how)
	TV-AD	For advertising that uses fullscreen, non-video ads, the app allows the user to immediately dismiss the ad with D-pad controls.
	TV-AU	For advertising that uses clickable, non-fullscreen, non-video ads, the app does not allow ads to link to a web URL.
	TV-AA	For advertising that uses clickable, non-fullscreen, non-video ads, the app does not allow ads to link to another app that is not available on TV devices.
Web Content	TV-WB	For web content, the app uses `WebView` components and does not attempt to launch a web browser app.
Media Playback	TV-NP	If the app continues to play sound or video after the user has left, the app provides a *Now Playing* card on the home screen recommendation row so users can return to the app to control playback. (Learn how)
	TV-PA	If the app provides a *Now Playing* card, selecting this card takes the user to a screen that allows playback to be paused. (Learn how)

	TV-PP	If the app plays video or music content, the app toggles between play and pause of media playback when a play or pause key event is sent during playback. (Learn how)

Auto App Quality (Android Auto)

Visual Design and User Interaction

These criteria ensure that your app follows critical design and interaction patterns to ensure a consistent, intuitive, and enjoyable user experience on TV devices.

Type	Test	Description
Launcher	TV-LM	App displays a launcher icon in the Android TV Launcher after installation. (Learn how)
	TV-LB	App displays a 320px x 180px full-size banner as its launcher icon in the Android TV Launcher. (Learn how)
	TV-BN	App launch banner contains the name of the app. (Learn how)
	TV-LG	If the app is a game, it appears in the Games row in the Android TV Launcher. (Learn how)
Layout	TV-LO	All app interfaces are presented in landscape orientation. (Learn how)
	TV-TC	App displays core text at 16sp or higher in size. (Learn how)
	TV-TA	App displays all text at 12sp or higher in size. (Learn how)
	TV-OV	App does not display any text or functionality that is partially cut off by the edges of the screen. (Learn how)
	TV-TR	App does not partially obscure other apps. App fills the entire screen and has a non-transparent background.
Navigation	TV-DP	App functionality is navigable using 5-way D-pad controls, unless the app requires a gamepad controller as specified in TV-GP. (Learn how)

	TV-DK	If the app requires a game controller, as specified in TV-GP, all functionality is navigable using standard Android game controller keys. (Learn how)
	TV-DM	App does not depend on a remote controller having a Menu button to access user interface controls. (Learn how)

Google provide various tool for development and test

Tools

SDK Tools

The SDK tools are installed with the SDK starter package and are periodically updated. The SDK tools are required if you are developing Android applications. The most important SDK tools include the Android SDK Manager (android sdk), the AVD Manager (android avd) the emulator (emulator), and the Dalvik Debug Monitor Server (ddms). A short summary of some frequently-used SDK tools is provided below.

Virtual Device Tools

Android Virtual Device Manager

The AVD Manager provides a graphical user interface in which you can create and manage Android Virtual Devices (AVDs) that run in the Android Emulator.

Android Emulator (emulator)

A QEMU-based device-emulation tool that you can use to debug and test your applications in an actual Android run-time environment.

mksdcard

Helps you create a disk image that you can use with the emulator, to simulate the presence of an external storage card (such as an SD card).

Development Tools

android

Lets you manage AVDs, projects, and the installed components of the SDK.

Hierarchy Viewer (hierarchyviewer)

Provides a visual representation of the layout's View hierarchy with performance information for each node in the layout, and a magnified view of the display to closely examine the pixels in your layout.

lint

The Android lint tool is a static code analysis tool that checks your Android project source files for potential bugs and optimization improvements.

SDK Manager

Lets you manage SDK packages, such as installed platforms and system images.

sqlite3

Lets you access the SQLite data files created and used by Android applications.

Debugging Tools

adb

Android Debug Bridge (adb) is a versatile command line tool that lets you communicate with an emulator instance or connected Android-powered device. It also provides access to the device shell.

ADB Shell Commands

Learn the commands available for advanced command-line operations.

Dalvik Debug Monitor Server (ddms)

Lets you debug Android applications.

Device Monitor

Android Device Monitor is a stand-alone tool that provides a graphical user interface for several Android application debugging and analysis tools.

dmtracedump

Generates graphical call-stack diagrams from trace log files. The tool uses the Graphviz Dot utility to create the graphical output, so you need to install Graphviz before running `dmtracedump`. For more information on using`dmtracedump`, see Profiling with Traceview and dmtracedump

hprof-conv

Converts the HPROF file that is generated by the Android SDK tools to a standard format so you can view the file in a profiling tool of your choice.

Systrace

Lets you analyze the execution of your application in the context of system processes, to help diagnose display and performance issues.

traceview

Provides a graphical viewer for execution logs saved by your application.

Systrace

Lets you analyze the execution of your application in the context of system processes, to help diagnose display and performance issues.

Build Tools

JOBB

Allows you to build encrypted and unencrypted APK expansion files in Opaque Binary Blob (OBB) format.

ProGuard

Shrinks, optimizes, and obfuscates your code by removing unused code and renaming classes, fields, and methods with semantically obscure names.

zipalign

Optimizes .apk files by ensuring that all uncompressed data starts with a particular alignment relative to the start of the file. This should always be used to align .apk files after they have been signed.

Image Tools

Draw 9-patch

Allows you to easily create a NinePatch graphic using a WYSIWYG editor. It also previews stretched versions of the image, and highlights the area in which content is allowed.

etc1tool

A command line utility that lets you encode PNG images to the ETC1 compression standard and decode ETC1 compressed images back to PNG.

Tracer for OpenGL ES

Allows you to capture OpenGL ES commands and frame by frame images to help you understand how your graphics commands are being executed.

Platform Tools

The platform tools are typically updated every time you install a new SDK platform. Each update of the platform tools is backward compatible with older platforms. Usually, you directly use only one of the platform tools—the Android Debug Bridge (adb). Android Debug Bridge is a versatile tool that lets you manage the state of an emulator instance or Android-powered device. You can also use it to install an Android application (.apk) file on a device.

The other platform tools, such as aidl, aapt, dexdump, and dx, are typically called by the Android build tools or Android Development Tools (ADT), so you rarely need to invoke these tools directly. As a general rule, you should rely on the build tools or the ADT plugin to call them as needed.

Note: The Android SDK provides additional shell tools that can be accessed through adb, such as bmgr and logcat.

bmgr

A shell tool you can use to interact with the Backup Manager on Android devices supporting API Level 8 or greater.

logcat

Provides a mechanism for collecting and viewing system debug output.

Performance Tools

Rendering Analysis Tools

GPU Overdraw Debugger

What it's good for:

- Showing where an app might be doing more rendering work than necessary.
- Helping you see where you might be able to reduce rendering overhead.

Profiling GPU Rendering Walkthrough

What it's good for:

- Quickly seeing how a UI window performs against the 16-ms-per-frame target.
- Identifying whether any part of the rendering pipeline stands out in using processing time.
- Looking for spikes in frame rendering time associated with user or program actions.

Hierarchy Viewer

(The Hierarchy Viewer tool visualizes your app's view hierarchy and profiles the relative rendering speed for each view.)

What it's good for:

- Simplifying your view hierarchy to reduce overdraw, and make it easier to manage.
- Finding potential rendering performance bottlenecks related to the structure and shape of your view hierarchy.

Device Setup for Hierarchy Viewer

Profiling with Hierarchy Viewer

Memory Profilers

Memory Monitor
Heap Viewer
Allocation Tracker

Trace View

What it's good for:

- Tracking down performance problems in your source code.

Systrace

What it's good for:

- Evaluating container performance.
- Finding performance bottlenecks in the execution of your code.

Batterystats & Battery Historian

Testing Tools

I don’t think it’s necessary to reiterate the importance of testing for an application release, and the benefits resulted in customer experience, development time allocated for troubleshooting and fixing defects, and in case of critical ones time spent on releasing patches and fixes and ship them to the client.

Android Testing Tools

Testing is a critical software development activity because it helps you improve the quality of your apps, ensure better user satisfaction, and reduce overall development time spent on fixing defects.

The following sections describe tools that help you test your mobile apps for the Android platform.

Android Testing Support Library

This library provides a set of APIs that allow you to quickly build and run test code for your apps, including JUnit 4 and functional user interface (UI) tests. The Android Testing Support Library includes the following test automation tools:

- AndroidJUnitRunner: JUnit 4-compatible test runner for Android
- Espresso: UI testing framework; suitable for functional UI testing within an app
- UI Automator: UI testing framework; suitable for cross-app functional UI testing across system and installed apps

Monkey

This tool runs on your emulator or device and generates pseudo-random streams of user events such as clicks, touches, or gestures, as well as a number of system-level events. You can use the Monkey tool to stress-test applications that you are developing, in a random yet repeatable manner.

monkeyrunner

This testing system provides an API for writing programs that control an Android device or emulator from outside of Android code.

Best Practices for Testing

These classes and articles provide information about how to test your Android application.

Testing Your Activity

(http://smus.com/android-phonegap-plugins/ for Android intents)

How to test Activities in your Android applications.

Setting Up Your Test Environment
Creating and Running a Test Case
Testing UI Components
Creating Unit Tests
Creating Functional Tests

Automating UI Tests

How to automate your user interface tests for Android apps.

Testing UI for a Single App

Testing UI for Multiple Apps

Building Effective Unit Tests

How to build effective unit tests for Android apps.

Building Local Unit Tests

Building Instrumented Unit Tests

Windows Store

I will start with a note:

Important The App certification requirements for the Windows Store have been replaced by the Windows and Windows Phone Store Policies as of October 23, 2014. **Please refer to the** Windows and Windows Phone Store Policies **for all current requirements and policies related to submitting apps to the Windows Store (and Windows Phone Store).**

Make sure you refer to correct policies, which can be found here

Windows Store Policies

Document version: 6.6

Document date: September 8, 2015

Thank you for your interest in developing apps for Windows and Windows Phone. We're committed to a relationship with you that supports your ambitions and encourages a diverse catalog of high-quality, engaging apps for customers worldwide. We publish these policies with the dual goals of enabling developers and delighting our customers.

A few principles to get you started:

- **Offer real value with your app.** Provide a compelling reason to download your app from the Store.
- **Don't mislead our joint customers** about what your app can do, who is offering it, etc.
- **Don't attempt to cheat customers, the system or the ecosystem.** There is no place in our Store for any kind of fraud, be it ratings and review manipulation, credit card fraud or other fraudulent activity.

Adhering to these policies should help you make choices that enhance your app's appeal and audience. Your apps are crucial to the experience of hundreds of millions of customers. We can't wait to see what you create and are thrilled to help deliver your apps to the world.

If you have feedback on the policies, please let us know by commenting in our forum. We will consider every comment.

App Policies

Distinct Function & Value; Accurate Representation

10.1

Your app and its associated metadata must accurately and clearly reflect the source, functionality, and features of your app.

- The screenshots, app name, developer name, tile, category and app description, and any other related metadata you provide with your app, should make it easy for a user to understand the functions, features, and any important limitations of your app.
- Your app must be fully functional and offer customers unique, creative value or utility, and your app may not use a name or icon similar to that of other apps.
- You should not represent your app to be from a company, government body, or other entity if you do not have permission to make that representation.
- If your app contains content or features restricted to certain languages, markets or geographies or has other important limitations, they should be clearly described.
- If your app declares an association with a given file type or extensibility point, it must provide the functionality that a customer would expect, given that association. For example, if your app declares an association with the .jpg file type, the app should be able to open the image for viewing or editing.

Security

10.2

Your app must not jeopardize or compromise user security, or the security or functionality of the Windows or Windows Phone device(s), system or related systems and must not have the potential to cause harm to users or any other person.

10.2.1

Apps must always run within the permitted sandbox provided by the system. Thus, the following applies:

- Your app must not invoke code outside the allowed security model. For example, see Security Changes in the .NET Framework 4.
- Your app must not communicate with local desktop applications (on Windows devices) or services via local mechanisms, including via files and registry keys.
- Your app may only depend on Windows Runtime APIs allowed for Store apps and must remain within the boundaries allowed for Store apps (e.g., cannot break out of the AppContainer).

10.2.2

Your app must not attempt to change or extend the described functionality through any form of dynamic inclusion of code that is in violation of Store Policies. Your app should not, for example, download a remote script and subsequently execute that script in a manner that is not consistent with the described functionality.

App is Testable

10.3

The app must be testable. If it is not possible to test your app for any reason, including, but not limited to, the items below, your app may fail this requirement.

· If your app requires login credentials, provide us with a working demo account using the Notes to Tester field.

· If your app requires access to a server, the server must be functional to verify that it's working correctly.

· If your app allows a user to add a gift card balance, give us a gift card number that can be used in the testing.

Usability

10.4

Your app must meet Store standards for usability, including, but not limited to, those listed in the subsections below.

10.4.1

The app must run on devices that are compatible with the software, hardware and screen resolution requirements specified by the application.
If an app is downloaded on a device with which it is not compatible, it should detect that at launch and display a message to the customer detailing the requirements.

10.4.2

The app supports graceful shutdown. The app must handle exceptions raised by any of the managed or native system APIs and not close unexpectedly. An app that closes unexpectedly fails certification. The app must continue to run and remain responsive to user input after the exception is handled.

10.4.3

The app must start up promptly and must stay responsive to user input.

10.4.4

Where applicable, pressing the back button should take the user to a previous page/dialog. If the user presses the back button on the first page of the app, then the app terminates (unless it is allowed to run in the background).

Personal Information

10.5

The following requirements apply to apps that access personal information. Personal information includes all information or data that identifies or could be used to identify a person, or that is associated with such information or data. Examples of personal information include: name and address, phone number, biometric identifiers, location, contacts, photos, audio & video recordings, documents, SMS, email, or other text communication, screen shots, and in some cases, combined browsing history.

10.5.1

If your app accesses, collects or transmits personal information, or if otherwise required by law, you must maintain a privacy policy. You must provide users with access to your privacy policy in the Description page of your app (including by submitting it to Microsoft for display to users). In addition, your privacy policy must be accessible from your app at any time. The privacy policy can be hosted within or directly linked from the app. Your privacy policy must inform users of the personal information accessed, collected or transmitted by your app and how that information is used, stored, secured and disclosed. It must describe the controls that users have over the use and sharing of their information and how they may access their information, and it must comply with applicable laws and regulations. Your privacy policy must be kept up-to-date as you add new features and functionality to your app.
Additionally, apps that receive device location must provide settings that allow the user to enable and disable the app's access to and use of location from the Location Service API. For Windows Phone 8 and Windows Phone 8.1 apps, these settings must be provided in-app. For Windows Mobile 10 apps, these settings are provided automatically by Windows within the Settings App (on the Settings->Privacy->Location page).

10.5.2

Your app can publish a customer's personal information to an outside service or another person only after obtaining opt-in consent. Opt-in consent means the customer gives their express permission in the app user interface for the requested activity, after you have:
(a) described to the customer how the information will be accessed, used or shared; and
(b) provided the customer a mechanism in the app user interface through which they can later rescind this permission and opt-out.

10.5.3

If your app publishes a person's personal information to a service or a third party, but the person whose information is being shared is not a customer of your app, you must obtain express written consent to publish that personal information, and you must permit the person whose information is shared to withdraw that consent at any time. If your app provides a customer with access to another person’s personal information, this requirement would also apply.

Capabilities

10.6

The capabilities you declare must legitimately relate to the functions of your app, and the use of those declarations must comply with our app capability declarations. You must not circumvent operating system checks for capability usage.
For more information about app capability declarations, see App capability declarations.

Localization

10.7

You must localize your app for all languages that it supports. The text of your app's description must be localized in each language that you declare. If your app is localized such that some features are not available in a localized version, you must clearly state or display the limits of localization in the app description. The experience provided by an app must be reasonably similar in all languages that it supports.

Financial Transactions

10.8

If your app includes in-app purchase, billing functionality or captures financial information, the following requirements apply:

10.8.1

- You must use the Microsoft in-app purchase API to sell digital items or services that are consumed or used within your app.
- In-app products sold in your app cannot be converted to any legally valid currency (e.g. USD, Euro, etc.) or any physical goods or services.
- If you created your app on or before June 29, 2015, you may continue using a third-party API for digital items or services until June 29, 2016.

10.8.2

You must use a secure third party purchase API for purchases of physical goods or services, real world gambling or charitable contributions. If your app is used to facilitate or collect charitable contributions or to conduct a promotional sweepstakes or contest, you must do so in compliance with applicable law. You must also state clearly that Microsoft is not the fundraiser or sponsor of the promotion.
The following requirements apply to your use of a secure third party purchase API:

- At the time of the transaction or when you collect any payment or financial information from the customer, your app must identify the commerce transaction provider, authenticate the user, and obtain user confirmation for the transaction.
- The app can offer the user the ability to save this authentication, but the user must have the ability to either require an authentication on every transaction or to turn off in-app transactions.

- If your app collects credit card information or uses a third-party payment processor that collects credit card information, the payment processing must meet the current PCI Data Security Standard (PCI DSS).

10.8.3

If your app collects financial account information, you must submit that app from a Business account type.

10.8.4

You must provide in-app purchase information about the types of in-app purchases offered and the range of prices. You may not mislead customers about the nature of your in-app promotions and offerings.

Notifications

10.9

Your app must respect system settings for notifications and remain functional when they are disabled. This includes the presentation of ads and notifications to the customer, which must also be consistent with the customer's preferences, whether the notifications are provided by the Microsoft Push Notification Service (MPNS), Windows Push Notification Service (WNS) or any other service. If the customer disables notifications, either on an app-specific or system-wide basis, your app must remain functional.
If your app uses MPNS or WNS to transmit notifications, it must comply with the following requirements:

10.9.1

Because notifications provided through WNS or MPNS are considered app content, they are subject to all Store Policies.

10.9.2

You may not obscure or try to disguise the source of any notification initiated by your app.

10.9.3

You may not include in a notification any information a customer would reasonably consider to be confidential or sensitive.

Advertising Conduct and Content

10.10

For all advertising related activities, the following requirements apply:

10.10.1

- The primary purpose of your app should not be to get users to click ads.
- Your app may not do anything that interferes with or diminishes the visibility, value, or quality of any ads it displays.
- Your app must respect advertising ID settings that the user has selected.

10.10.2

If you purchase or create promotional ad campaigns to promote your apps through the “Promote Your App” capability in Dev Center, all ad materials you provide to Microsoft, including any associated landing pages, must comply with Microsoft’s Creative Specifications Policy and Creative Acceptance Policy.

10.10.3

Any advertising content your app displays must adhere to Microsoft’s Creative Acceptance Policy.
If your app displays ads, all content displayed must conform to the advertising requirements of the App Developer Agreement, including the following requirements:

10.10.4

The primary content of your app may not be advertising, and advertising must be clearly distinguishable from other content in your app.

10.10.5

Your privacy statement or terms of use must let users know you will send personal information to the ad service provider and must tell users how they can opt-out of interest-based advertising.

10.10.6

If your app is directed at children under the age of 13 (as defined in the Children’s Online Privacy Protection Act), you must notify Microsoft of this fact in Dev Center and ensure that all ad content displayed in your app is appropriate for children under the age of 13.

Mobile Voice Plans

10.11

Your app may not sell, link to, or otherwise promote mobile voice plans.

Content Policies

The following policies apply to content and metadata (including publisher name, app name, app icon, app description, and app screenshots and any other app metadata) offered for distribution in the Store. Content means the app name, publisher name, app icon, app description, the images, sounds, videos and text contained in the app, the tiles, notifications, error messages or ads exposed through your app, and anything that’s delivered from a server or that the app connects to. Because Windows and Windows Phone apps and the Store are used around the world, these requirements will be interpreted and applied in the context of regional and cultural norms.

General Content Requirements

11.1

- Apps that contain content that would warrant a rating over PEGI 16 or ESRB MATURE are not allowed, unless the app is a game, is rated by a ratings board that is supported in the Store, and otherwise complies with all Store Policies.
- Metadata and other content you submit to accompany your app may contain only content that would merit a rating of PEGI 12, ESRB EVERYONE 10+, or lower.

Content Including Names, Logos, Original and Third Party

11.2

All content in your app and associated metadata must be either originally created by the application provider, appropriately licensed from the third-party rights holder, used as permitted by the rights holder, or used as otherwise permitted by law.

Harm to Others

11.3

Your app must not contain any content that facilitates or glamorizes extreme or gratuitous violence, human rights violations or the creation or illegal use of weapons against a person or animal in the real world.

Defamatory, Libelous, Slanderous and Threatening

11.4

Your app must not contain any content that is defamatory, libelous, slanderous, or threatening.

Offensive Content

11.5

Your app must not contain any potentially offensive content or content that advocates discrimination, hatred, or violence based on considerations of race, ethnicity, national origin, language, gender, age, disability, religion, sexual orientation, status as a veteran, or membership in any other social group.

Alcohol, Tobacco, Weapons and Drugs

11.6

Your app must not contain any content that facilitates or glamorizes excessive or irresponsible use of alcohol or tobacco products, drugs, or weapons.

Adult Content

11.7

Your app must not contain or display content that a reasonable person would consider pornographic or sexually explicit.

Illegal Activity

11.8

Your app must not contain content or functionality that encourages, facilitates or glamorizes illegal activity in the real world.

Excessive Profanity and Inappropriate Content

11.9

- Your app must not contain excessive or gratuitous profanity.
- Your app must not contain or display content that a reasonable person would consider to be obscene.

Country/Region Specific Requirements

11.10

Content that is offensive in any country/region to which your app is targeted is not allowed. Content may be considered offensive in certain countries/regions because of local laws or cultural norms. Examples of potentially offensive content in certain countries/regions include the following:
China

- Prohibited sexual content
- Disputed territory or region references
- Providing or enabling access to content or services that are illegal under applicable local law

General Rating Requirements

11.11

- If a rating under any particular rating system is required by law in any country, you must assign a rating to your app. If applicable law also requires documentation, include the rating documentation for that rating system along with your app.
- If your app is optionally rated by a Ratings Board for Windows Apps, you must submit a valid rating certificate for the app and include the appropriate content descriptors in your product description.

Age Ratings

11.12

Your app must have a Windows Store age rating, and you must submit third-party ratings for your app if you have them.

11.12.1

You must assign a Windows Store age rating that most accurately matches your app. See Windows Store age ratings and boards for detailed descriptions of the content that is suitable for each Windows Store age rating.
The screenshots, app name, developer name, tile, category and app description, and any other related metadata you provide with your app must be appropriate for the lower of 1) the age rating you assign your app or 2) Windows Store age rating 12+. If your app has multiple rating board ratings that each correspond to different Windows Store age ratings, you should assign the Windows Store age rating that corresponds to the older audience.

11.12.2

You must assign a rating of at least 12+ if your app:

- Provides a user with uncontrolled access to online social networks, or uncontrolled sharing of personal information with third parties, including other gamers or online acquaintances. For such activity to be considered "controlled", your app must be governed by controls that require parental permission to use these features.
- Is a web browser, enabling broad access to the internet, or a search engine that enables users to search broadly across the Internet.

11.12.3

If your app provides content (such as user-generated, retail or other web-based content) that might be appropriate for a higher Windows Store age rating than you assigned, you must enable users to opt in to receiving such content by using a content filter or by signing in with a pre-existing account.

11.12.4

If a rating under any particular rating system is required by law in any country, you must include the rating documentation for that rating system along with your app.
If your app is optionally rated by a Ratings Board for Windows Apps, you must submit a valid rating certificate for the app and include the appropriate content descriptors in your product description.

Windows Store age ratings and boards, they are enumerated here for convenience

To publish an app in the Windows Store, you must give it an age rating. The age rating is about the suitability of the content in the app, rather than the age of the target audience. Some countries and regions require that you also rate your app through a specific ratings board.

Windows Store age ratings

Here are the age ratings used in the Windows Store with descriptions of appropriate app behavior and content.

Age rating	Description
3+ (Suitable for young children)	These apps contain content suitable for young children. There may be minimal comic violence in non-realistic, cartoon form. Characters should not resemble or be associated with real life characters. There should be no content that could be frightening, and there should be no nudity or references to sexual or criminal activity. Apps with this age rating also cannot enable features that could access content or functionality unsuitable for young children, such as uncontrolled online sharing of information (such as that described under the 12+ ratings category).
7+ (Suitable for ages 7 and older)	Apps with this age rating have the same criteria as the 3+ applications, except these apps can include content that might frighten a younger audience and can contain partial nudity, as long as the nudity doesn't refer to sexual activity. This rating should only be used for apps where the content is suitable for children.
12+ (Suitable for ages 12 and older)	Apps with this age rating can contain increased nudity of a non-sexual nature, slightly graphic violence towards non-realistic characters, or non-graphic violence towards realistic human or animal characters. This age rating might also include profanity, but not of a sexual nature. Also, apps with this age rating or higher may allow for uncontrolled: (i) access to online social networks, or (ii) sharing of personal info with third parties, including other gamers or online acquaintances. (For such activity to be considered controlled, your app must include parental control features that require parental permission to use such sharing features, and you must identify those and explain their functionality in the Notes for certification.)
16+ (Suitable for ages 16 and older)	Apps with this age rating can depict realistic violence with minimal blood, and they can depict sexual activity. They can also contain drug or tobacco use and criminal activities, and more profanity than would be allowed in a 12+ app, within the limits laid out in the Store Policies.
18+ (Suitable for adults)	Games with this age rating may contain intense, gross or specific violence, blood or gore which is only suitable for an adult audience. All content must meet the content policies criteria.

When thinking about age ratings, make sure to consider the following:

- The content that your app provides. Remember, this content could be both stored locally with your app, or accessed from external sources, such as through an Internet connection.
- The images that might appear when a user opens your app.
- Any services or features that your app connects to that may have additional age requirements, such as an age minimum.

If you can't decide between two age ratings, or if your app has content that is suitable for different age groups, choose the strictest age requirement. For example, if you have an app with content that is appropriate for audiences who are 12 years old or older, but it uses an account on a service that requires members be 16 years old or older, your Windows Store age rating would be **16+**.

If you obtain an age rating for your app from a ratings board, you must select the Windows Store age rating that corresponds to that rating. If your app has multiple rating board ratings that each correspond to different Windows Store age ratings, you should assign the Windows Store age rating that corresponds to the older audience.

For more about age rating requirements, see the Age ratings section in the Windows Store Policies.

Ratings boards for Windows Store apps

Some countries and regions require that you rate your app through a ratings board, particularly if it is a game. (Note that the app’s purpose, rather than its listing category in the Windows Store, determines whether it is a game for this requirement.)

When you obtain a rating from one of these ratings boards, you will be able to submit the required certificates in the **Other age ratings** section of the App properties page during the app submission process. To do so, choose an age from the drop-down list that appears by each rating board name. These values correspond to the values used by each board. In some cases, checkboxes will appear; if so, check any of the boxes which apply to your app's content. Then, upload your certificate by dragging it into the field, or by browsing your files to select it.

Once your app is published, your app's listing will show this rating to customers in the region to which the rating applies. You must also select the Windows Store age rating level that corresponds to your rating certificate. See the Windows Store Policies for more details.

Here are the rating boards supported by the Windows Store.

System	Country/Region	Process	Required for games?

ESRB—Entertainment Software Rating Board	United States	Online form (no cost)	No
PEGI—Pan European Game Information	Europe	Online form (no cost)	No
CSRR—Computer Software Rating Regulation **Note** The Chinese version of these regulations take precedence over any translated versions.	Taiwan	Self-rating	Yes
DJCTQ—Departamento de Justiça, Classificação, Títulos e Qualificação	Brazil	Submission (or self-rating, if you already have a PEGI or ESRB rating)	Yes
GRB—Game Rating Board	Korea	Submission	Yes
FPB—Film and Publication Board	South Africa	Submission	No
USK—Unterhaltungssoftware Selbstkontrolle (Entertainment Software Self-Regulation)	Germany	Submission	No
CERO—Computer Entertainment Rating Organization	Japan	Submission	No
COB—Classification Operations Board	Australia	Submission	No

OFLC-NZ—Office of Film and Literature Classification	New Zealand	Submission	No
PCBP—Russian Age Rating System	Russia	Self-rating (based on PEGI)	Yes

Ratings boards which use an online form or a self-rating process tend to work very quickly; ratings boards which require you to submit your app for review usually take longer (often several weeks). For more information on the processes, times, and any fees charged, visit the website for each specific ratings board.

Windows has reached 200,000 milestone,
Windows Store finally reaches the 200,000 app milestone, but are you impressed?
To have the app being one 200,000+ , this is what you need to do. And how.

Using the unified Windows Dev Center dashboard

The new unified Windows Dev Center dashboard lets you manage and submit all of your apps for Windows devices in one place. New features simplify processes while giving you more control. You'll also find detailed analytic reportscombined payout details, ways to promote your app and engage with your customers, and much more.

Note You must have a developer account in order to access the Windows Dev Center dashboard.

When you create a create a submission for your app (for the first time or as an update), you can upload app packagesfor any supported operating system, and the Store will automatically offer customers the package that will work best for them. You can use the same descriptions and artwork for all of the operating systems your app supports, or you cancustomize the description for specific platforms.

In-app products (IAPs) can now be managed independently of your app's submission. This brings you greater flexibility and the ability to make changes independently. For more info, see IAP submissions.

The new dashboard also offers capabilities for app management and configuring services, detailed analytic reports,payout info, opportunities for promotion and customer engagement, ways to conduct beta testing, and more.

Tips for using the new dashboard

The first time you visit the new dashboard, you will be prompted to read and accept the App Developer Agreement, even if you had already accepted the latest version in the earlier dashboard.

If you had previously published apps, take a look at How your app appears in the Store for Windows 10 customers to find out more about some of the changes in your app's listing for Windows 10 customers, and how we categorize your app's info in the new dashboard.

Apps that were linked to share identity in the earlier Windows Phone Dev Center and Windows Store dashboard will appear as the same app in the new Dev Center dashboard. No matter which OS versions your app supports, you'll now manage it as one app. The app will share the same details such as price and category across devices. We're continuing to add features and capabilities to the new Dev Center dashboard. If you have suggestions, let us know by using the Windows Platform Developer (WPDev) Feedback site to send your feedback.

How your app appears in the Store for Windows 10 customers

If you had previously published apps to the Store for either Windows or Windows Phone, those apps will be made available to customers on Windows 10 devices as well. Since there are some changes in the way the Store presents and categorizes apps to customers running Windows 10, this topic will help you understand what may have changed.

Note If you want to change any of these details, create a new submission and make your changes, then submit the update to the Store.

Consideration for apps that shared identity in the Windows Store and Windows Phone Store

If you have used the same reserved name for an app published to both Stores (often referred to as sharing your app's identity), these will now be considered one app, not two. In the dashboard, you'll see them as a single app with Windows and Windows Phone packages.

Most developers had set the same pricing and other properties for the app and any in-app products (IAPs) in each Store, but if some of these values were different, it's important to understand which ones are shown to your Windows 10 customers.

- **Pricing:** If you had chosen different base prices for your app (or IAP) in each Store, the base price from the Windows Store is used.

Note If you had set per-market pricing in the Windows Phone Store, custom prices will also be shown to your Windows 10 customers.

- **Free trials:** Trial options were different in the two earlier Stores, so if you used them, it's possible that chose different options for each Store. The trial option available to your Windows 10 customers are determined based on the following table.

Windows 8 app	Windows Phone app	Trial setting for Windows 10
No free trial	No free trial	No free trial

No free trial	Trial never expires	No free trial
Trial never expires	Trial never expires	Trial never expires
Trial never expires	No free trial	No free trial
Time-limited trial	Trial never expires	No free trial on Windows Phone 8.1 and earlier; otherwise, time-limited trial
Time-limited trial	No free trial	No free trial on Windows Phone 8.1 and earlier; otherwise, time-limited trial

- **Markets:** Your app will be available to Windows 10 customers in every market where you had previously published the app. This applies even if you had different market selections for each Store.
- **Categories:** If your app appeared in different categories in the two Stores, we'll use the category from the Windows Store to determine its new category. Note that some categories are different in the Store for Windows 10 customers, so be sure to review the table below.
- **Age rating:** If you provided different age ratings, the stricter (higher age) rating is used.
- **Privacy policy:** If your app has a privacy policy, the one you provided when submitting your Windows 8 app is shown to your Windows 10 customers as well.
- **Screenshots:** We take all of the screenshots you've submitted and use the appropriate version to display to Windows 10 customers, based on the type of device they're using. In the rare case where your supported languages differ for each Store, some customers might see a screenshot from another language that best represents the experience they will get when buying the app.
- **Descriptions:** We try to show the most appropriate description to your Windows 10 customers, based on their language. When descriptions are available from more than one source in the same language, the description fron your Windows Store app is shown to your Windows 10 customers. In the rare cases where your supported languages differ for each Store, some customers might see a description from your Windows Phone app, if that is the only description you provided in that language.

If you want to update the description that your Windows 10 customers see to let them know about experiences that work across multiple devices, you can do this by updating your app's description. Customers on Windows 10 will see your app's default description, but you can also create platform-specific descriptions if you want your description to appear differently for customers on different OS versions.

Category changes

In many cases, the new categories and subcategories for apps and games are the same as they have been in the Store for previous OS versions. However, there have been a few changes. Review the table below to understand how your app is categorized in the Store for customers on Windows 10, based on its previous category.

Note You'll see the new category listed in the dashboard when viewing your app's category in the App propertiespage of a submission, and customers viewing the Store on Windows 10 devices will see your app in the new category. However, customers viewing the Store from an earlier operating system will continue to see the app listed in its original category.

Category changes for Windows Phone apps:

Previous category	**New category**
Government + politics > commentary	Government + politics
Government + politics > legal issues	Government + politics
Government + politics > politics	Government + politics
Government + politics > resources	Government + politics
Health + fitness > diet + nutrition	Health + fitness
Health + fitness > fitness	Health + fitness
Health + fitness > health	Health + fitness
Lifestyle > art + entertainment	Lifestyle

Lifestyle > out + about	Lifestyle
Lifestyle > food + dining	Food + dining
Lifestyle > shopping	Shopping
News + weather > international	News + weather
News + weather > local + national	News + weather
Utilities + productivity	Utilities + tools
Travel + navigation	Travel
Travel + navigation > planning	Travel
Travel + navigation > tools	Travel
Travel + navigation > with kids	Kids + family > travel
Travel + navigation > language	Education > Language
Travel + navigation > mapping	Navigation + maps
Travel + navigation > navigation	Navigation + maps
Games > classics	Games > Action + adventure

Games > family	Games > Family + kids
Games > sports + recreation	Games > Sports
Games > strategy + simulation	Games > Strategy

Category changes for Windows 8 apps:

Previous category	**New category**
Books + Reference > Kids	Kids + family > Books + reference
Music + Videos > Video	Photo + video
Music + Videos > Music	Music
Government	Government + politics
Finance	Personal Finance
Games > Action	Games > Action + adventure
Games > Adventure	Games > Action + adventure
Games > Arcade	Games > Action + adventure
Games > Card	Games > Card + board
Games > Kids	Games > Family + kids

Games > Family	Games > Family + kids
Games > Puzzle	Games > Puzzle + trivia
Games > Racing	Games > Racing + flying

In addition, to communicate with the customers, you can make use of the app declarations.

App declarations

You can provide additional information about your app in the **App declarations** section of the **App properties** page during the submission process. These declarations can help make sure your app is displayed appropriately and offered to the right set of customers, or can indicate how customers can use your app.
The following sections describe each declaration and what you need to consider when determining whether each declaration applies to your app.

This app allows users to make purchases, but does not use the Windows Store commerce system.

Most apps should leave this box unchecked, since apps which offer opportunities to make in-app purchases generally use the Microsoft in-app purchase API to create and submit the IAPs. Per the App Developer Agreement, apps that were created and submitted prior to June 29, 2015, may continue to offer in-app purchasing functionality without using Microsoft's commerce engine, so long as the purchase functionality complies with the Windows Store Policies. If this applies to your app, you must check this box. Otherwise, leave it unchecked.

This app has been tested to meet accessibility guidelines.

Checking this box makes your app discoverable to customers who are specifically looking for accessible apps in the Store.
You should only check this box if you have done all of the following items:

- Set all the relevant accessibility info for UI elements, such as accessible names.

· Implemented keyboard navigation and operations, taking into account tab order, keyboard activation, arrow keys navigation, shortcuts.

· Ensured an accessible visual experience by including such things as a 4.5:1 text contrast ratio, and don't rely on color alone to convey info to the user.

· Used accessibility testing tools, such as Inspect or AccChecker, to verify your app, and resolve all high-priority errors detected by those tools.

· Verified the app’s key scenarios from end to end using such facilities and tools as Narrator, Magnifier, On Screen Keyboard, High Contrast, and High DPI.

When you declare your app as accessible, you agree that your app is accessible to all customers, including those with disabilities. For example, this means you have tested the app with high-contrast mode and with a screen reader. You've also verified that the user interface functions correctly with a keyboard, the Magnifier, and other accessibility tools.

For more info, see Accessibility for Windows Runtime apps, Testing your app for accessibility, and Declaring your app as accessible in the Windows Store.

Important Don't list your app as accessible unless you have specifically engineered and tested it for that purpose. If your app is declared as accessible, but it doesn’t actually support accessibility, you'll probably receive negative feedback from the community.

Customers can install this app to removable media, such as SD cards.

This box is checked by default, to allow customers to install your app to removable storage media such as an SD card.

If you want to prevent your app from being installed to removable media, uncheck this box.

Note For Windows Phone 8.1, this was previously indicated via StoreManifest.xml.

Windows can include this app's data in automatic backups to OneDrive.

This box is checked by default, to allow your app's data to be included when a customer chooses to have Windows make automated backups to OneDrive.

If you want to prevent your app's data from being included in automated backups, uncheck this box.

Say hello to Xamarin Profiler Xamarin Dev Days Available On-Demand