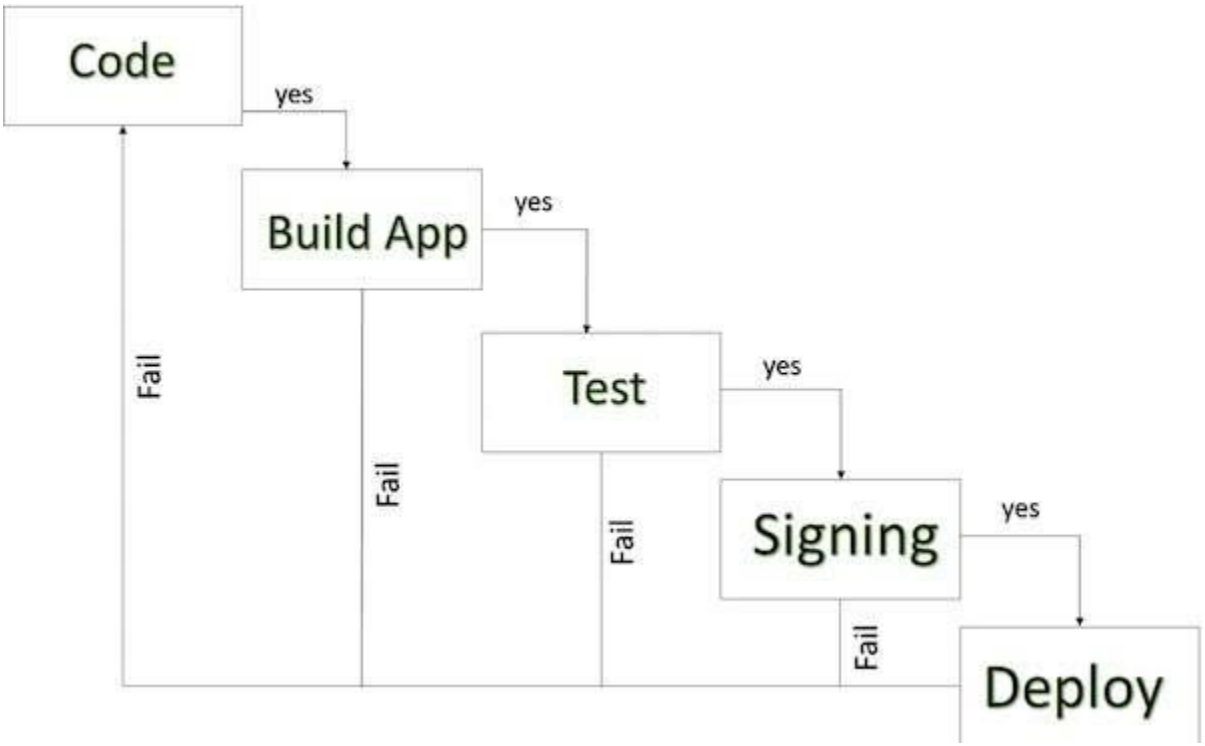
LECTURE NOTES

Android application publishing.

Android application publishing is a process that makes your Android applications available to users. Infect, publishing is the last phase of the Android application development process.



Android development life cycle

Once you developed and fully tested your Android Application, you can start selling or distributing free using Google Play (A famous Android marketplace). You can also release your applications by sending them directly to users or by letting users download them from your own website.

Here is a simplified check list which will help you in launching your Android application −

Step	Activity
1	**Regression Testing** Before you publish your application, you need to make sure that its meeting the basic quality expectations for all Android apps, on all of the devices

	that you are targeting. So perform all the required testing on different devices including phone and tablets.
2	**Application Rating** When you will publish your application at Google Play, you will have to specify a content rating for your app, which informs Google Play users of its maturity level. Currently available ratings are (a) Everyone (b) Low maturity (c) Medium maturity (d) High maturity.
3	**Targeted Regions** Google Play lets you control what countries and territories where your application will be sold. Accordingly, you must take care of setting up time zone, localization, or any other specific requirement as per the targeted region.
4	**Application Size** Currently, the maximum size for an APK published on Google Play is 50 MB. If your app exceeds that size, or if you want to offer a secondary download, you can use APK Expansion Files, which Google Play will host for free on its server infrastructure and automatically handle the download to devices.
5	**SDK and Screen Compatibility** It is important to make sure that your app is designed to run properly on the Android platform versions and device screen sizes that you want to target.
6	**Application Pricing** Deciding whether your app will be free or paid is important because, on Google Play, free apps must remain free. If you want to sell your application then you will have to specify its price in different currencies.
7	**Promotional Content** It is a good marketing practice to supply a variety of high-quality graphic assets to showcase your app or brand. After you publish, these appear on your product details page, in store listings and search results, and elsewhere.

8	**Build and Upload release-ready APK** The release-ready APK is what you will upload to the Developer Console and distribute to users. You can check complete detail on how to create a release-ready version of your app: Preparing for Release.
9	**Finalize Application Detail** Google Play gives you a variety of ways to promote your app and engage with users on your product details page, from colorful graphics, screenshots, and videos to localized descriptions, release details, and links to your other apps. So you can decorate your application page and provide as much clear crisp detail as you can provide.

Distribution platforms

Benefits of App Distribution Platforms

Mobile app distribution platforms are all similar in that they allow users to download a variety of apps across all categories, but they have some differences, particularly for developers and companies that add their app to their specific platform.

For example, each platform offers different ways of tracking app analytics, like the number of downloads, search results, and more.

For this reason and more, it's a smart move to upload your app to as many of these platforms as possible.

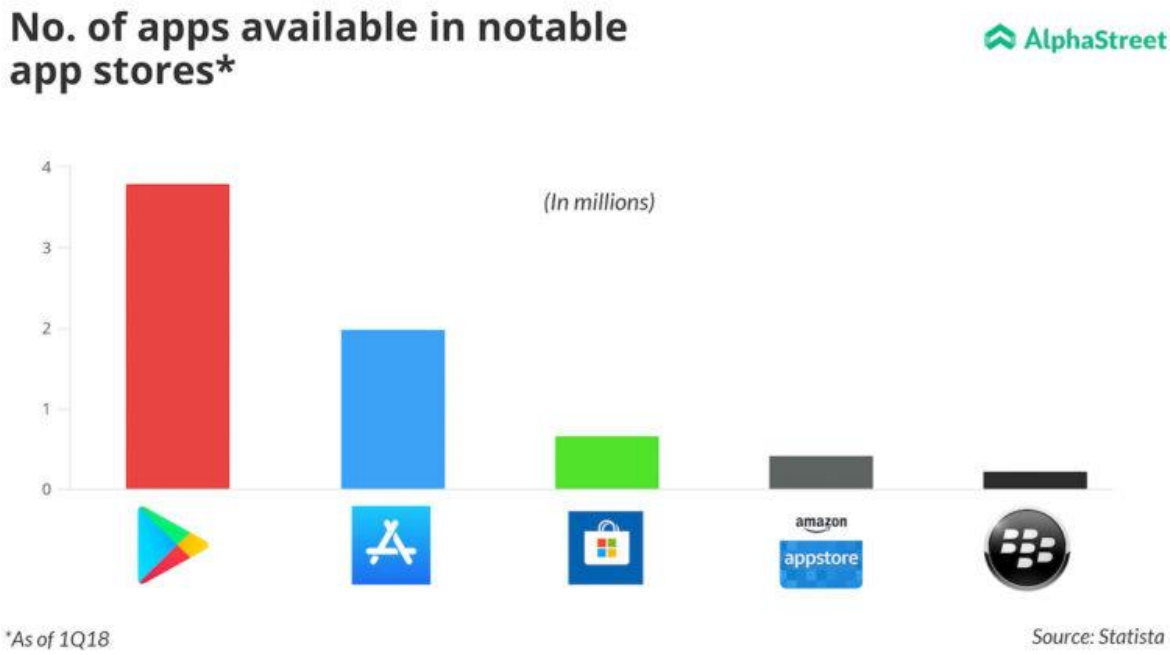
Not only does it widen your reach, but it also expands access to tools and resources that may not be available on just one platform.

But if you have a cross-platform app that can be used on all devices, it's worth exploring the major mobile platforms.

Top Mobile Software Platforms

Everyone is well-aware of the major mobile application marketplaces, like the App Store and Google Play, which we'll cover first, but what other contenders are out there and how do they differ?

Some are android app distribution platforms and others are distribution platforms for ios.



1. **Google Play**

Google Play, formerly known as Android Market, is the official app store for devices running on Android OS and has over 3.48 million apps available as of 2021, making it the largest app store on the market.

It also offers users a digital media store with books, music, movies, and tv programs, in addition to mobile apps.

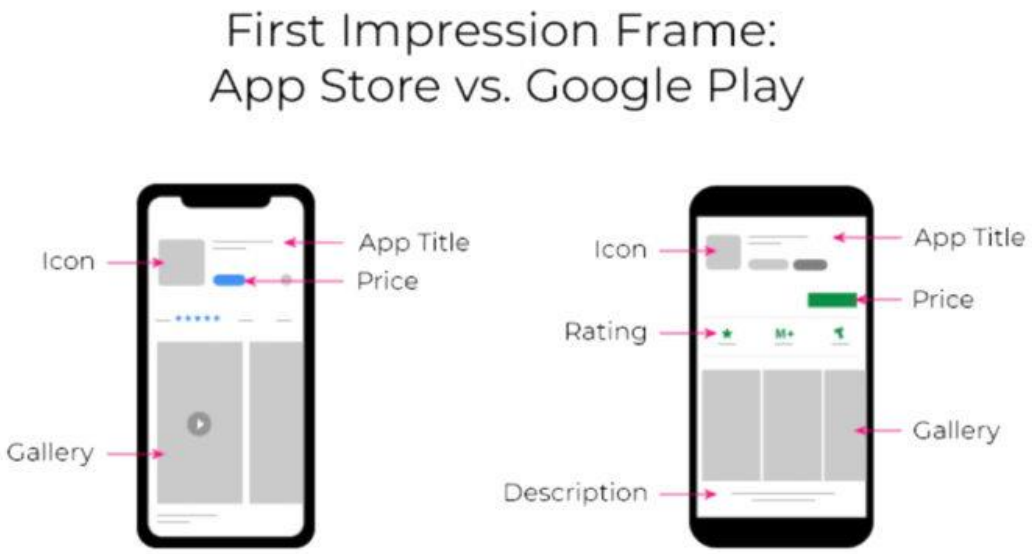
For developers to distribute and manage apps in Google Play, they'll need to pay a one-time $25 fee.

Opening a Google Play developer's account comes with features like multiple android package (APK) support, APK expansion files, application licensing, and access to Google Play developer API.

This API helps developers more efficiently manage their APKs and track purchases and subscriptions. It automates a variety of tasks, such as the following:

- Uploading and releasing new mobile app versions
- Editing your app store listings, including text and graphics
- Managing your in-app product catalog, purchases status, and app subscriptions

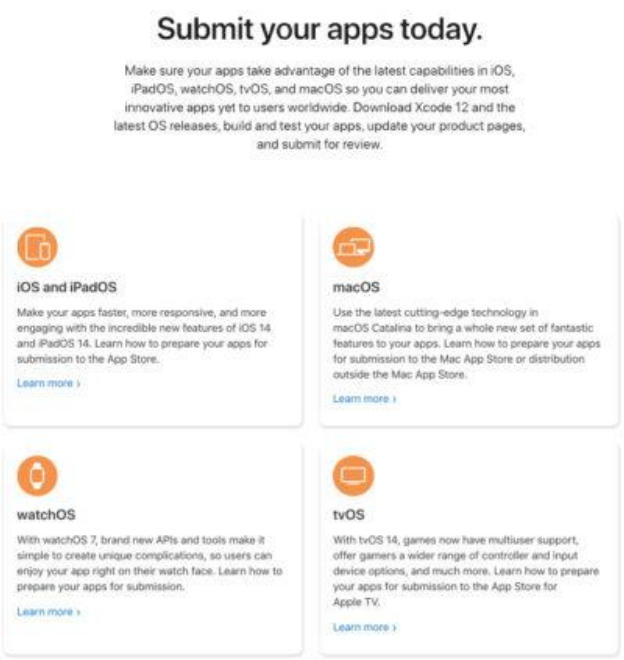
Another useful feature is device filtering, which allows developers to ensure their app gets to their target audience by filtering on a vast range of characteristics, like hardware features and platform versions.



2. **App Store**

Apple's App Store opened its doors in 2008 and is another one of the biggest and most widely used mobile app distribution platforms for iOS.

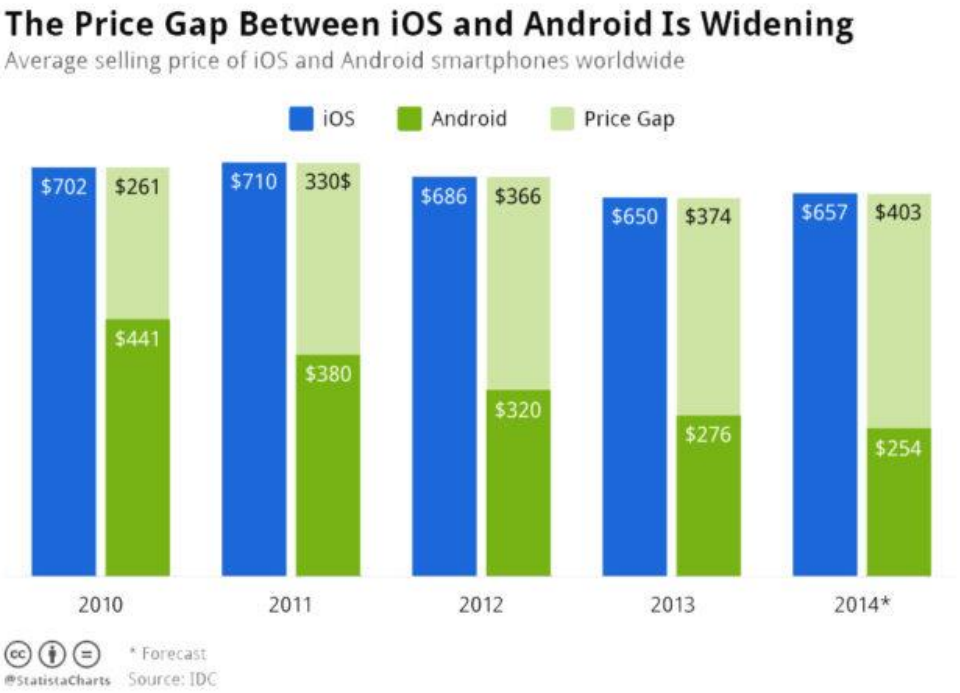
Users can browse and download apps on iPhone smartphones, iPad, Apple Watch, and other iOS devices in 175 regions.



As of the first quarter of 2021, there are roughly 2.2 million apps available in the App Store, making it the second-largest app distribution platform in the world.

Apple's Developer Program

Apple's Developer Program offers beta software and testing tools and advanced app capabilities, like Siri integration, Apple Pay, and iCloud data storage.
It also offers app analytics where developers can measure their marketing campaigns, user engagement, and monetization.



Publishing Cost

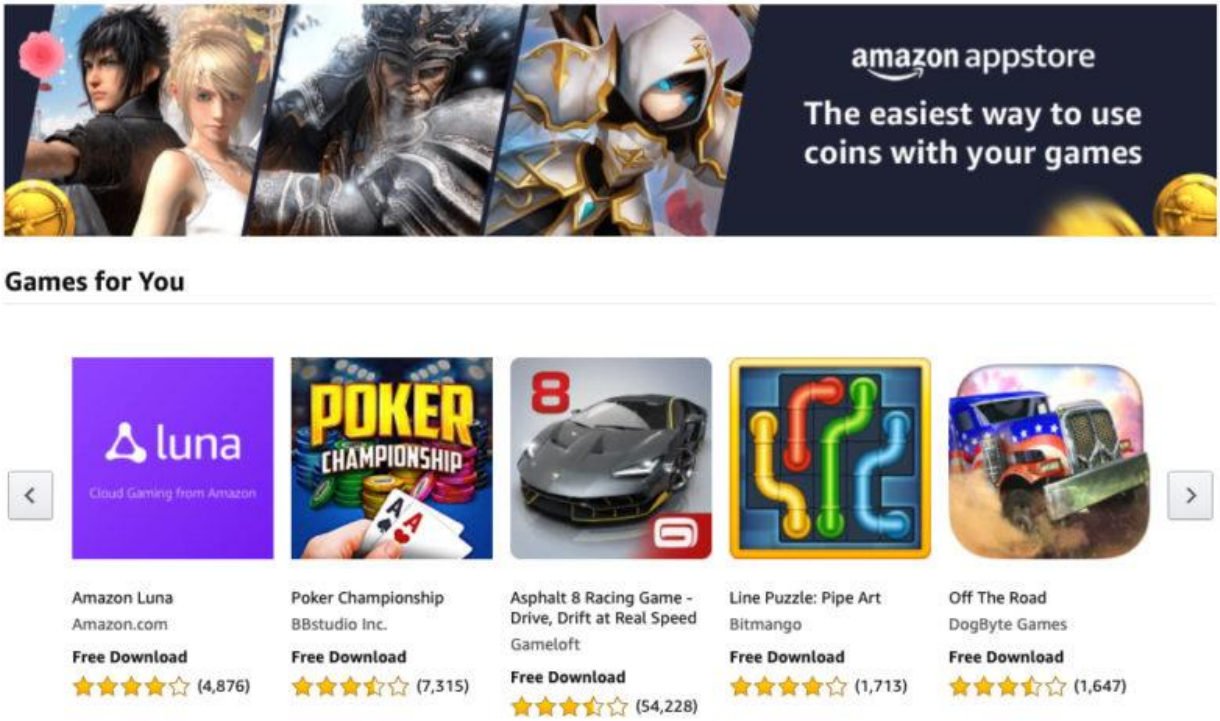
Developers will need to pay $99 each year to publish their app in the App Store, which also gives them access to Apple's Developer Program.

For non-profit organizations and governments, the yearly fee is waived.

3. **Amazon Appstore**

This is an app store for Android OS created by Amazon.com. As of Q1 2021, the Amazon Appstore has 460,619 mobile apps available for download.

In addition to offering Android apps, the Amazon Appstore also offers apps for Fire OS, an Amazon mobile operating system created for their Fire tablets, Fire TV devices, and Echo smart speakers.

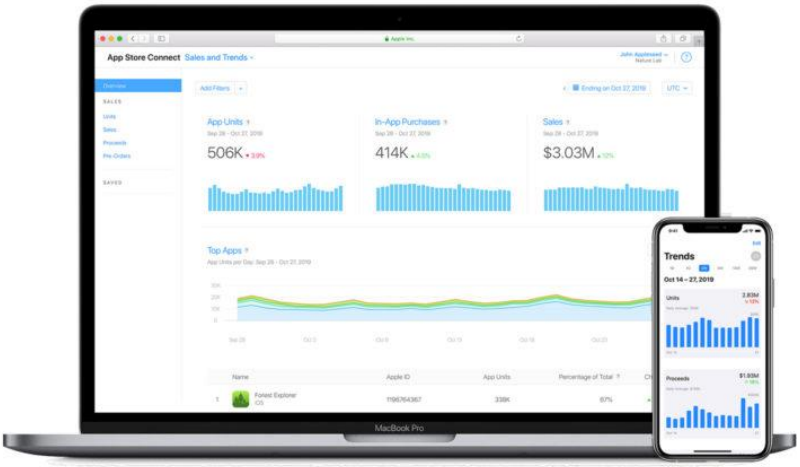


Developers can create an account on the Amazon Appstore developer portal for free, which allows them to submit and make their apps available.

There are no fees and an account comes with localized support.

Amazon boasts that 65% of developers on their store see similar or better revenue than their competitors and 67% say that publishing on the Amazon Appstore has allowed them to reach new audiences.

App Store Analytics Tools



Amazon Publisher Services (APS)

This service allows developers to monetize their apps through in-app bidding. A unified auction permits buyers to compete for your inventory, which can improve CPMs and boost revenue.

Microsoft and Amazon recently announced their plans to bring the Amazon Appstore to Windows 11, further expanding Amazon's user reach.

4. **Microsoft Store**

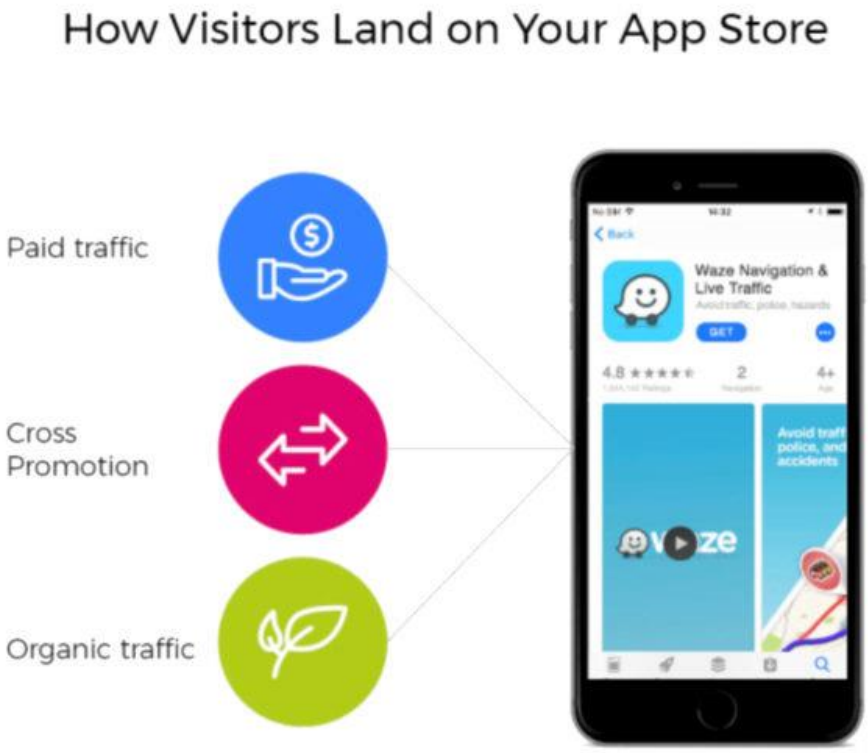
Formerly known as Windows Store, the Microsoft Store is a mobile app store by Microsoft.

With Windows 10, Microsoft merged their other storefronts like Windows Marketplace, Xbox Music, Xbox Video, Xbox Store, and Windows Phone Store all into Microsoft Store, creating a unified place for not only mobile apps but also console games, digital music and more.

Microsoft Store currently has more than 800,000 apps available for users to download.

To get started in uploading mobile apps to the Microsoft Store, developers can either create an individual account for $19 or a company account for $99, which is a one-time registration fee.

Developers uploading apps to the Microsoft Store can choose their markets, set and schedule pricing, and configure specific release dates.



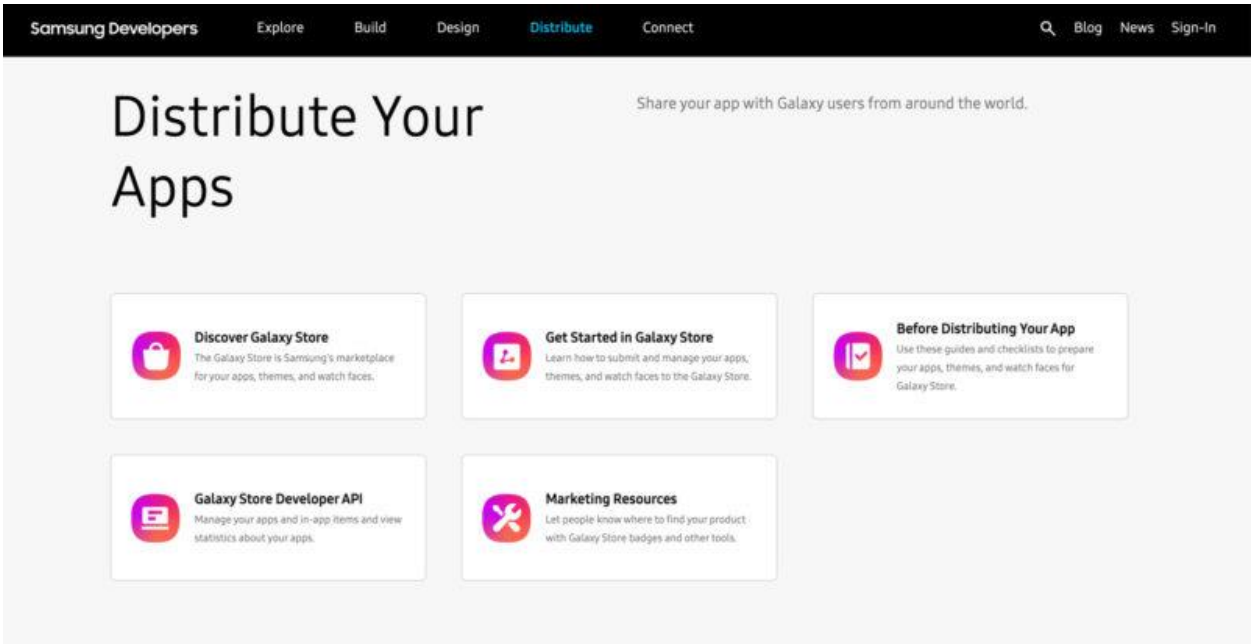
Similar to the App Store and Google Play, developers can manage their app listings, promote their apps with ads, and engage with their customers using targeted offers and push notifications.

The Microsoft Store also allows for beta testing and targeted distribution, package flights, gradual rollout and more.

5. **Samsung Galaxy Store**

Formerly known as Samsung Apps and Galaxy Apps, the Samsung Galaxy Store is an app store for Samsung devices, like Samsung Galaxy smartphones, Samsung Gear, and

feature phones.



Available in 188 countries, the Samsung Galaxy Store offers apps for Android, Windows Mobile, Tizen, and Bada platforms.

Unlike other app stores, it's completely free to upload and distribute Samsung Galaxy apps.

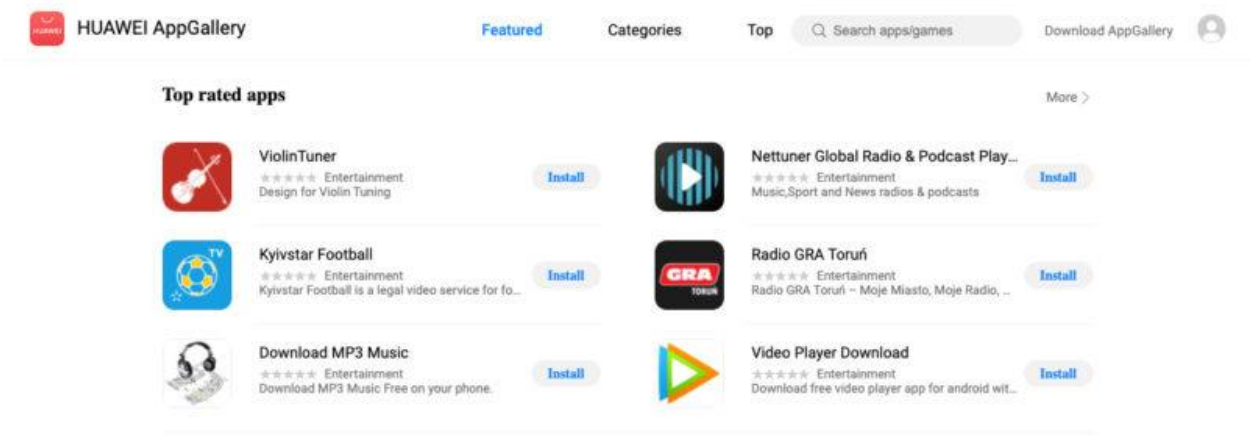
Developers can market and promote their app to the hundreds of millions of active Galaxy device users for Android apps and games, Galaxy Themes, or Galaxy Watch apps.

Another Galaxy Store feature for developers is Galaxy Store Statistics (GSS), which is a free tool that can be used for tracking and monitoring app performance and metrics like conversion, downloads, subscription, best SEO keywords, and more.

6. **Huawei AppGallery**

Huawei AppGallery is an app marketplace and package manager by Huawei technologies.

This application distribution platform was launched in 2011 in China but became available to the rest of the world in 2018. It is used by nearly 420 million active users on 700 million Huawei devices.

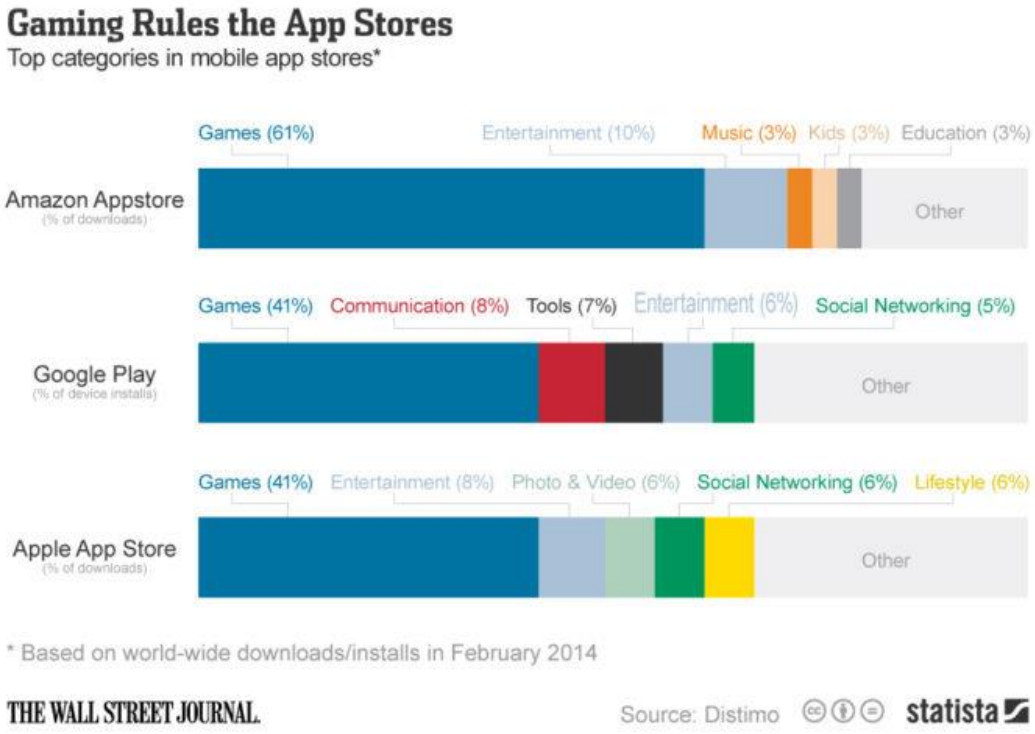


Their platform has over 45,000 apps available for download for Android OS and Huawei's HarmonyOS. Huawei AppGallery is the first app marketplace to have a developer identity verification system. It also offers a four-layer detection system for ultimate app security.

Similar to other mobile app distribution platforms, Huawei AppGallery allows developers to promote their apps for a wider audience reach. It's also completely free to create an account and upload apps.

Joint Operations

One feature unique to Huawei is its Joint Operations framework. This enables developers to work with Huawei AppGallery to manage and distribute their apps worldwide.

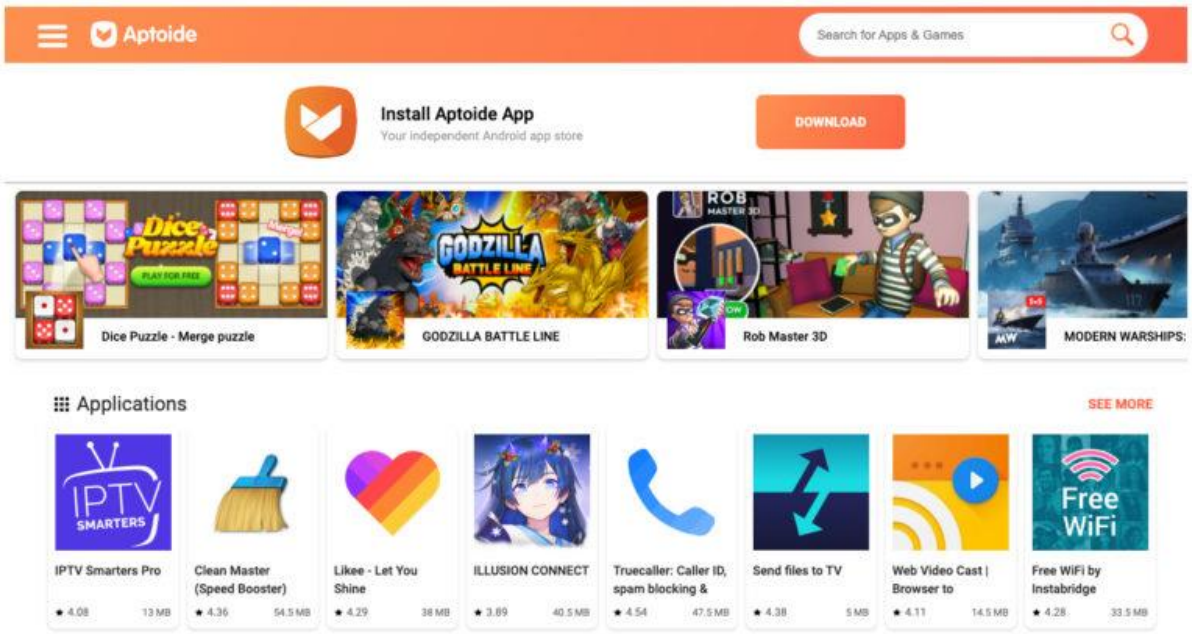


It offers development services and operation support and resources while also conducting various marketing activities to increase app use and transaction growth.

7. **Aptoide**

Aptoide is an app marketplace for Android OS and it has several app versions. There's Aptoide TV, Aptoide for tablets and smartphones, Aptoide Kids for children devices, and Aptoide VR.

It gets its name from "APT", which is the Debian package manager, and "oide" which is the last syllable of Android.



As of 2017, Aptoide has been used by 200 million users and contains roughly one million Android apps.

Developers can reach over 300 million users by creating an account to distribute their apps on Aptoide, which is free.

Catappult

Developed by Aptoide, Catappult is software that developers can use to distribute their APK through multiple app stores and other distribution channels worldwide using just one payment system.

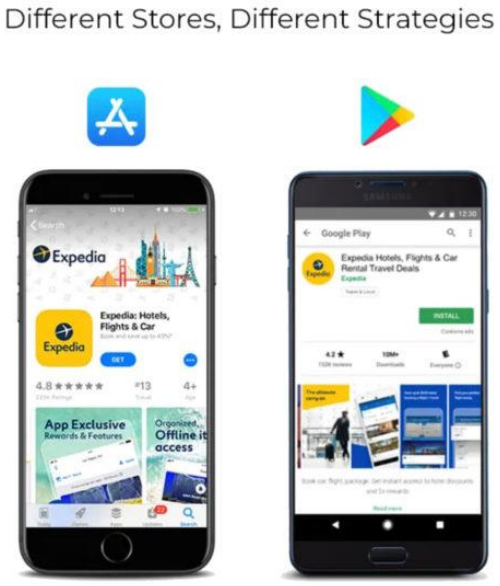
Catappult also makes use of blockchain technology to help prevent fraud. Their AppCoins open protocol guarantees decentralization and transparency.

Submitting Your iOS and Android App

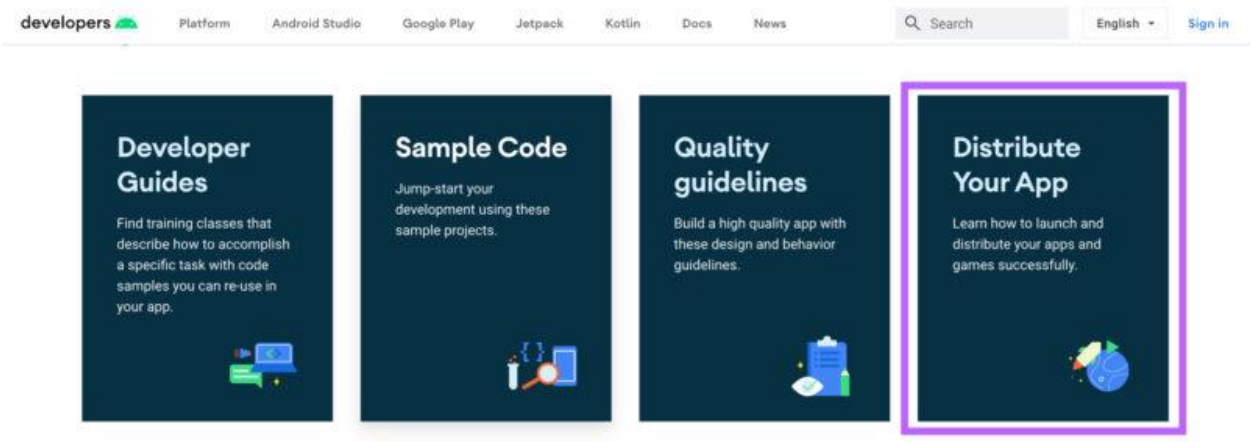
As you can tell there are many apps and places you can distribute them, and not just the two major mobile platforms like the App Store and Google Play. This app platform list could go on and on.

Something to keep in mind is that there are some similarities between the various app marketplaces out there, like the ways in which developers can monetize their app, but there are also some big differences, like submitting your app and adding your app listing.

Each mobile application marketplace will have it's own requirements and guidelines when it comes to submitting an app.



These guidelines include rules and requirements regarding things like monetization methods, data privacy, design, security, and so on.
Some things like security and user privacy may overlap for the major mobile platforms, but things like design and app listing requirements are sure to be different, even if only slightly so.
Additionally, when submitting your app to Android and App Store, or other app distribution platforms, you'll find that the features offered may also vary.
For example, Apple's App Store doesn't offer push app installation. Users have to do it manually on their mobile devices.



The Google Play Store, however, gives users more options like push app installation.
As long as you take time to read through each platform's documentation, you'll have a good understanding of what each one has to offer.

API Monetization

API monetization is the process by which businesses create revenue from their application programming interfaces (APIs). APIs are the cornerstone of what is widely seen as the next iteration of business development, where having well-developed APIs establishes and maintains relationships in a digital economy. APIs are the wholesale version of a web presence, allowing others to access and integrate your data and resources into their public or private sites and applications.

Hopefully by the time you have put an API management plan in place, you already have a healthy business model as well, which should provide a framework for your monetization goals. API monetization isn't just about how you are going to generate revenue via your API, it is also about how you will keep your API in operation and performing for consumers.

Not all APIs are created equal and reasons for deploying an API can vary widely.

COMMONLY EMERGED PATTERN FOR API MONETIZATION

- **Free**

A popular way to provide access to an API is by offering a free tier so that anyone can sign up, start using an API, and understand the value it delivers. This allows consumers to test the API and see if it will meet their needs before spending money. While *free* is a good option for many API monetization strategies, it works well in conjunction with other strategies. If implementing a free tier of the API alone, without a strategy to sell services to those with greater demands, your organization can face problems.

- **Consumer pays**

After providing free access, the next approach to API monetization is to establish a price that consumers will pay for the services or resources the API provides. We are seeing 3 common ***approaches to APIs charging consumers***:

- **Tiered.** Some API providers set up multiple tiers of paid access, such as bronze, gold, or platinum. Each tier has its own set of services and allowances for access to API resources with pricing stepping up in cost for each tier.
- **Pay-as-you go.** Another option is a utility-based model, where API consumers pay for what they use. Depending on the amount of bandwidth, storage, and other hard costs incurred around API consumption, providers charge based upon their cost, plus a logical profit.
- **Unit-based.** Finally, other API providers define each API resource in terms of units and assign a unit price. API consumers pay for the number of units they anticipate using, with the option to buy more when necessary

Some API providers mix and match different combinations of tiered, pay-as-you-go, and unit-based API pricing to recover operational costs as well as generate revenue.

- **Consumer gets paid**

In some cases, an API will drive other revenue streams for companies and can actually share revenue with API consumers. This approach acts as an incentive model for API consumers, encouraging integration and quality implementation of the resources that drive revenue for an API provider. ***3 distinct models*** for sharing API revenue with consumers have emerged:

i. **Ad revenue share.** Some API providers offer an advertising network as part of their platforms. API consumers embed advertising in their sites and apps, providing revenue for API providers. In turn, the API provider returns a portion of the revenue from advertising.
ii. **Affiliate.** Some approaches to monetization of websites have been applied to API ecosystems. Cost per acquisition (CPA), cost per click (CPC), and one-time or recurring revenue-sharing models are commonly used.
iii. **Credits to bill.** A smaller group of API providers employ a paid model for consumers. Based upon advertising revenue share or affiliate revenue, the provider will credit the API consumer's bill—reducing a developer's overhead for integration and potentially reducing the API provider's expenditure.

- **Indirect monetization**

Indirect monetization of an API isn't always about generating revenue from API access, advertising, or other revenue. There are indirect ways that an API can deliver value.

- **Marketing vehicle.** APIs can serve as a marketing vehicle for a company and its online presence. Through sensible branding strategies, developers can become third-party marketing agents, working on behalf of a core company and its brand.
- **Brand awareness.** As a new tool in an overall marketing and branding strategy, an API can provide a type of brand exposure via third-party websites and applications, thereby extending the reach of a brand using third-party API consumers as the engine.
- **Content acquisition.** Not all APIs are about delivering content, data, and other resources to consumers. APIs often allow for writing, updating, accessing, and deleting

content. Content acquisition via API can be a great way to build value within a company and its platform.

- **SaaS.** Software-as-a-Service (SaaS) has become a common approach to selling software online to consumers and businesses. Oftentimes an API will complement the core software and its offering, providing value to SaaS users. API access is often included as part of a core SaaS platform, but also can be delivered as an option for premium SaaS users.
- **Traffic generation.** APIs can also be used to drive traffic to an existing website or application. Designing an API to use hyperlinks directed at central websites or apps—and encouraging consumers to build their own websites and apps that are integrated with the API—provides a great opportunity for increasing traffic.

Many companies start by focusing on launching and evolving their API strategy and gaining essential experience before fully executing on their API monetization strategy, relying completely on indirect value from an API.

While it is better to have a monetization strategy in place early, many are finding success by prioritizing the API first and monetization second. With APIs being deployed in various capacities—within a company, privately between partners, or in the public—a wide mix of monetization strategies can be used.

Some API resources lend themselves better to a pay-as-you-go model, while some markets demand that data be freely accessible without the need to register or be charged for access. There is no one-size-fits-all approach to API monetization.

One way to think about an API is as an external research and development lab within a company. A lab that accepts ideas and integrations from partners then incubates these ideas, applications, and business relationships. Companies are using APIs to allow the introduction of outside ideas and talent in hopes of inciting innovation.

Some API providers will hand-select the best integrations and invest in individuals and companies, sometimes resulting in acquisitions of companies and technology.

Just like there is no one-size-fits-all approach to API monetization, there are few constants in pricing or access. Successful API providers are constantly adjusting, tweaking, and experimenting, trying to find the most competitive approach possible. APIs are about business development and finding new ways to monetize your new and existing resources.

API opportunities

Beyond these trends in API usage, there are several opportunities emerging for both businesses and developers to take advantage of. The simplistic, web-based approach of providing access to resources is moving into emerging areas, like:

- **3D printing.** With the help of web APIs, 3D printing is moving beyond hobby and art. And it has serious potential for redefining the global manufacturing landscape. There are several platforms focusing on providing APIs for 3D printing.
- **Automobiles.** Major automakers, like Ford and GM, are turning vehicles into API platforms, creating opportunities for businesses and developers to provide new products and services in-vehicle.
- **Home.** Devices play a central role in our daily lives. We carry smart devices from home to work, and everywhere else. APIs have made their way beyond our computers and integrate directly with our homes. The next generation of home automation technology is being developed, ranging from thermostats for heating and air conditioning to lighting and home security. While much home automation technology is still without APIs, many providers are introducing developer ecosystems and are using APIs to stimulate innovation around home technology integration.
- **Buildings.** Many buildings already have automated heating, air, electrical, water, and other systems. Building equipment manufacturers are quickly seeing the importance of allowing API access to their hardware and software. Imagine if were able to teach us to reduce energy consumption, diagnose problems in real time, and self-correct when necessary or call a service provider when repairs or tuning are required.
- **Quantified self.** Devices that allow the wearer to understand more about themselves are ubiquitous. Sports and fitness-realted personal activity measurement devices are common among quantified self devices. They can be used for everything from lifestyle tracking to healthcare.

App Monetization Strategies

1. In-app Ads

In-app ads are one of the most common forms of app monetization. They're also one of the easiest to set up, which is why they're so popular with app developers. Basically, you embed ads into your app content and then get paid every time someone clicks on an ad. The most common type of in-app ad is the banner ad, which is a small rectangular ad that appears at the top or bottom of the screen.

Banner ads can be effective if they're well-designed and placed in strategic locations. But they can also be annoying and intrusive, which can lead to users uninstalling your app. If you do decide to go with banner ads, make sure you test different placements and designs to find what works best for your app and your users.

2. Freemium Upgrades

The freemium model is one of the most popular ways to monetize an app. Basically, you offer a basic version of your app for free, and then charge users for premium features or upgrades. This can be a great way to get users hooked on your app and then make money off of them later. It's also a good way to get users to try out your app before they commit to paying for it.

However, the freemium model can be tricky to pull off. If you don't offer enough value in the free version of your app, users will simply move on and never come back. And if you don't upsell users effectively, you won't make much money from the premium features. You need to strike a balance between offering enough value to keep users hooked and providing enough incentive for them to upgrade to the paid version.

3. Paid App Downloads

Paid app downloads are another monetization strategy. Basically, you charge users a one-time fee to download and use your app. This is a great way to monetize an app if you have a loyal following or if your app is high quality and provides a lot of value.

The downside of paid apps is that they can be difficult to market effectively. You need to convince users that your app is worth the price, which can be a challenge. According to Statista, the average price for an app on the Apple App Store is $0.80, meaning consumers are not expecting to spend much, if anything, on the apps they download.
Additionally, once someone pays to download the app, they're less likely to upgrade to a premium version or make in-app purchases. So you need to make sure your app is worth the price tag from the start.

4. **Sponsorships and Partnerships**

Sponsorships and partnerships are another way to monetize an app. Basically, you find a company that's willing to pay you to promote their product or service within your app. This can be a great way to monetize an app if you have a large and engaged user base.

The downside of sponsorships and partnerships is that they can be difficult to find. And, once you do find a sponsor, you're usually locked into a contract for a certain period of time. So it's important to make sure you find a sponsor that's a good fit for your app and your users.

5. **Premium Subscriptions**

Premium subscriptions are another popular monetization strategy. Basically, you offer a basic version of your app for free, and then charge users for premium content or access. This can be a great way to monetize an app if you have a lot of valuable content.

The difference between Premium Subscriptions and the Freemium model is that Premium Subscriptions typically refer to content, such as videos, articles, etc., while Freemium tends to refer to features or upgrades (such as resources or levels in mobile games).

The challenge with Premium Subscriptions is that you need to strike that balance between offering enough engaging content for free and putting appealing content behind the paywall. If users don't see enough value in your free content, they will leave and never come back. If they don't see the value in the paid content, they won't bother subscribing and you won't make any revenue.

6. **In-App Purchases**

In-app purchases are a great way to monetize an app if you have a lot of valuable content or features. Basically, you offer users the ability to buy virtual goods or services within your app. This can be a great way to make money off of users who are already using and engaged with your app.

The downside of in-app purchases is that they can be difficult to market effectively. You need to convince users that the content or features you're offering are worth the price, which can be a challenge. Once a user has paid for something in an app, their loyalty to that app is likely to be much greater. So give them a reason to make that first purchase!

7. **Location-Based Services**

Location-based services are another way to monetize an app. Basically, you charge users for access to content or features that are based on their location. This can be a great way to monetize an app if you have a lot of valuable *localized* content or features.

The downside of location-based services is that they require you to have content for as many areas that your app services as possible. These services can be a challenge to scale, and may require paying users themselves to create the content for you.

8. User Data Mining

User data mining is another way to monetize an app. Basically, you collect data about your users and then sell it to third-party companies. This can be a great way to monetize an app if you have a lot of valuable user data.

The downside of user data mining is that it can be difficult to do effectively. You need to make sure you're collecting the right data and that you're doing it in a way that's compliant with privacy laws. Additionally, users may be less likely to trust your app if they know you're selling their data.

9. Physical Goods and Services

Selling physical goods and services is another way to monetize your app. Basically, you sell physical goods or services through your app. This can be a great way to monetize an app if you have loyal users that enjoy sharing your brand or content.

The downside of physical goods and services is that they may require you to have a lot of inventory on hand, or find a vendor to handle production on demand. Managing the logistics of selling physical goods adds a significant layer of complexity to operating your app. On the other hand, some physical goods – such as branded shirts, hats, stickers, etc. – can be a great way of turning your users into marketers!

10. Crowdfunding

Crowdfunding is another way to monetize your mobile app. Basically, you offer users the ability to donate money to support your app. This can be a great way to monetize an app if you have a loyal following that believes in your mission.

The downside of crowdfunding is that it requires you to have a large and engaged user base. If you don't have a lot of users, or if your users aren't particularly engaged, it will be difficult to raise money through crowdfunding. Additionally, you may need to offer rewards or perks to users who donate, which can add to the cost of running your app.

Common mobile ad formats

Ad format can have a noticeable impact on your app's user experience. Picking the right ad format is very important for ensuring users aren't turned off from your app.

Banner Ads
Banner ads are the most common type of mobile ad. They are typically small rectangular ads that appear at the top or bottom of the screen.

Banner ads can be effective, but they are also the most intrusive type of mobile ad. As a result, they can have a negative impact on user experience.

Interstitial Ads
Interstitial ads are full-screen ads that appear between app content. They can be effective, but they are also very intrusive.

Video Ads
Video ads are becoming more common as mobile devices become more powerful. Video ads can be effective, but they require a significant amount of bandwidth and can have a negative impact on user experience.

Rewarded Video Ads
Rewarded video ads are a type of video ad that allows users to watch an ad in exchange for a reward. This reward can be in-app currency, access to premium content, or other perks.

Rewarded video ads are less intrusive than other forms of video ad, and they can be more effective because users are voluntarily watching the ad.

Update and patch

An update to an application can be provided to users as a Windows Installer patch package.

A patch can contain an entire file or only the file bits necessary to update part of a file.

A patch is a software update comprised code inserted (or patched) into the code of an executable program. Typically, a patch is installed into an existing software program. Patches are often

temporary fixes between full releases of a software package. Patches may do any of the following:

- Fix a software bug
- Install new drivers
- Address new security vulnerabilities
- Address software stability issues
- Upgrade the software

Type of update	**Productcode**	**ProductVersion**	**Description**
Small Update	No change	No change	An update to one or two files that is too small to warrant changing the **ProductVersion**. The package code in the **Revision Number Summary** Property does change. Can be shipped as a full installation package or as a patch package.
Minor Upgrade	No change	Changed	A small update making changes significant enough to warrant changing the **ProductVersion** property. Can be shipped as a full installation package or as a patch package.
Major Upgrades	Changed	Changed	A comprehensive update of the product warranting a change in the **ProductCode** property. Shipped as a patch package or as a full product installation package.

Difference between a Patch and an Upgrade

Key Difference: **An update or a patch is a very small snippet of code or multiple snippets of code that are packaged and sent to the user. On the other hand, an upgrade is something that includes a huge change and is a complete version change that at times requires payment.**

Comparison between a Patch and an Upgrade:

	Patch	**Upgrade**
Definition	A patch is a small snippet of code or multiple small snippets of code that are meant to address some issues, fix some vulnerabilities or activate certain functionalities	This is a large set of code or a brand new version of the software. It is often released as a more superior version of the same product
Installation	Installs over the already present software, without requiring any installation to the current version	Usually uninstalls the older version before it installs the later version
File Size	Smaller in size	Often larger in size
Cost	Free	Is often priced
Intensity	Low intensity, no major changes	High intensity, major changes
Overall Look and Feel	No huge changes to look and feel of the software	Sometimes huge changes to the look and feel of the software
Frequency	More frequent depending on the software	Less frequent compared to patches