



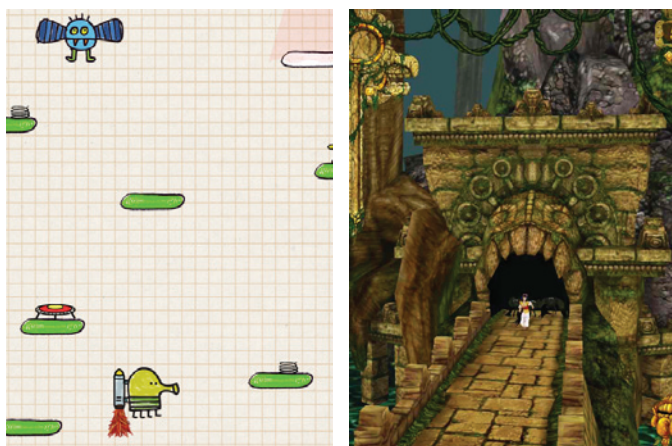


1.0 Introduction

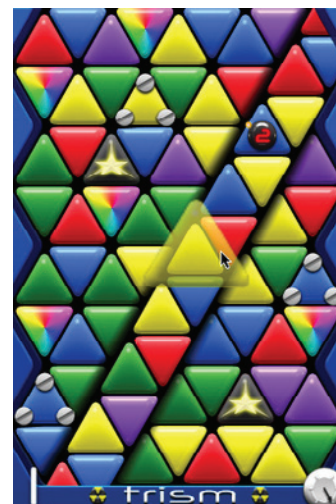
From Angry Birds to Doodle Jump, many millionaire app developers started from their bedrooms, with little money, and only a rough idea of what they wanted to achieve. This unit is about helping you to understand the market for mobile applications. You'll discover how to take advantage of this fledgling software business, whether you have your sights set on the Google Play, or the Apple App Store.

In the Apple App Store, over 25 billion apps have been downloaded since the introduction of the iPhone in January 2007. It is estimated that there are over one billion downloads from the Google Play every month.

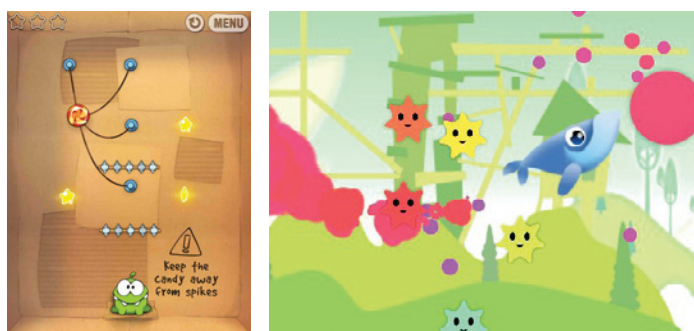
Making an app can be an extremely enjoyable and rewarding experience and has the potential to evolve into a profitable business. Developers like Steve Demeter, who created the hit app *Trism* in his bedroom, made over \$250,000 in just two months – he continues to profit significantly from his apps. The two brothers who created Doodle Jump in their spare time have sold over 10 million copies of their app.



Then there are the more recent indie hits, like the husband-and-wife team who have racked-up an incredible 40 million downloads of *Temple Run*. However, app development is a tricky business, and success is far from guaranteed. The trick is to invest your time and money wisely. A very basic app, like *Trism*, will take around a month to develop, and cost around £5,000 (\$8,000) to build – assuming you are hiring in programming expertise.



A more complicated app, similar to *Angry Birds*, will cost around £50,000 to create, and something



as sophisticated as *Cut The Rope* or *Whale Trail*, £200,000 or more. These might seem like large sums, but you can reduce the cost considerably by sticking to a simple yet effective idea for your app. Don't forget that *Tetris*, which today would cost just a few hundred pounds to code, is one of the most successful games of all time.

1.1 App design basics

Coming up with an idea

Everything stems from this moment of inspiration. The biggest problem you'll have at this stage is originality. You'll quickly discover that 99 per cent of the ideas you have for a mobile app, no matter how wacky, have already been done. It's astounding, but go on, think of a few mobile app ideas and Google them.

An app that dares you to throw your mobile phone as high in the air as possible? Done.

An app that encourages you to lick your mobile phone screen and watch a graphical ice cream disappear? Done. An app that measures the speed and power of your punch? Done. Done. Done.

Okay, what now? You have two options: keep thinking up ideas until you strike a new one, or develop an app similar to an existing one, but make it better. Both options have benefits and pitfalls.

If you choose to improve an existing app, you may find that you're the underdog competing against the existing standard. But at least you know there's a market for it. If you choose to persist with a brand-new idea, then you're dealing with a complete unknown: *Will people like your game format?* We'll cover this problem in more detail in Unit 2.

Important first steps

The best way to develop your app is not to hire a programmer at all until the design and graphics are finalised. This will avoid unnecessary expense.

Don't start any coding work until you know how every tiny bit of your app will look graphically. Having a clear idea from the start about what you're building, and how you want everything to look, will help you save weeks of painful re-writing later on.

Now is also a really good time to reserve your app's name. You can do this by going through the app-submission process, but without uploading the 'binary' - the name programmers give to the application code itself. This process has been likened to domain-name squatting, but it's quickly become apparent this is the only way you can use the name you want. If you don't register the name early enough, someone else might, and you risk losing it in a cyberspace ether from where it cannot be retrieved - there's no way of contacting the person who's nabbed your awesome moniker.

If you're doing something unique or technically challenging with a mobile app, it's worth hiring a programmer to build a crude prototype to test its feasibility. We discuss the process of finding

a programmer in Unit 5.

1.2 Differences between Android, iPhone and iPad

The debate over which platform offers the best return rumbles on. You'll find all kinds of weird and wonderful theories on the subject. The users of each device differ socio-demographically, however.

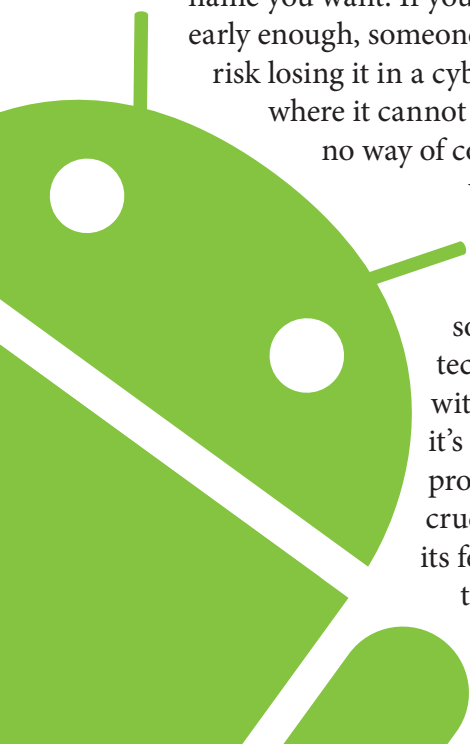
For instance, companies like Rovio, makers of *Angry Birds*, have found that iOS users are more likely to pay for *Angry Birds*, but that Rovio can still make a profit from Android users by giving the app away for free and using in-app advertising to offset this loss.

As for pricing your app, or making it free, the bottom line is, to quote William Goldman, 'nobody knows anything'. We discuss pricing issues in more detail later in the course. Check out Unit 17.

If you price your app low, it'll be more of an impulse buy. But price it too low and you might give the impression of a low-quality product. If you price it high and you might sell far fewer copies, but you might make more profit overall from the copies you do sell. Or you might not sell any, because it's too expensive.

A similar dilemma hangs over 'lite' versions of apps, like *Alice for the iPad Lite*. These are stripped-down versions of your app, designed to give potential customers a free taste of what you're offering, and leaving them hungry for more. Although they're a brilliant marketing tool, you have to strike a careful balance between how much you give away for free, and what you hold back to entice a full purchase.

Hold back too much and you won't be able to create a good impression with the lite version. Give away too much, and there's no incentive for anyone to buy the full version of the software - they're already satisfied. This is a curious dilemma and we'll return to it in later units.



1.3 Choosing a category of App

The best policy is to ignore the statistics and just build an app you love. When a product is built from the heart, it's usually the case that care, attention and love have been lavished on it – and that tends to lead to sales. However, the more shrewd designer can use the resources at App Annie (appannie.com) and Flurry (flurry.com) to assess which app categories



are the most profitable. This varies from month to month, and don't forget: most users do not discover apps via the App Store or Google

Play, which are both something of a maze to explore. Most users hear about your app through word-of-mouth or press coverage, about which you'll learn more in Unit 18.

1.4 What to expect when you launch your app

You've tested the app, your friends have tested the app, your parents have tested the app, your hairdresser has tested the app, and everyone thinks it's great. Time to let the rest of the world have a go.

In order to get your app on to the App store or Google Play, there's some mystical hand-waving to digitally sign the app for distribution. Once the gods have been appeased, you can upload the application to Apple or Google for approval. As far as we can tell, this process on the Apple App Store is taking about 10-14 days for most apps, but some designers report mysteriously long waits to get their app approved.

If it's the iPhone you're designing for, you're now in the Kingdom Of Apple. The only communication you can submit is via email. Gallons of patience, and trying to find answers from other sources such as Internet forums, are key to getting through this stage.

Google are more liberal with the entry requirements for the Google Play, and there is a less formal app-approval process. However, Google are progressively becoming more Apple-like in their approach to app submission and are tightening-up store guidelines. Gone are the days when anything went on the Google Play (now Google Play). At the end of 2011, Google removed 22 'malicious applications' from Google Play after they were tipped off about apps attempting to trick users into accepting fraudulent SMS charges. Google is also not above pulling apps for purely editorial reasons. The developer of *Reddit is Fun* found his app deleted from Google Play because it contained, in Google's words "sexually explicit material". The only problem was, the *Reddit is Fun* app did not actually contain any explicit material, it simply allowed access to the Reddit community website.

As Google seeks to share in the kind of massive success that Apple has seen with its store, it has become increasingly less-tolerant of any apps that might mire the company in controversy. If you're thinking of picking Google Play because the submission process is more tolerant, be aware that times are changing. Having said that, Android is different from iOS in that users can 'side-load' apps – this means that users can install apps from any source. Android apps do not need to be officially sanctioned by Google. But, in practice, very few users will have any idea how to enable this feature on their phone, let alone find your app outside the store, or trust it enough to download and install it. Side-loading is great in theory, but the reality is you'll be invisible to the majority of Android customers.

Along with the approval application, you can also submit screenshots and descriptive text for your app. These are all anyone will see when browsing through apps on the App Store or Google Play, so make sure to make them clear, descriptive, and enticing. You'll eventually get a number of 'free download' codes you can give out to reviewers and influential websites so they can test out your app. It's worth using these wisely to get the word out about your creation.

You'll learn about this process in more detail as you work through the units in this course. Good luck, and welcome to the exciting world of app design!