A Step by Step Guide to Creating an App

by Carter Thomas, Bluecloud LLC

Disclaimer

Table of Contents

The Idea

> "Carter, love your blog post. I've got an idea that I think is going to be the next Angry Birds but don't know where to begin. HELP!!" - **_Jim_**, _Seattle_

If this sounds familiar, you're in the right spot. The most important part of creativity is coming up with ideas and apps are no exception. Whether it be from seeing apps that are already in the store, thinking of a good solution to a problem you face, or simply thinking something would be cool are all **great** reasons to develop an app.

Being passionate about your idea is going to carry you through the process. If you're not passionate, you're going to burn out.

As a marketer and businessmen, I look at all ideas in the app world through the lens of "[how is this going to make money.](#)" There are very few people out there that want to spend a lot of their own cash on something that's just cool, and this isn't for them. This is about making sure your idea is not only good, but will win in the app market.

In traditional business models, there are two general areas of success:

a.   Developing something that no one else has
b.   Beating your competition by being better than them

When you're coming up with your idea, ask yourself these two basic questions. My own experience is that it's going to be very hard to come up with an idea that is completely unique (other than games) that can guarantee interest. It's possible, but very hard to do unless you have a big budget. **My advice: focus on an existing model and make it better.**

As you think about your idea and discuss it with others, it is very important to make sure it's bulletproof from the start.

"How many levels are you going to have?"
"What's the design going to be like?"

And most importantly:

"What's special about YOUR app?"

On this note, I will leave the rest up to you and your devices, but remember that an app needs to be special in some way. This can come through the following:

1. **Unique functionality** - "The screen is going to use a swipe instead of a tap"
2. **Unique design** - "I'm going to create a cartoon version of the weather app."
3. **An awesome story** - "This app is going to catalog a personal or non-personal story from my life that I can share with people."
4. **Marketing/Distribution** - "A company with 2 million Facebook fans signed a contract saying they will promote it for 6 months."

The following are NOT good reasons to embellish an idea:

1. **Copycat approach** - "I could DEFINITELY make a stick figure running game."
2. **Greed** - "I just want to make money and this is going to get me there."
3. **Glamor** - "This idea is just so sick, dude."

And so on.

Come up with an idea that means something to you and your chances of success skyrocket.

Everyone's got GOOD ideas, few people have GREAT ideas. Be very honest about where you stand and, when you've got something that's truly awesome, begin the process.

Vision to Paper

"A really great talent finds its happiness in execution." - Johann Wolfgang von Goethe

You've got your idea. Yes! Awesome. You've gone through as many of the different scenarios as you can in your head to find holes in the business logic. You've got a great story to tell and an incredible vision to paint. Let's go hire a developer and rock and roll!

Not so fast.

Before you walk into any developer or designer shops, you need to take everything in your head and put it down on paper.

Here are a few examples of the first steps:
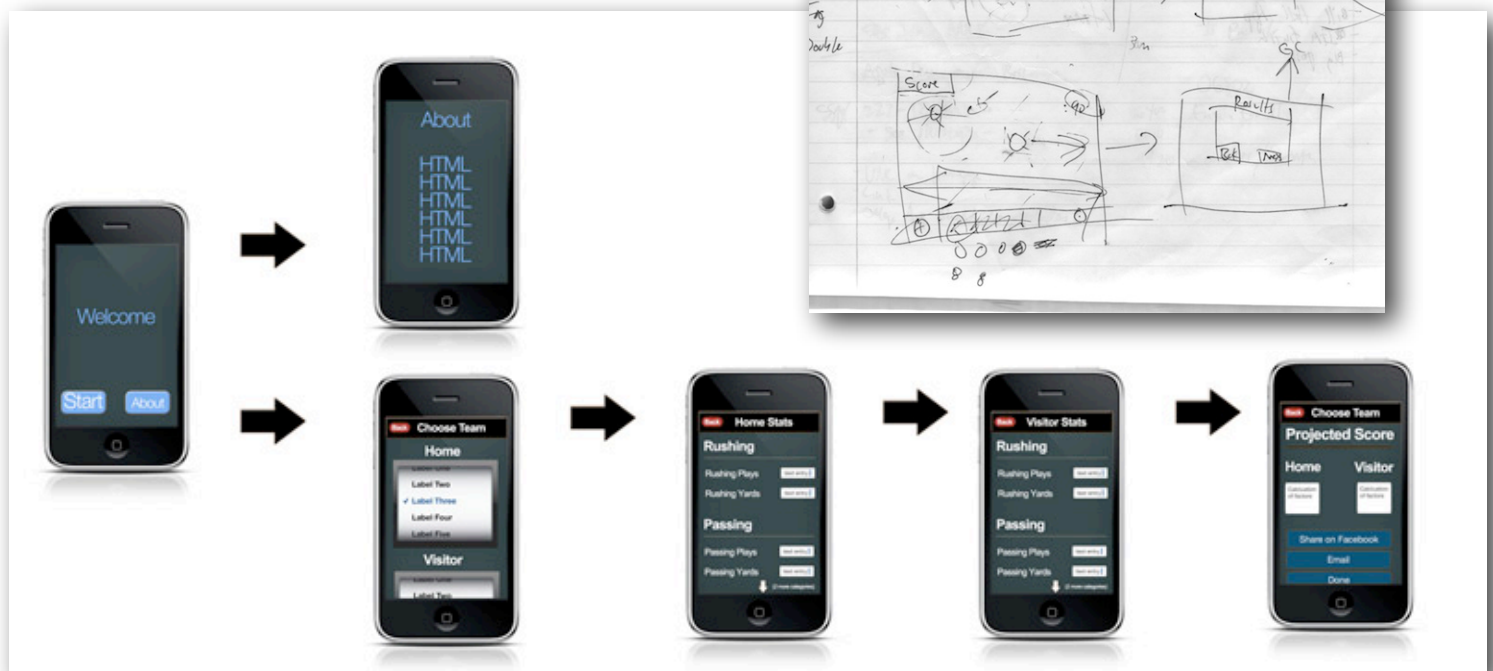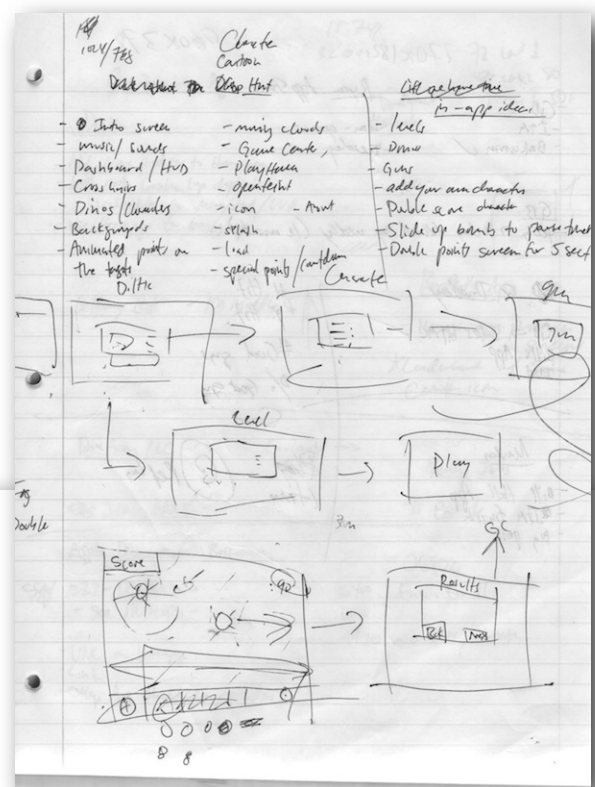
Clearly these are only very, very rough drafts of first thoughts, but you get the idea. They begin the process of laying out your idea so that you can illustrate what's in your head through a logical and meaningful way.

Bluecloud Tip: *Remember who you're dealing with - developers and designers. These types love detail and love direction, so give it to them. Be an overwhelming wealth of information and they'll love you for it, both in their work and in how little they charge you.*

I would recommend building something along the lines of a **hierarchy** - the top being the opening screen and then every path representing a possible path for the user to take. Take into account as many different scenarios as possible and give each an "if -> then" type of logic. Make every "what if they do this?" question be simply your pointing to a spot on the hierarchy.

Similarly, **do not feel like you need to do this all on your own**. Many of us don't think this way and have a very hard time conceptualizing large ideas into very small parts. You can go a developer with more of a the broad strokes and they'll walk you through as much as they can. BE WARNED - this is usually going to take on a Q&A type of relationship with the developers/designers firing questions at you left and right.

Essentially they're going to build that hierarchy for you, but will do so by asking you "so what do you want it to do here?" instead of you doing this all yourself. It's up to you, but you can imagine which costs more :-)

If you are looking for a developer to "run with the idea" and just implement whatever best app practices are or whatever they think would work well, you're in for a disappointment. THIS IS YOUR APP AND YOUR DREAM! Take the time to go through the details and don't let anyone else fill in the gaps for you. This is especially true if you're working with a firm overseas, which I will get into later.

Bottom Line: Map out your ideas with a painful level of detail - you won't regret it. It's not sexy, it's business.

Steal This App - NDAs and Intellectual Property Issues

"From the business point of view - not to overstate it - intellectual property is dead; long live intellectual process. Long live service; long live performance." - Anonymous

Beyond the 20-30 emails I get a day from people looking for app quotes, I would say the second most often question I'm asked is **"How do I protect my idea?"**

Let me start by saying I understand that position - I own a company that is literally a single logo and I went through about $20,000 in trademarking through the USPTO, and I don't regret that for a second. The value of my brand and product is worth about $100,000 right now....all because no one else can use that mark easily. Owning intellectual property and an idea can be a great way to skyrocket your value, along with ensuring that you're first to market and inhibiting competition.

But the app world is a very different beast than the patent and trademark world. I hate to break it to you, but unless your app has to do with something very specific to you or and proprietary idea already, chances are someone's already developed something like it, or will develop something like it in the near future. Honestly, that's what I think is so great about the app world - it's a free market. I always get skeptical about people who are so insecure about "sharing their ideas" because I think they don't believe that they could bury anyone who tried to steal it. **Without that killer instinct, how do you expect to be a millionaire?**

I digress. My point is that there are lots of ways to spend your time, energy, and money and protecting an app idea should not be one of them. For those of you who are concerned about this, here's what you need to know.

Any development shop under the sun is going to to either assume they are under NDA or will readily sign whatever you'd like before any negotiations take place. Any developer

that has a business built on stealing ideas or source code is going to fail, I can guarantee you that, so don't think about it too much.

Similarly, these guys see hundreds of ideas every month, so the chances of them stopping that all to take your source code and build their own app is very, very unlikely.

Designers will also be happy to sign an NDA before doing any work, protecting all your designs and characters, etc from making their way into something else. Without question, the teams that you hire to to make this app a reality are the last people you need to worry about with intellectual property, so rest easy.

> *"I don't want someone ELSE to steal my idea. I know everyone's looking for a good app idea."*

You're right. A good idea for an app is a huge leg up and can get the juices rolling. If you want to protect it completely, your best bet is to just have NDAs signed by anyone you talk to on a formal level. It may be kind of a turn off if you're not paying them, but it usually shouldn't be an issue if you do all the paperwork and heavy lifting.

On a personal note, I've seen some people who "steal" ideas from others and try to build the app themselves. It happens. What I will say is that the people who steal ideas tend to be the guys doing black hat SEO and looking for any sort of "get rich quick" scheme. They're not going to drop $50K on an idea they stole from someone else, trust me. And even if they do, you'll crush them in the long run because you're going to want it more.

So - for intellectual property, don't sweat it too much. Anyone you hire will be sensitive to protecting your idea and you wont have it stolen.

Now, let's focus on the good stuff :-)

Hiring a Developer

"There are no shortcuts to victory." - Richard Lugar

I'm going to start this section with a cliché - **you get what you pay for**. The truth to this statement continues to manifest itself not only in my business but in my life. Quality costs money. It's as simple as that.

When you're hiring a developer, you've got a lot of options - a firm, an individual, a small team, domestic, overseas, etc. The hardest part can be figuring out where to start and who to trust. Let's break this down.

Finding a Developer

When you're beginning your search, you're essentially starting at zero. It's even harder when you don't have a clue what exactly you're looking to buy other than "an app." Here are a few great ways to start your developer search:

1. **Ask around** - This can be through friends, family, or co-workers OR it can be on discussion boards, websites, and review sites. Ask people for referrals and explain what you're looking for. About 4-5 people a day call me just asking for a good developer to contact. Please don't call me though, I don't have the time :-P
2. **Google** - Doing the old search engine scour is another way to do it. You'll be able to see a lot of different firms and their portfolio/service offerings. Be specific - type in your state (if you're in the USA) and iphone app developer and go through a few pages before making any calls. Ask about their process, pricing, and overall work flow. If they are willing, ask them to send you a proposal.
3. **Freelancing Sites (oDesk, eLance)** - This is a good way to get a sense of pricing right off the bat and also to get into the world of outsourcing. You can browse posted jobs and find ones that are similar to yours ("I need a simple word game") and see what people write as their job request. You can then research these firm on their profile pages or on their main webpages, and maybe even follow up with their contact form. This is going to be a bit more impersonal, so treat it as though you've got a very

clear idea and set of deliverables that you want built for you. Unless you know what you're doing with software, you will want to make sure they have some project manger dedicated to your project so that you're not stuck answering questions like "Which Xcode framework would you like this to run on - Cocos2d?" You get the drift.

No matter which path you decide to take, you should definitely take your time and find the firm that works best for you. You'll know when you find them and then you just have to go for it. Be honest about who you are and what you're looking for and they will respect that - no one is going to take advantage of you, especially in a market as competitive and global as this one.

Choosing a Developer

Now that you've got 10 proposals in front of you, what should you be looking for? How do you know which is going to deliver you what you want and which won't? Here's a side by side:

Individual Developers vs Firms

Individuals
• **Personal and collaborative** - if it's the right fit they may act more like a partner than a hired gun
• **Specific skill sets** - every developer has their own strengths. If you get one that matches exactly what you're doing, you're going to get an awesome product
• **Cheaper** - for better or for worse, you're not paying for project management and overhead.

Firms
• **Highly deliverable and time-oriented -** firms thrive on project plans and timelines. You'll have reports sent to you all the time telling you where things stand.
• **Project management -** you'll have one person who can take the lead on everything. You'll never have to speak to a programmer about what you want and will always have someone to talk to.

- **Hit By A Bus -** this means that if something comes up with any of the developers, there's a back up plan. Your project is not going to suffer.
- **Testing/QA -** in my experience, the more people working on the app, the better. This helps get the bugs out and get multiple perspectives on the project. You can be rest assured that 10 people cleaning up an app are going to comb through things better than one person.

Domestic vs Off-shore (this is mostly for USA, but can be applied to most countries)

Domestic
- **Language -** this can actually be very detrimental to a project. Not only is it annoying (sorry) but details get lost in translation. Nuances that you are used to expressing in your native language may be perceived as something completely different. Example - "I want this home screen to integrate nicely with the about screen." ---translated to--- "We need to put a button on the home screen that links to the about screen [with no regard to design or user flow]. Domestic firms are easy to communicate with.
- **Talent** - I don't mean this in the sense that the talent is better by any means, but I do find that the domestic firms' products are more technically sound and robust. Consistency may be the best word for this.
- **Accountability -** working with a firm that's in the States allows you to rest easy that you are protected by the governing body of the USA, along with the mighty power of online reviews. If something goes awry, you can go visit the firm and play on the same level as them in terms of legal action (RARELY necessary). Because this is unspoken, the quality is much higher and is basically baked into the price.
- **Good for the Country** - yeah, I'll say it. It's awesome to be able to re-invest money back into the USA if you can. I'm definitely a proponent of outsourcing overseas, but I think there is a lot to be said for developing something in your own backyard.

Off-Shore
- **Cheap -** God, is it cheap. The reasons for this are many, but the biggest being is that the labor supply is enormous. This point is not necessary a plus, but it's a fact. Another reason is true is because a lot of offshore firms will not do all the bells and whistles of a domestic firm in terms of proposals and timelines. They'll say "what do you want, we

will do that for X dollars." It's not warm and fuzzy, often doesn't look great out of the gate, but you can save some serious coin.

- **Fast -** I had a company build me a copy of a complicated content display app (see Martha Stewart's Cocktail app, will show you video later on) just to see how the process worked. I figured this would be a 4-5 month job. Wrong - I got the full Xcode project, entirely populated with dummy content, in 5 weeks (including a 6 day national holiday). If you've got a simple, clear idea of what you need done, this can be a lifesaver.
- **Very little BS -** Believe it or not, the professionalism I have seen with overseas firms is far and away better than most domestic firms. These guys are HUNGRY for business and will be knocking down your door once the project begins to get feedback and move onto next steps. You'll never have them tell you "Oh, well we've got to whiteboard this for a while" or any of that. They get things done.

Payments

Making sure you understand how the payment process works will save you a lot of headache and will also make you look like you know what you're doing. Most development firms work in phases, or milestones, which represent points where deliverables are completed and payment is sent. A typical project would be broken out like this:

a. **25% -** Initial Deposit/Kickoff
b. **25% -** Approval of Graphics/Wireframes/Lite Prototype (assuming they are designing it)
c. **25% -** Approval of Beta Version (on iPhone simulator or iPhone device if you can get to the firm's office)
d. **25% -** Approval and release into Apple Store

This can all change depending on the type of app, size of the project, and use of outside graphics. Usually this is a pretty easy negotiation and the developer will be happy to make arrangements however you'd like.

One piece that you may want to discuss with your developer is what's called a "Discovery" phase. This is typically only a realistic option if your project is going to cost more than $50,000 because otherwise it doesn't make a lot of financial sense for the firm. What this does it pays for the firm to write you a blueprint for the entire project - all the way from conceptual to tactical to marketing - and then present it to you as a way to see the entire scope of the project before investing fully. You can roll this into you final project cost or you can take the blueprint to another firm - it's totally up to you.

Developer Wrap Up

If there are a few golden rules to leave you with, here they are:

1. Get a lot of proposals from different firms and just assume that the price spread is roughly proportional to the quality that you're going to get.
2. Expect to be engaged in the project if you want it to turn out the way you want it to.
3. No developer should be able to give you a solid price until they've gotten you on the phone or been able to ask you a lot of questions. I've heard of people asking for a quote and saying "I've got a $15K budget and here's my idea" and then the developer says "Oh wow, that's amazing. That project is exactly $13,800, which fits perfectly in your budget." Code red.
4. Go with your gut. At the end of the day, all these guys can code and provide you with an app - it's the one that makes you feel like they're going to do a good job that should be getting your money.

App Costs

Development for iPhone apps deals with the programming side of the process. With most projects, an app development firm is going to roll up the design into the entire project since it's a bear to try and piecemeal images into an app unless you build the entire thing dynamic and through XML via online databases (*read* - edit images externally vs. housed inside the app). One thing to note is that these costs can vary depending on what devices you are developing for. Make sure you ask your developer what you are developing for – basic iPhone, iPhone 4, iPad or all of the above. I'll see what I can do break out typical development costs for different example projects.

- **Simple, table based app** - **$1,000-4,000** - you provide all the content, clear direction, and example apps of what you want it to do. If you know your way around Photoshop, you can probably supply the graphics which will cap this project at $1,500. The additional costs are when you want to GPS locators, Social media integration or bells and whistles like that.
- **Database App (native)** – **$8,000-$50,000** – Again, you provide every piece of content, image, writing, sound, etc. The cost is going to come from creating the logic within the app and architecting all the usability and/or game play. The content will usually be dropped in and then parsed accordingly. These projects tend to be front heavy since the data is what's driving the entire game and the framework is so important.
- **Games** – **$10,000-$250,000** – The hardest to ballpark. As a benchmark – I've heard Angry Birds cost anywhere from $125K-$180K to develop (although they were pioneers). Talking to some developers who are into the hardcore game source code (render, sound, maths, physics, etc), many of the racing games will that use the gyroscope will be $125,000 without even blinking – and that's just for the code. Even if you try to keep it hyper simple, games get complicated quickly. Hooking into game center, having top scores, and integrating with an online community can be tricky. The benefit of a game is that they download in much greater number. As a marketer, there is nothing more viral than a fun game, which is something to keep in mind for your ROI. Ask yourself: how much do I need to spend to make a "fun" game? That's only something you can answer.

- **Additional Includes – See below** - here's a quick list of additional functions you may want to add into your current app:
    - *In-App Purchasing – $1,000-$3,000* – this allows for users to buy new content or full versions of the apps. The cost spread comes from the amount of in-app purchasing, the complexity, and whether or not you build it all into the first app or if you are doing it from a server.
    - *Web Services – $1,000-$5,000* – This is taking the content to a remote access point so that you can update your app with an XML files instead of raw code changes. The degree to what you need varies but I would recommend having this conversation with your developer before getting too deep – it can save you HUGE headaches down the road.
    - *Game Center – $1,000* – Apple's done a good job at making this integration easy with the SDK. As long as you keep the numbers clean, you should be able to integrate easily.
    - *Share Capabilities – $500-$1,500* – This is mostly for social media (twitter, facebook) and emailing, but there can be other integrations. WordPress websites, for example, may be one. Lots of options and most of these platforms have robust APIs to make it work well.

Design Costs

- **iPhone only** (usually the base line is not done for the retina display, as seen in the iPhone 4 as they require higher resolution files**) – $500-10,000** – This will get you a base level design that you can send over to the developer who will be able to add these images into the code. HELPFUL TIP: ask the developer what they want the files named and do that for them before sending. It will save you both lots of time and headaches.
- **iPhone 4 Compatible – Add 25% to above cost** - these images will be sized at 960 x 640px height (twice the 480×320 for the regular iPhone)
- **iPad – Add 50% to above cost** – this is assuming that the layout is very similar and user flow is also similar. In many cases you will have to create alternate functionality to accommodate for the iPad size and speed, which requires new designs.

The icons should be included in this entire package and you can probably ask for screen shots if you want to have some that are better than doing a simple screen grab of your phone. I'll say it again – don't skimp on design….you'll have a MUCH harder time making your money back.

Extra Costs

The only other cost is going to be the $99 that Apple charges you to be a developer. They will also take 30% of everything you make, so if you're selling an app for $0.99, you're only going to get paid $0.70.

Apple pays you every month and will do a direct deposit to your bank account. They have detailed financial reports so that you can see what countries made the most money. This will all be rolled up for paid apps, in-app purchases, and iAds revenue into one payment, but can be broken down you wish.

Getting a piece of software (I use [AppViz](#)) that helps you analyze the data can be very helpful.

Hiring a Designer

"Everything is designed. Few things are designed well." - Brian Reed
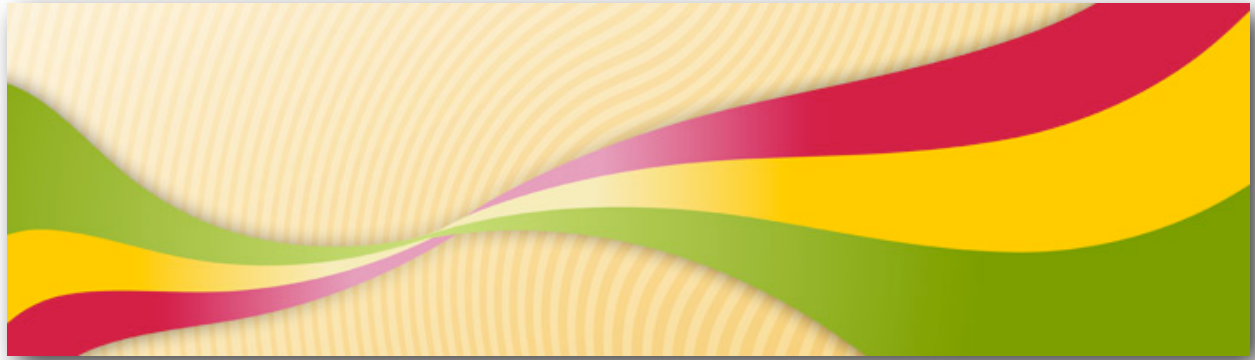
Somewhere along the way developers managed to assert themselves as the most important and crucial part to a software project, and for good reason. Functionality and bullet proof coding make a product great and will lead to success time and time again.

But we're not building some word processor for Windows which people use as a tool, we're building engagement platforms, which is how you should think of apps. **It's a way to engage with a user**. Functionality plays a part, but so does visual appeal. Delivering something that pleases the visual sense is sure fire way to create a positive experience.

When you hire a firm to develop your app, often they will have a designer in house or have some design solution for you. Let me start by saying that many times a designer that's on the team can be top notch - they know how to design for mobile and work with developers. So - if you like the designs that the firm does with their own team, awesome. You can skip the rest of this section.

If, however, you're not that impressed with a firm's design capabilities, you're going to want to hire someone to do the graphics separately. This designer will create a suite of graphics and send them over to the developers in a layered format (Photoshop, probably) who will then slice them up and add them to the appropriate areas in the software.

Luckily, hiring a designer is a lot easier than hiring a developer because you can pretty much figure out what to expect based on the portfolio. Also know that you can often hire designers who are specifically for mobile apps and ones that are not. Unless you're asking them to do User Experience design, you can hire anyone you please.

Different Design Needs

This is very important - before you get into the design side of hiring, you need to know what kind of relationship you need. There are two basic buckets here:

1. You need someone to create the **entire look/feel/theme** and everything else that comes with that (characters, buttons, fonts, etc). Also known as "branding."
2. You've got a **very specific idea** of what kind of graphics you want and can write down what you're looking for ("I need a splash screen of a fighter jet shooting rockets with the title *Alpha Combat* in big letters.")

The biggest difference is in the planning stage. A designer for the first set of items is going to have to brainstorm, do a bunch of mockups, ask you 1000 questions, and work closely with you to create the visual impact you want.

The second will say "Send me a few examples of what you like" and then they will send you their version.

The final point with design needs is what sort of project you're developing - is it a game with 50 levels or is it a simple app that displays recipes?

Hiring a Designer

Honestly, the domestic/offshore difference is much smaller for design. The only issue may be a language barrier, so if you are looking for someone to create the "essence" of your game, you're going to want someone who speaks the same language. Nowadays all

of their work is displayed in online portfolios, so even if they're in Argentina, you know how they stack up to someone in Minnesota.

I would use the same strategy to find a designer that you used to find a developer, but with more emphasis on the Ask someone and Freelance than on the search engine game. Design firms that win on search engines are actually marketing companies that do design - they do volume ("$25 logo!! $199 for a website!!) and are not going to be your best bet. I personally love perusing Elance and seeing what sort of designer talent pool is out there. Type in keywords like "cartoon designer" or "iPhone 3d designer" and take a peek at their portfolio AND their previous work history. You'll learn a lot about someone by how they did for other people on previous jobs.

So what should you look for?

1. **Style** - every designer has their own flair and you'll see it after looking through a few of their projects. If it's what you're into, get in touch with them and see if they're interested.
2. **Scope of project** - has this designer or design firm spent most of their time designing brochures? Or have they designed websites, apps, icons, and email templates? Have they done work for $300 or $15,000? Does your project look like a natural fit into their current portfolio?
3. **Credentials** - I'm usually not a fan of judging someone based on their credentials, but having a designer that is fluent in multiple platforms makes a big difference in mobile design, especially when you get into 3D images. The reason this is important is because they're going to need to deliver different pieces in different ways to the developer - the designers who openly share what software they work in will make your life easier. You want skills in Photoshop and Illustrator as a base, and then any of the other CAD or 3D frameworks if you're getting into that level.

Once you've selected a handful of designers to reach out to, you're going to want to ask them point blank if they can handle an application project. Be careful of the ones that say "Yes, no problem" without then backing it up with projects they've done already, because those are the ones that assume they can do it.

Make it very clear what set of assets you want - iPhone, iPhone retina (4, 4s) and iPad (and soon iPad 3 with even more resolution). This is going to be critical in the overall project cost and design - if they design for the regular resolution iPhone and then you say you needed it in the Retina size, it's a nightmare. Just for your information, the sizes are:

iPhone Regular - 320px wide, 480px high
iPhone Retina   - 640px wide, 960px high (double)
iPad - 1024px wide, 768px high
iPad 3 - 2048px wide, 1536px high

BLUECLOUD TIP: *Offer the designer a small commission to do a "test" run for you. This may be one character or a homepage screen. This will help you get a sense of what it's like to work with the person before going all the way. I always do this with any design project I'm working on.*

Working With Your Designer

One word - **examples**. This is, without question, the #1 best way to work with a designer. Provide them with as many examples as you can of apps or websites you like, with a detailed explanation of why you like it. Despite being widely accepted as a really creative field, designers actually tend to be very calculating people - ever notice that most computer and graphics designers love clean, simple things? That's because they take your ideas and process them into a visual asset - the creation piece is very creative, but translating ideas into assets is very much left brained and UN-creative. So, get some examples.

You should have your hierarchy available for discussion with your designer just the way you did with your developer. You're going to need them to be able to see the entire project as a whole before they can dive into any one piece. They're also going to be able to tell you how much it's going to cost and how long it's going to take based on the number of "screens" you have.

Introduce them to the development team from the start so that they can both agree on how the files will be delivered and in what format. That's going to make everyone happy and save you a lot of time down the road.

Pull the trigger and hire them and begin the process. They'll start sending you mockups and getting your feedback. Typically the first few rounds are the slowest, but once you decide on something that's working, it goes quickly.

Designer Wrap Up

When selecting a designer, a few things to remember:

1. **There are thousands of different types of designers**. Make sure you look through enough to know what sort of look and feel you want.
2. **Be patient at the beginning** - it takes a few rounds before everything syncs up from what you're thinking to what they're producing.
3. **You can always hire a different one**! If you hate the way things are going, screw it. This is a must easier transition than getting a whole new development team.
4. **Be clear on your deliverables**. Mobile design mandates files come in the formats listed above. If you don't make that clear from the start, they might not be working in the right resolution and could really hurt you.

The Development Process

"The whole difference between construction and creation is exactly this: that a thing constructed can only be loved after it is constructed; but a thing created is loved before it exists." - Charles Dickens

You're now in the game, congratulations! By going from having a dream to actually building it puts you ahead of 95% of the people out there, and for that you are destined for great things.

You've paid your deposit, the developer and development team have gotten all the information they need, and now you wait for next steps. But what exactly are they doing? How does an app REALLY get created? Well, here's a very top level view for anyone who's interested in the mechanics of this whole thing.

Xcode

I've mentioned this a few times now and for those of you who don't know what I'm talking about, I apologize. Xcode is the software that developers use to build apps in. It's what Safari is for web browsing and what Mail is for email. It's a piece of Apple software that allows you to write commands that interact with each other and perform actions.
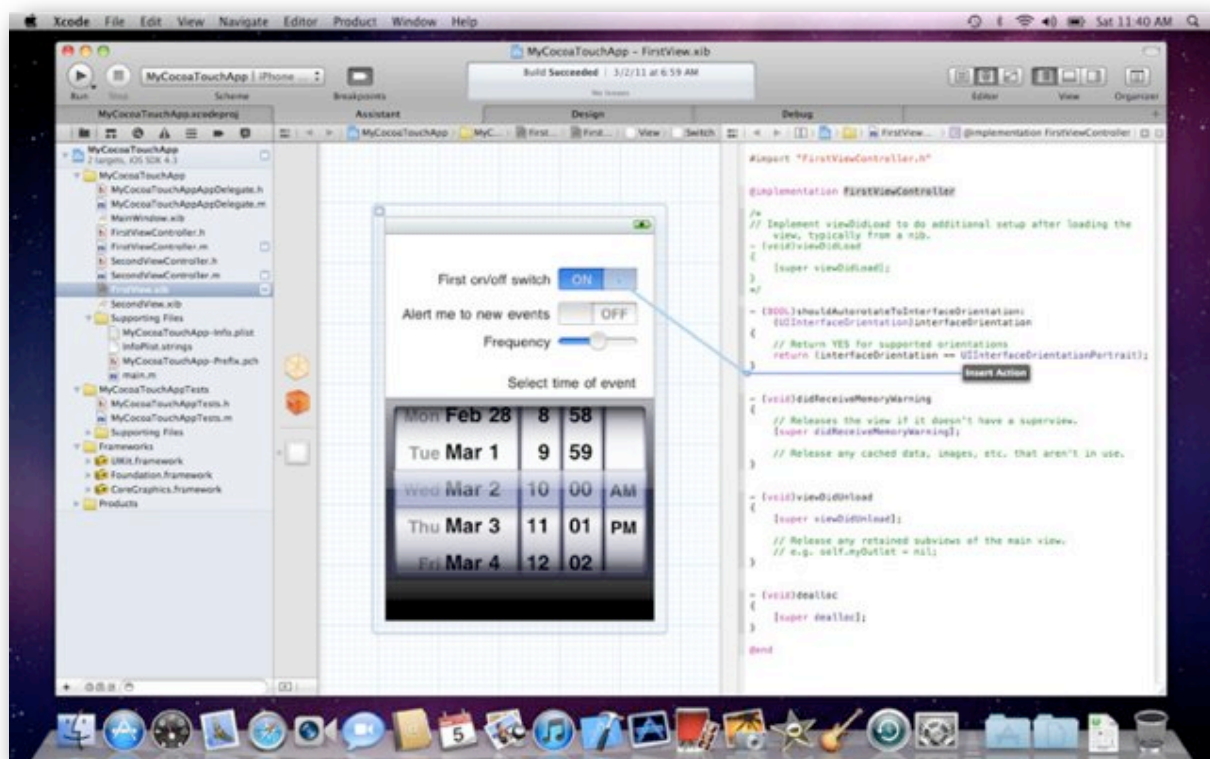
The developers go into Xcode and start building out the architecture of your app - this is why you spent all that time on your hierarchy at the beginning. On the left side of the above diagram there are all the files to edit, which are batched into different screens and functions. This may look like homescreen.m or splashscreen.m and so on. The middle screen is what's called XIB file view (pronounced zib) which is kind of like a design viewer and is drag and drop. The right area is the raw editor where you can begin to associate different parts of the app to different actions (Touch this button, go here).

The developers will use what are called **Frameworks** within Xcode to do a lot of their core function building. These frameworks are pre-packaged sets of code that Apple gives to all developers so that the heavy lifting is already done and the developers can focus on the customization stuff. Steve Jobs was a really nice guy :-)

Frameworks will be things like map functionality, OpenGL (graphics rendering), Address book, etc. The developers are going to import all the frameworks they need and start building the foundation of what your app is going to do. Once they have those "imported" into the areas they want, they start writing the custom code to do things that you want to happen.

Xcode provides a very robust environment but is wildly technical for anyone who's not used to it. The funny thing is that the Xcode 4 release is a cakewalk compared to the early days, so ask any developer and they'll say it's so user friendly now it's ridiculous. Weird.

The good news is that you don't really need to worry about Xcode until the very end, and that's only if you want to. By having Xcode installed on your Mac (must be a Mac), you'll be able to download the project when the developers complete it and run it on a virtual iPhone simulator on your computer. Do yourself a favor, though, and please don't touch the code unless you know what you're doing.

Most people will never see this software and that's fine - the developers should and can take care of everything.

The process continues and the developers begin importing all the design files from…..

Photoshop

Some people prefer Illustrator or another design program, but I use Photoshop. At the end of the day, it's all pretty similar and exports the same files.

You designer is going to be hard at work creating layered files that get sliced up and send over to the developer for implementation. Photoshop is going to allow for the designer to create buttons, graphics, backgrounds, and everything in between. If they're designing characters or planes, etc, they'll probably use Illustrator.

Photoshop is going to provide the environment where all the design happens, then sent over to Xcode. This means that the design files are edited independently from the development code and then re-added to the images directory. This provides you with SOME control if you really want it in terms being able to update images without changing any of the code.

Your designer will be compiling a set of files that will have all the master design files for your app which can be edited in Photoshop at any point after the project completes.

Creating a Prototype

One of the first milestones will probably be to create a wireframe and a prototype. The wireframe is essentially the design, which is straightforward. A prototype is where it gets interesting because you start to see how things interact with the user and realizing that you need a "back" button here or a "done" button there.

When I had that cocktail app copy developed, the firm would send me videos of the prototype on the iPhone and iPad simulators so that I could the thing in action.

Check out the video that I received for the first "prototype" build on iPhone (this will take you out of this PDF and to the movie): <u>CLICK THE SCREEN</u>

Once you see the app live and on a simulator you're able to see where any disconnect happened between you and the developers. Obviously when I saw this, I had mixed reactions - for one, the design looks awful. Secondly, the functionality was pretty much exactly what I asked for, but without that "magic" that the original (ahem, $250K app) had. I quickly realized that I was going to be getting an app that did exactly what the other one did, but without the attention to detail. A great lesson.

Different development firms may have different ways of showing you their prototype, but I think this is a great method - anything that shows a virtual user experience in action is better than screen shots. There is a distinct flow to each app and you need to see it live before you know what that is.

If you're working with a firm that's local or where you can go in an be in person, you'll be able to have some feedback loops in real time, and they may even be able to load a test version onto your phone for you to take home. If you're hiring a firm that's overseas (like I did for this) they will send you this video and include a note that says "Please let us know your feedback" and it's up to you.

The most important thing is to make sure everything is in there and that it's interacting correctly. The design can be modified easily down the road (most of the time), but the core functionality needs to be in place from the start. Review your original hierarchy and proposal and make sure everything you want is in there.

BLUECLOUD TIP: *Don't be afraid to ask for anything new once you see the prototype. If you think that there is something that adds to the user experience, just ask. The developers will tell you whether or not its feasible within the project you signed up for. It's hard to foresee these types of additions until you see a live app.*

Creating the Beta Version

Once all your feedback has gone in and you've gone through another set of Q&A, the development and design teams will go back and make all the necessary updates and changes. This process will probably take half the time, assuming nothing was way off,

and will be handled in a similar fashion. Here's the video I received back from the team after all the edits were in place: <u>CLICK THE SCREEN</u>

HUGE improvement! This made me realize the importance of taking control over the outcome of this app.

If you want your app to be awesome, make it awesome.

This means give tons of feedback and don't settle for something that isn't exactly what you want. The developers can build it, the designers can design it - you just need to keep working at it until you get it out of them.

Once the beta version was built and I was able to walk through it all, the team sent over the Xcode project for me to test on my own computer and device. **Please note** *** this is not standard practice and most developers will not send you the project because you can very easily mess it up.

Testing Your App

You probably won't need to do too much of this on your own since the developers should be responsible for all testing, but here's a quick check list you're going to want to run through before giving the go-ahead.

1. **In-App Purchases** - for goodness sake, this is how you're going to make money! Make sure this is working correctly and logging all the correct actions (note - you can only test in-app purchases on the physical phone, not the simulator)
2. **GameCenter** - if you have a game and you built in the GameCenter functionality, make sure that it's working correctly and leaderboards and/or achievements are logging.
3. **Different Phones** - Sure it works on your hot new iPhone 4S, but how about that old iPhone 3G?
4. **Different iOS Versions** - believe it or not, there are some people that don't update their software regularly. Make sure everything's compatible across the board.
5. **Bug Crashes** - this is hard to pin point, but if you stress an app enough, you're going to find out if there is a bug that crashes the app
6. **Everything else** - you get the picture.

At this point, you've done all your testing, you love how your app came together and there is only one thing left to do - upload it!

And so begins the notorious iPhone Application Publishing process....

iPhone Application Publishing

"Winning isn't always finishing first. Sometimes winning is just finishing." -
Manuel Diotte

Publishing refers to the process that takes the completed Xcode project and moves it into the Apple store. It has it's own name because, honestly, it's an entire process in itself, even for seasoned veterans. There are companies whose entire business is publishing apps for people - Chillingo is actually considered a "publisher" but has turned into a marketing juggernaut in the process.

To publish an app, you need to interface with the iTunes Connect portal - this is going to be your mission control moving forward. iTunes Connect allows you to manage the following:

- **Sales and Trends** - All the revenue and download data
- **Contracts, Tax, and Banking** - Legal information
- **Payments and Financial Reports** - Shows payouts and the reporting behind them
- **Manage Users** - Test users and developers who want access
- **Manage Your Applications** - Data like icon, screen shot, description, pricing
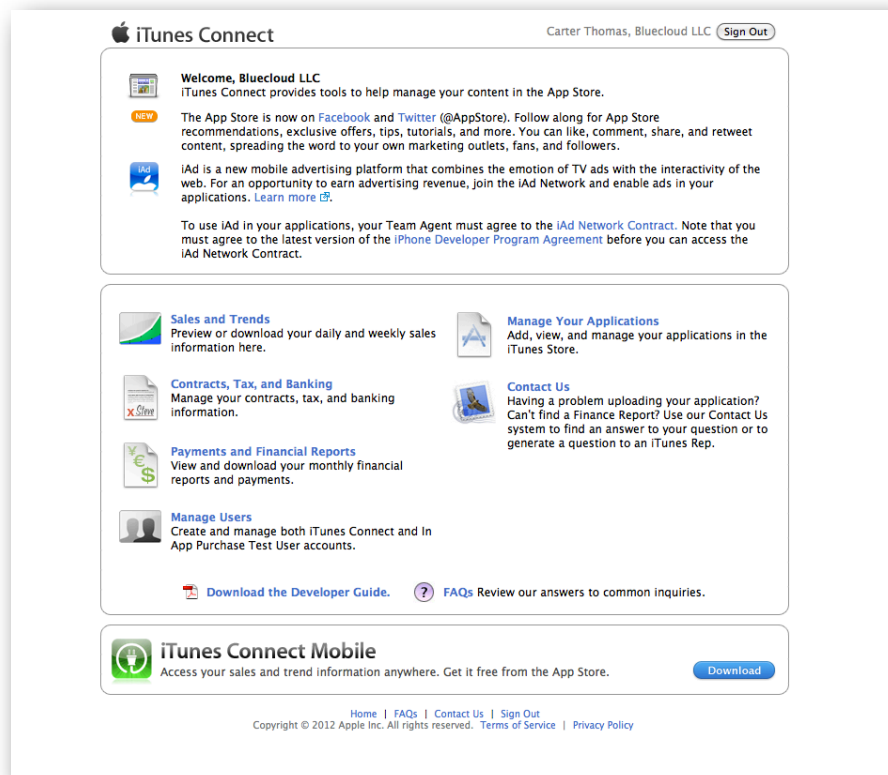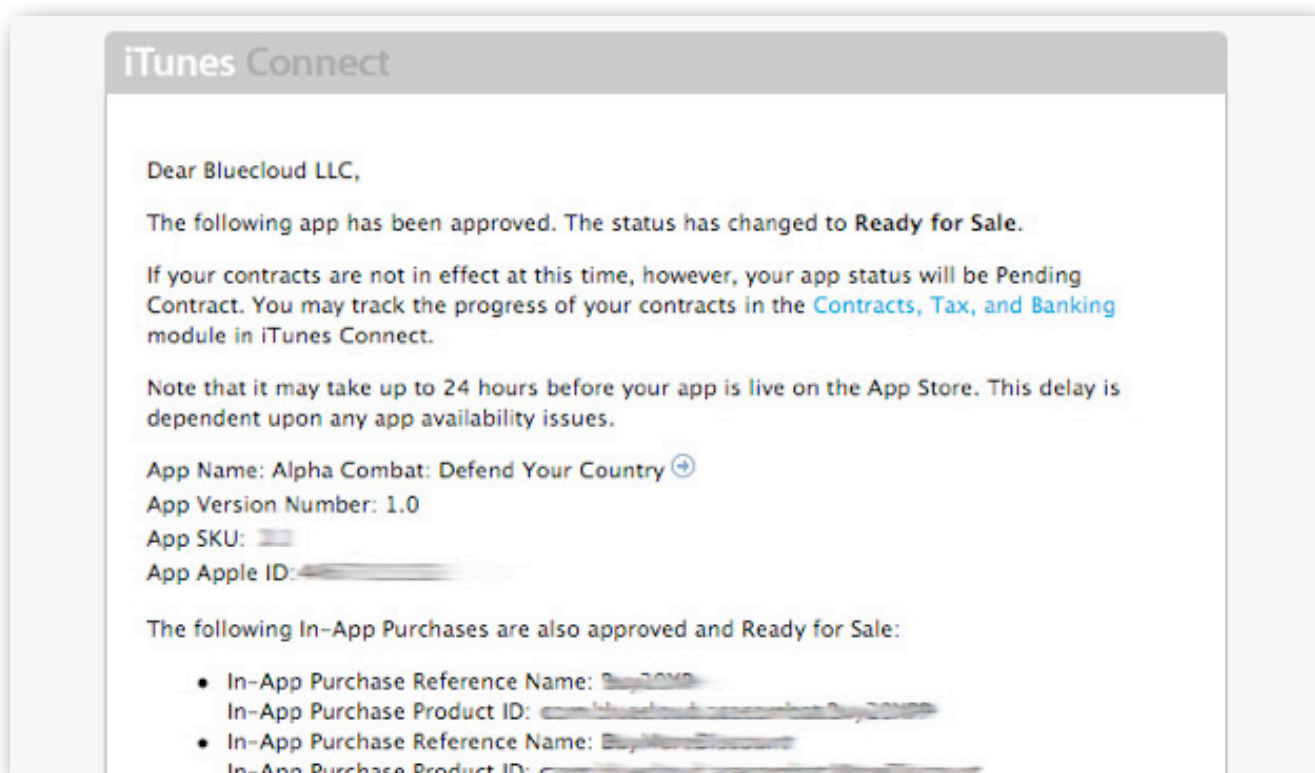
For sheer information load, I'm going to skip the whole Provisioning Profile fiasco, which is something your developer should take care of any way. In short, provisions profiles are Apple's permission slips for developers - it allows your phone to be a test device and then gives the go ahead to be able to distribute your apps. It's a HUGE headache and something you should leave to the pros.

Here is a basic breakdown of the publishing process:

1. Development finishes up and the developers have an code project that is built and ready for upload.
2. You (or developers) go into the iTunes Connect portal and set up a new app - name, version, language, in-app purchase IDs, GameCenter, etc
3. Once you set up the app, you will fill out the information such as the large icon (512px by 512px), categories, copyrights, meta description of the app, keywords, and screen shots.
4. Then you'll have your developers log into the iTunes Connect, prepare the app for upload, set the distribution certificates (these are provisioning items, don't worry about it), and load the archived build in Xcode up to iTunes Connect.

Then you wait about a week and you will get email notifications about when it goes in for Review, Processing for the App Store, and Ready for Sale. Let's talk briefly about the parts that YOU will probably care about using one of my other apps as an example.

iTunes Connect

Dear Bluecloud LLC,

The following app has been approved. The status has changed to **Ready for Sale.**

If your contracts are not in effect at this time, however, your app status will be Pending Contract. You may track the progress of your contracts in the Contracts, Tax, and Banking module in iTunes Connect.

Note that it may take up to 24 hours before your app is live on the App Store. This delay is dependent upon any app availability issues.

App Name: Alpha Combat: Defend Your Country ⊕
App Version Number: 1.0
App SKU:
App Apple ID:

The following In-App Purchases are also approved and Ready for Sale:

- In-App Purchase Reference Name:
  In-App Purchase Product ID:
- In-App Purchase Reference Name:
  In-App Purchase Product ID:

INSIDE iTUNES CONNECT

Name - pretty straightforward

SKU - kind of arbitrary, haven't found anyone who's given me a great explanation

Bundle ID - what the developers use to "sign" the game and in-app purchases

Apple ID - how Apple identifies your app

Type - iOS, Mac, Safari app

Default Language - English or whatever

You can see some parts on the right there - rights and pricing just allows you to change the price and the in-app purchases are more of a development deal. I wouldn't worry about most of that, except the pricing, so we'll skip it.

Version - 1.0 is usually the first release. If you do an update, you'll change that to 1.1 or something.

Primary Category - Where you want your search to come up (games will allow for 2 subcategories, as seen above)

Secondary Category - Listed in a secondary category, but usually does not rank

Copyright information - Add whatever you'd like

Rating - When you upload the game, you need to select different criteria about violence, nudity, etc to determine the audience that can download it.

Localizations - Also known as "translations" - this is when you have your app in different languages

Metadata - The information you see in the app store

Keywords - you are allowed 100 bytes (characters) of keywords here. Choose wisely.

Screenshots - now these all need to be retina resolution (960px x 640px). Your designer should be able to supply these.

Game Center - Lists and leaderboards and achievements if you have any

Now you've got your app in the store and you're ready to start getting SERIOUS downloads!

Right?

iPhone App Marketing

"Don't fight a battle if you don't gain anything by winning" - Erwin Rommell

If you made it this far, I want to give you a congratulations. That's really awesome! Having your own app in the store is a big accomplishment. I remember when I released Alpha Combat I was about to book my tickets to Aruba and pour myself a margarita as the money poured in.

But it didn't.

Well - not really. Definitely not how I had hoped, but there was some life. I had very high expectations and refused to settle for less. Thus began the great iPhone marketing journey.

What is iPhone App Marketing?

I actually get this question a lot. Is it exposure? PR? Downloads? Reviews? People come to me with an app that's cost them $30,000 and it's getting 25 downloads a day, making them close to nothing. Marketing is supposed to get your product in front of more people, right?

Well - yes and no. iPhone app marketing is a bit different than internet marketing or television marketing because it hinges on two pieces almost equally:

1. EXPOSURE
2. CONVERSION

Most marketing on the web is done via exposure - SEO, PPC, advertising, blogging, social media - all geared to get more eyeballs on a website. THEN people start talking about conversion.

The problem with the Apple platform is that you have very little idea what sort of eyeballs you're getting, but you do know how many downloads you're getting. **Thus, conversions are the only real way to measure impact.** Yes, there are third party software packages to get around this, but I'm not talking about all the bells and whistles. You need to get downloads....and those come from getting in front of more people but also convincing the **tens of thousands of people who see your app that it's worth downloading.**

So - I'll talk about what I did so that you can see it as a working model, then I'll add a few other options if you want to get creative. I'll bucket this as well as I can and put a relative importance weight against each.

This is by no means a strategy or a silver bullet to success. This should be treated as a tool set which you can select from. Remember - the best marketing is built on passion!

Design - HIGH IMPACT

The most fundamental of all marketing is going to be the design of the app. **Icon and screenshots make a huge difference in conversion.** Think about it - every single app has the same basic opportunity from the very start and will get about the same number of eyeballs as the apps released on the same day in that category.

The difference is when people see an icon they like with a name that's catchy and maybe a lot of positive reviews (we'll get to this later). They open the app and see some great screen shots and a description. **THIS IS YOUR FIRST IMPRESSION!** Please make sure your design is done well before you start getting into the external stuff.

BLUECLOUD TIP: You don't need an amazing app to make it look amazing. Hiring a professional designer to do some really nice high-resolution screen shots can make any app look awesome and exciting. You can illustrate someone using your app on a photoshopped iPhone screen or whatever - just make it look great.

Reviews - MEDIUM IMPACT

Part of marketing is to create the world you want people to see, to build a self-fulfilling prophecy. This can be done through getting reviews and making sure you continue to get reviews. I did this for Alpha Combat - I had about 60 people provide reviews over the first 45 days in order to make it look really popular. Do I have proof? Not necessarily, but I think everyone would agree that having reviews there makes an app look more legit.

You can get these by checking out a site like www.fiverr.com or going on elance.com and putting up a bid for someone to review your site. Be careful - do it too quickly or with poorly written reviews and it can actually hurt you, even if they're positive.

External Web Assets (Website) - MEDIUM/LOW IMPACT

I built www.alphacombat.com to support my app and it really doesn't get much traffic - maybe 5 or 10 hits a day when the app is free. The important thing is not to think about websites or web properties as inbound traffic funnels, rather **think about them as depth to your "brand."** When you start going to review sites or blogs, you'll be

amazed at how much more credible you seem if you can send them to a legit website instead of your Apple page. It just adds value to your sales proposition, which goes a long way.

YouTube/Twitter/Facebook - Medium IMPACT

I have a [YouTube video for Alpha Combat](#) and a [Facebook Page for Alpha Combat](#) to make my brand look more legit - but it definitely does not drive a lot of downloads. All it does it make people like my app more, which can lead to more purchasing, better likelihood of being featured on a review site, and people telling their friends. I don't have Twitter because I don't have the time to update it, but I integrated Twitter into another one of my apps ([@UBartending](#)) and that can be a fun way to engage your fans.

Both YouTube and FB are free and I would definitely recommend doing them. If you have an app that needs even the slightest of explanations, it's a good idea to have a YouTube video that walks through someone using the app.

Review Sites - Paid and Free (MEDIUM and LOW IMPACT)

I spent about $300 on paid reviews which got me a lot of exposure and press out of the gates. Interestingly, though, they did very little for my download numbers. The day on which my app was featured I saw no increase in downloads, nor did I see one the following day.

What I did gain, however, was some great people to quote, enhancing my credibility. You'll notice that in my meta description I quote one of the top paid review sites which is awesome for looking legit. I'd recommend doing at least one paid review and getting the full package (Video review, in-depth review, twitter blast, etc) and milk it for everything. Just don't expect a lot of downloads from it.

Press Releases - MEDIUM/LOW IMPACT

I say Medium/Low for the 90% of apps out there that really don't do anything special. I didn't do a press release because Alpha Combat itself doesn't have a killer story. What I COULD do a press release about, though, is that I'm an internet marketer who licensed

the source code in order to start putting out information on the web that no one else does. That would be a great PR move - telling people about how Alpha Combat has 10 cool levels and is integrated with Openfeint is just lame.

If you're going to do a Press Release, sell a story, not functionality.

Promotions - HIGH IMPACT
Some promotion networks suck, but some are great. I was lucky enough to get featured as an Openfeint Free Game of the Day and it was awesome for my app. [Read the full post here](). I'm doing other promotions and you definitely see a spike when you get released on these networks. Your best bet is to Google "free app promotions" and contact the teams to get pricing. You can expect to pay anywhere from $300 - $25,000 to be featured for a day. This will get you on their website, app, and social networks. It can be enormous for growing your user base.

If you're looking to spend some money and get your downloads juiced, this is a good option.

Advertising Networks - MEDIUM/HIGH IMPACT - DIFFICULT ROI
I really can't recommend this to anyone unless you've got a serious monetization strategy because the numbers are just so hard to justify. You'll get lots of downloads, but it's going to cost you an arm and a leg. You'll have two options:

1. CPM models - you pay per thousand eyeballs on your banner (iAds, AdMob)
2. CPI models - you pay per install that users in your network perform

The CPI model can range from $.50-$3.00 per install. Think about that - you're paying up to $3 for one download! You better have some serious in-app purchasing to make those numbers work.

I leave the advertising for the big guns who've got a few hundred thousand to spend. Also, it costs $100K just to get in the door and advertise on iAds, FYI.

Keywords and Meta Description - HIGH IMPACT
What you write in the description area can make a big difference. This is what's going to convince people to download your app, along with being picked up on any search terms coming through the iTunes network. I have seen people double their downloads once they updated their meta descriptions - I did my own experiment where I changed the keywords in the app and tripled my download numbers.

BLUECLOUD TIP - Keyword research for the app market is NOT like a website. Don't use keyword optimizer. Instead, find as many apps that are just like yours that have huge download numbers, and use the names of those apps as your keywords. You should be able to get about 8 app names in your keyword area and it will skyrocket your downloads.

Pricing and Price Cycling - MEDIUM/LOW IMPACT
This refers to changing the price, specifically changing the price from paid to free. Let me preface this by saying two things:

1. If you don't have a few hundred downloads a day when the app is free, this probably isn't a great option for you.
2. This is NOT good for making your brand look really legit - each time you shift this pricing, you get a spike but it gets notably smaller.

That being said, switching from Paid to Free can be a powerful tool if you're looking to get a lot of downloads quickly. [Here's a blog post I wrote](#) on what I did to cycle the pricing on Alpha Combat that worked pretty well. You'll see the marketing model and why this happens with all the site aggregators and the way the Apple store currently works.

A good analogy is to imagine that when you're paid, you're filling up with water. When you go free, you're releasing that water. Water means downloads. So - if you try to do this too often, you're only going to have a small impact vs. doing it sparingly which will give you a larger impact.

All Systems Go - The Ultimate iPhone App Marketing Power Move

I've talked to a lot of developers and marketers and pretty much everyone agrees with me on this - the absolute best thing you can do in this market is to put all your eggs in one basket and go for gold on one day with the hope of climbing the charts as high as you can across the world.

When I did the Openfeint promotion, I went to the top 10 in Italy for about 4 days and had HUGE download numbers. I talked to a few guys who organized four or five different promotions to all set off on the same day and had enormous success.

Here's what you should do:

1. Make your app paid for at least 60 days.
2. Work on an update and have it set to Manually release. Make this update as awesome as you can (new levels, new graphics, whatever)
3. Find at least 3 "free app for a day" promotion networks and sign up all of them for the same day (***note - you may have to update your meta description as part of the promotion terms, so make sure you're clear that only one of these requires you to do that).
4. On that day of the promotions, switch your app to free and release your update.

This is without a doubt the #1 way to get a massive spike in downloads. This will begin to put you on a different level because of the word of mouth effect, along with giving you a much more potent update strategy.

In closing, the best tips I can give are:

1. Tell a story and sell that story, not an app.
2. Make the design awesome
3. Create a really great brand out of your app

The rest is easy :-)

Yo, Carter - What About Monetization?

Honestly, making money on apps is going to require an entirely separate ebook (which I am working on), but let me save you all a lot of time and energy by cutting through a year of testing and furiously working with every method imaginable.

Here's the list of potential monetization paths:

- Paid vs Free
- In-app purchasing
- [Advertsing](#)
- Direct Deals - Cross Promotions
- Email Capture and Affiliate Marketing
- Lite Versions
- Subscription Models
- Analytics
- Sponsorships

If you're interested in this stuff, stay tuned for my next ebook release. In the meantime, you need to check out [my blog post about this stuff](#). Here's why:

[RevMob](#) created an ad network that sells downloads ONLY to high quality and high paying advertisers. This means that instead of having advertisers who can only pay YOU $0.10 per install, RevMob can pay out $1-3.50 per install. Think about that for a second.

I installed [RevMob](#) (one line of code, by the way) into all my apps and increased my income from $8 day to $175/day. That's game changing....and happens to pretty much everyone who uses this service.

The best part? It's designed for YOUR free apps and sells OTHER free apps. Everything is free to the consumer. You can have apps in any category, not just games.

Free to sign up, also. If you have an app or are developing an app, make sure you read this blog post before you decide on a monetization strategy.

[Here's a full post about my experience and why I use this service](#)

Conclusion

I know this is an enormous amount of information and should probably be a 200 page ebook, but I really wanted to get at least something out there that can answer questions and help steer people into the right direction.

I didn't talk about Android or other platforms because I don't really deal with them and they are built on entirely different SDKs (software packages). The biggest question I get is how much it costs to build for both iPhone and Android at the same time - which I usually ballpark at about a 30% increase in price.

Personally, I think apps are an amazing engagement model and are going to slowly become a powerful tool for brands and consumers. As a marketer, I see the greatest opportunity for apps in it's ability to promote all other areas of a brand's strategy into an awesome user experience. **The future of apps will be in the ability to blend creativity with traditional business.**

I urge you to stay in touch and comment on my blog. I really appreciate your time. If you have any questions, the best thing you can do is comment on the blog - I like to respond openly to questions so that everyone can see the answer and learn from it. You'll get a lot more out of me if I know it's going to help everyone and not just you.

In the meantime, keep dreaming and play to win!