

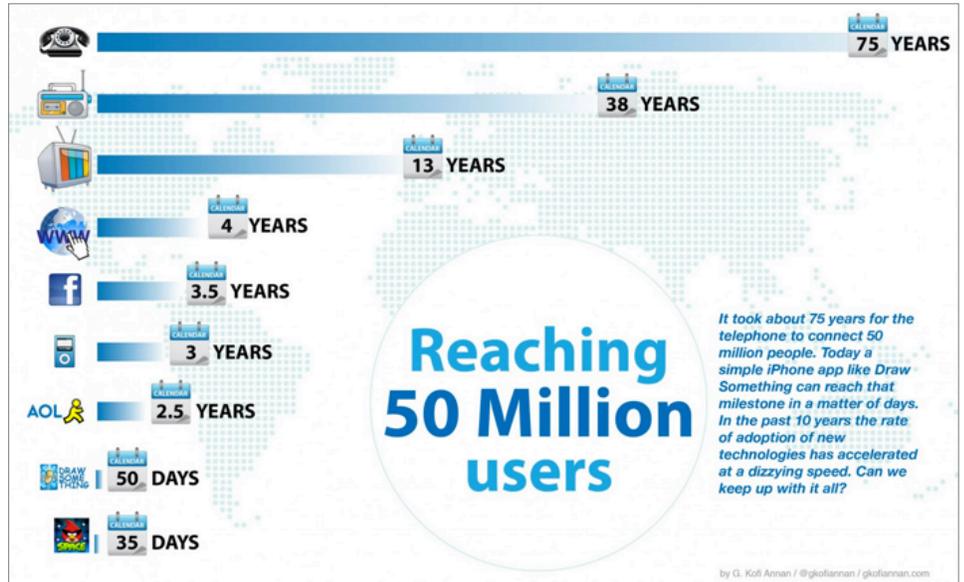
# APP DYNAMICS

An AppDynamics Business White Paper

## Going live with a mobile app

According to a report from Kleiner Perkins, mobile applications now account for 15% of all Internet traffic, with 1.5 billion users worldwide. Whether you are a brick and mortar retailer or an ecommerce web site the need to go mobile is greater than ever. With mobile market share growing year over year, mobile applications are the new web site. If you don't have one available you are missing a major market. Mobile adoption has accelerated like never before.

---



<http://visual.ly/reaching-50-million-users>

As of May 2013 91% of American adults own a cell phone and 56% of American adults own a smart phone according to a recent survey by Pew Internet.

So you want to launch a mobile application and skyrocket to your first 50 million users. There are many concerns when going live with a modern mobile application. From project planning, to development, to quality assurance and managing a successful launch. In this white paper we'll talk about some important considerations for planning, building and launching a mobile application in order to ensure a good experience for your end users and a good outcome for your business.

## Phase 1: Planning a mobile application

Planning a major mobile application is the one of the most difficult projects to execute. What platforms do you build for and what devices should you support? Do you build your own backend or use a platform as a service? Do you build a native mobile application or a web based HTML5 application? All of these questions and more will come up and you need to understand the market to make decisions you won't soon regret.

### **Design consideration: Native or HTML5?**

The promise of HTML5 is to write once, run everywhere. HTML5 is appealing because it requires less development effort and will work on all platforms (in a browser). This promise comes at the expense of the features and user experience, however. While build HTML5 mobile applications are a good start, they will never have the fluid user experience of native applications. The best application experience comes from a native integration.

HTML5 does have some strategic benefits:

- Low cost to deploy cross platform
- Immediate updates and complete control of distribution
- Larger talent pool to hire (HTML vs iOS vs Android)
- Device compatibility

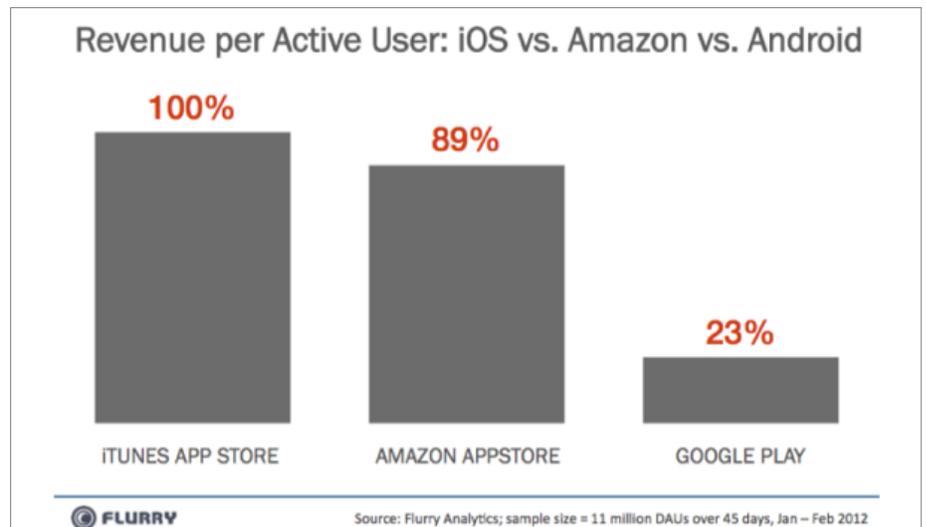
Native applications have some benefits as well:

- Richer user experience
- Better performance
- Native integrations that enable better monetization

If you already have an HTML5 application but would like to switch to a native integration, there are several intermediary solutions available to take your HTML5 app and convert to a native application available for iOS and Android. Appcelerator Titanium, PhoneGap, and Sencha Touch all claim to provide a native experience without having to write a native application. In my experience, however, all of these solutions fall short in delivering a truly compelling mobile experience. Several high-profile companies have regretted pursuing an HTML5 mobile strategy, most famously Mark Zuckerberg, CEO of Facebook has publicly admitted, “On iOS and Android, you can do so much better by doing native work...” and that focusing on HTML5 was a mistake in the long run. If your only metric is time to market these solutions provide a good result, but if you care about the user experience then native is the only mobile strategy.

**Design consideration: Apple App Store or Google Play?**

The Apple App Store and Google Play app ecosystems have distinct communities with the Apple having stricter regulations and a manual review process for releases compared to the much more agile and developer friendly Android community. Google has a top price of \$200 compared to the high end of Apple of \$999 with much more variance in pricing than the Google Play market.



<http://blog.flurry.com/bid/83604/For-Generating-App-Revenue-Amazon-Shows-Google-How-to-Play>

In 2013 if you want the most downloads you build for Android first, but if you want the most revenue you build iOS first. The reality is you are building a business with your mobile application, and the money exists mostly in the iOS App Store community. Even though the Android market is larger, Apple apps make more money. Apple owns only 18 percent of the app universe, but it makes almost 500 percent more than Google, pulling in a \$5.1 million in revenue daily. Meanwhile, Google owns 75 percent of all app downloads, but it only takes in \$1.1 million per day.

The biggest difference between the app store markets is the regulations enforced on app developers. Apple, while the biggest earner, has the strictest functionality requirements and restrictions and quality standards that ensure it's the highest quality market, but also the most censored. While the Google Play Store is more developer-friendly with a straightforward app submission and distribution process with better tooling. Both markets have more than 100 million devices and over a billion app downloads.

Due to the constraints and expectations of the venture capital markets that so many app developer are constricted by it doesn't make sense to choose Android as a launch platform. The expectation for traction and revenue means Apple iOS is the better business bet when you have limited time and mobile resources. It's the fastest platform to prove an app concept and the viability of a business. The harsh reality of the mobile app means that for every top-earning app there are 10,000 failures.

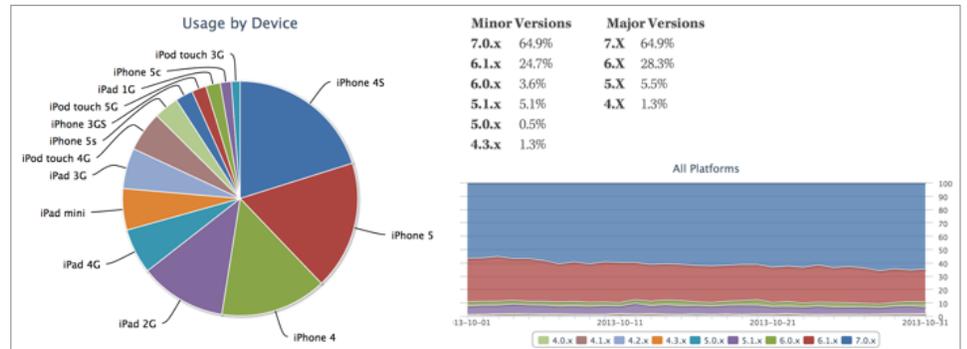
A look at the current state of the mobile device market share:

<b>Operating System</b>	<b>2Q13 Unit Shipments</b>	<b>2Q13 Market Share</b>	<b>2Q12 Unit Shipments</b>	<b>2Q12 Market Share</b>	<b>Year-over-Year Change</b>
Android	187.4	79.3%	108	69.1%	73.5%
iOS	31.2	13.2%	26	16.6%	20.0%
Windows Phone	8.7	3.7%	4.9	3.1%	77.6%
BlackBerry OS	6.8	2.9%	7.7	4.9%	-11.7%
Linux	1.8	0.8%	2.8	1.8%	-35.7%
Symbian	0.5	0.2%	6.5	4.2%	-92.3%
Others	N/A	0.0%	0.3	0.2%	-100.0%
<b>Total</b>	<b>236.4</b>	<b>100.0%</b>	<b>156.2</b>	<b>100.0%</b>	<b>51.3%</b>

Source: IDC Worldwide Mobile Phone Tracker, August 7, 2013

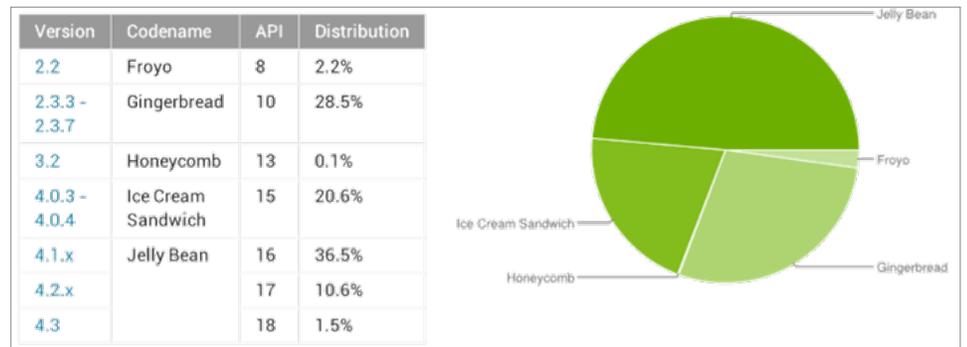
<http://www.idc.com/getdoc.jsp?containerId=prUS24108913>

Once you decide which platform to focus your mobile engineering resources on you need to figure out what devices and versions you are going to support. As of October 2013, iOS 7.x accounts for 64.9% of all iOS devices:



<http://david-smith.org/iosversionstats>

As of October 2013, Android 4.x market share accounts for 48.6% of all Android devices:



<http://developer.android.com/about/dashboards/index.html>

Based on market share it makes business sense to offer compatibility with 95% of users by supporting Android 2.3.3+ and iOS 5.1+ devices. While you can adjust to fit your engineering resources this gives the best coverage for developer investment while still supporting the latest APIs.

**Design consideration: Tablet or phone?**

As PC shipments declines to 8.4%, tablets are expected to grow 53.4% according to an October 2013 Gartner Forecast. There is a clear shift from traditional desktops to tablet and smart phones. One of the design considerations you must factor in is the variety of form factors in mobile devices. Crafting an application experience that adapts to both landscape and horizontal modes that works across high definition and standard screens from 3 to 11 inches is an impressive feat. Especially when having to balance between high-speed low-latency Wi-Fi connections versus low-speed high-latency 2g/3g/4g connections.

Device Type	2012	2013	2014
PC (Desk-Based and Notebook)	341,273	303,100	281,568
Ultramobile	9,787	18,598	39,896
Tablet	120,203	184,431	263,229
Mobile Phone	1,746,177	1,810,304	1,905,030
<b>Total</b>	<b>2,217,440</b>	<b>2,316,433</b>	<b>2,489,723</b>

Source: Gartner (October 2013)

<http://www.gartner.com/newsroom/id/2610015>

The harsh reality is that mobile applications have a fickle audience that is heavily reliant on curated app stores and reviews. Reviews can make or break a mobile application as can being featured in an app store.

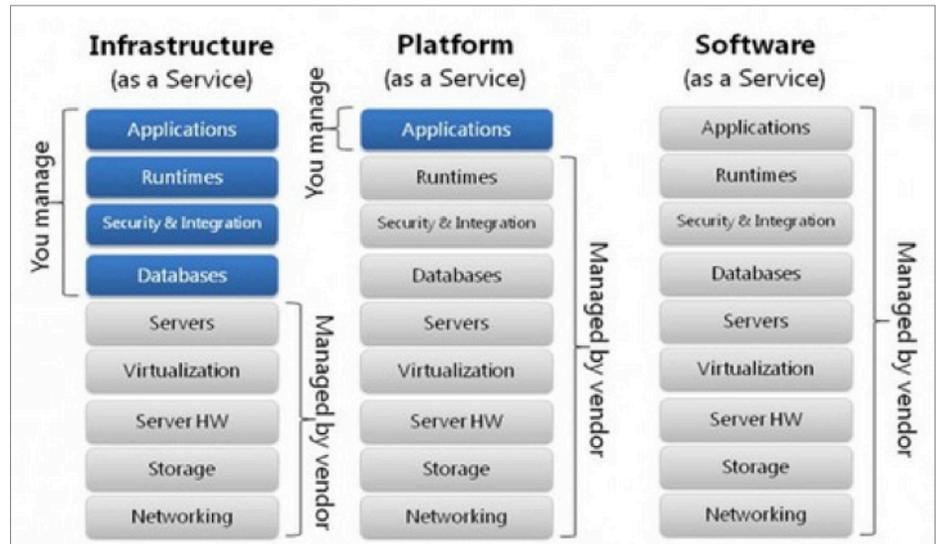
This is why being proactive with quality assurance and production monitoring has a significant impact on the success of an application. Not only must the application work as designed but also the experience must be refined. My recommendation is to ignore the tablet form factor until you have a polished phone experience, but recognize it is a growing market.

## Phase 2: Developing a mobile application

With mature open source frameworks for both iOS and Android it is easy to have a solid foundation by standing on the shoulders of giants. Github is your friend when it comes to discovering the open source communities that power so many iOS and Android applications. While iOS is developed privately there are many libraries available like AFNetworking, RestKit, SSToolkit, and QIF. Android is a completely open source operating system with many libraries available, like GreenORM.

### Development consideration: PaaS vs IaaS vs SaaS

When undertaking a new mobile application you must decide to integrate your existing backend or build a new one. In the case of building a new backend for your mobile application you must consider whether to use an existing mobile cloud like Parse, Stackmob, and Kinvey. These mobile cloud platforms allow you to develop new applications quicker with APIs for writing custom code, storing user and app data, managing push notifications, and instrumenting with analytics. Mobile cloud platforms as a service are the fastest way to get to market if you don't have existing server side APIs. More often than not you will need to integrate your existing application data to your mobile experience. When you choose infrastructure as a service, or platform as a service, you are really deciding which problem set do you want to focus on. Either you can focus 100% on your application or have more flexibility (and ongoing maintenance) by using a lower level provider that provide infrastructure as a service.



[http://www.appdynamics.com/blog/wp-content/uploads/2013/11/iaas\\_vs\\_paas.png](http://www.appdynamics.com/blog/wp-content/uploads/2013/11/iaas_vs_paas.png)

The benefits of the mobile cloud are bountiful from managing application data and user state to social integrations and push notifications. There are a large variety of providers to choose from like Parse, Stackmob, Kinvey, Heroku, and DotCloud.

**Development consideration: Degrade gracefully for a smooth online and offline experience**

When designing a modern mobile application you must plan for a variety of conditions for high speed, low latency Wi-Fi connections to low speed, high latency 2g connections, and then being offline all together. Adapting to the sheer variety in devices, form factors, CPU speeds, and memory variability is a daunting feat. It is critical to be able to support online interactions and gracefully fail back to offline while maintaining state and a consistent user experience. The best strategy is to build for failure conditions with retries, long timeouts, and extensive caching.

**Phase 3: Launching a mobile application**

Mobile app audiences are a notoriously fickle bunch and a poor first impression often results in a very harsh app store review that will negatively impact your app’s growth and adoption. When an app store rating can make or break your business you have to be diligent in making sure every user has a stellar experience. The best way to do this is thoroughly test your mobile experience and load test your backend can ensure you can handle traffic.

The key to a successful launch is great planning and testing. Launching mobile applications are significantly more difficult than the common web application. Not only is the audience more fickle, but also you have to adhere to third-party processes and procedures while testing is more cumbersome do the sheer number of devices/form factors.

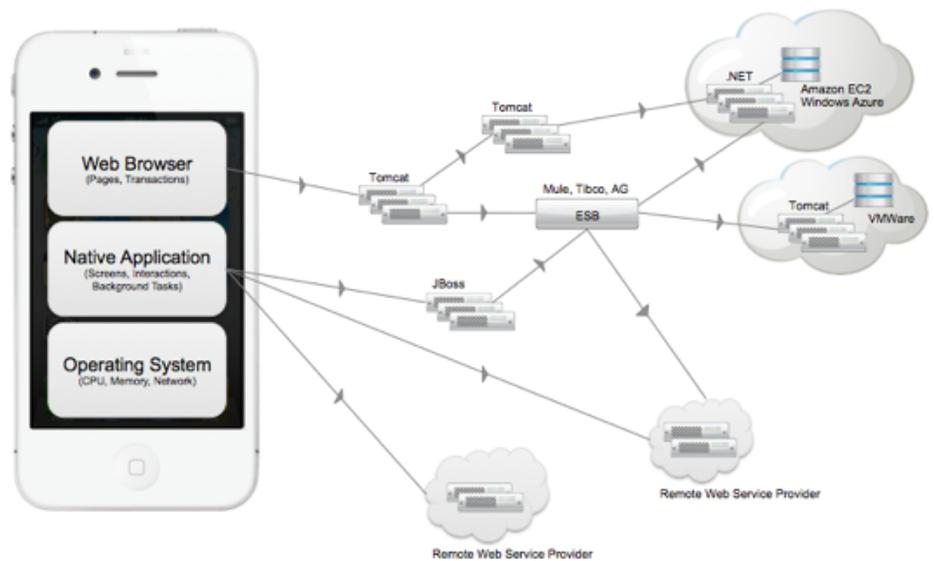
Thorough quality assurance, crash and error reporting, load testing, and proactive production monitoring are essential to launching a successful mobile application.

**Launch consideration: Testing native applications across mobile devices**

Testing mobile applications is notoriously difficult due to the vast number of devices. There are a few services that make this easier for engineers. I have seen a few strategies for testing mobile devices – usually you go to Amazon and buy the top 10 devices for Android and iOS and manually test your application across every device. Mobile device labs of this sort are quite expensive to setup and maintain and often require some level of automation. Alternatives to setting up your own mobile lab is to use a mobile app testing platform like Apphance and TheBetaFamily. They offer an easy way to test your native application across many different devices and audiences with ease.

**Launch consideration: Capacity planning and load testing**

Capacity planning is key to the successful launch of any web application. If you want to understand what can go wrong look no further than the failure to launch of healthcare.gov. Understanding your limits and potential capacity is a requirement for planning how much traffic you can handle. Making an educated guess about potential growth and you can come up with a plan for how many concurrent users you might need to support (and then double it).



Once you understand your maximum concurrent users you can test your backend infrastructure to be sure your mobile experience doesn't suffer. There are quite a few tools available to help you load test and evaluate the scalability of your backend platform. Apica, Soasta, and Blazemeter offer services that allow you to simulate your mobile application being used at high levels of concurrency.

**Launch consideration: Beta testing**

Beta testing is the last quality assurance step before you can make your app generally available. Testflight, HockeyApp, and Ubertesters allows you to distribute your application for testing to a select group of users. When it comes to beta testing the more users you can convince to give feedback and the larger distribution of devices the better. These beta testing and distribution tools enable you to easily gather feedback early on about what isn't working in your application and save you from the embarrassment of negative app store reviews due to obvious problems. This is an essential step to a successful launch – the more beta testers you can find, the better.

**Launch consideration: Hard launch or Soft launch?**

Once you have beta tested and decided you have a great application that is battle tested for production you need to decide how to launch. The real question is hard launch or soft launch. The traditional hard launch is straight forward your app is approved in the app store and you go live. There are a few different strategies for soft launches of major applications. The most common is to soft launch outside of your primary market. If you are planning to release in the USA you can easily pick another region with similar characteristics like Canada, Australia, or the United Kingdom. The benefit of soft launching in a secondary market means you can validate assumptions and the last step in beta testing your audience.

If you launch are concerned about the overwhelming demand you can use the common approach of inviting and throttling creation of users.

Soft launching can validate product/market fit, app experience and usability, and app/game/social mechanics. The result is your first experience with you key demographic will be based on data you learned from your sample audience. The end result will be much more polished and informed.

**Launch consideration: Understanding the App store submission process**

The application submission process varies greatly depending on the app store. My simple recommendation is to not forget to invest time and patience to market your app correctly and allocate enough time for approval and release. This is where you get to sell your application with a marketing description, search keywords, and screenshots of your app in action. You can specify pricing and what regions/markets you want your app to be available in.

With Apple it is customary to wait up to two weeks for Apple to review your application and approve it for production. Apple routinely rejects applications for being low quality, using unsupported apis, and for not following design guidelines. Google on the other hand offers a streamlined release process that takes less than one hour, but doesn't offer the first line of protection that Apple offers by not allowing apps with obvious flaws. When you have to wait two weeks to release an update you need to make sure there are no critical bugs in the first place. While Apple offers emergency updates it is frowned upon and said to be unforgiving after more than one or two.

## Phase 4: Monitoring your mobile application when it goes live

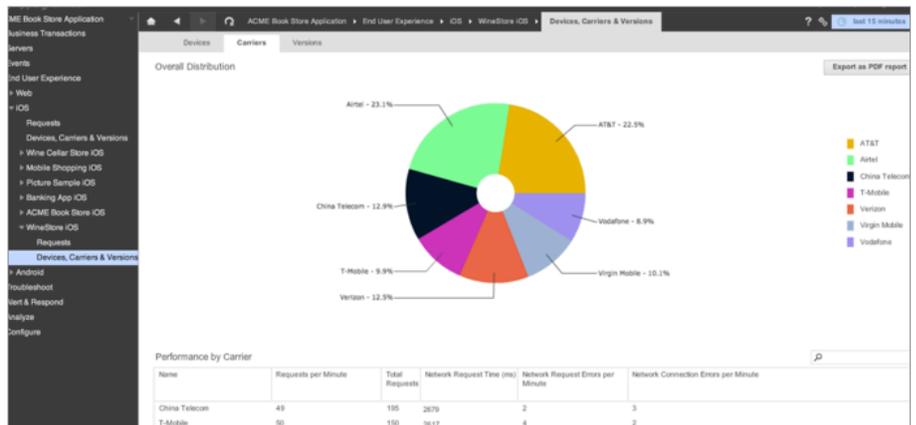
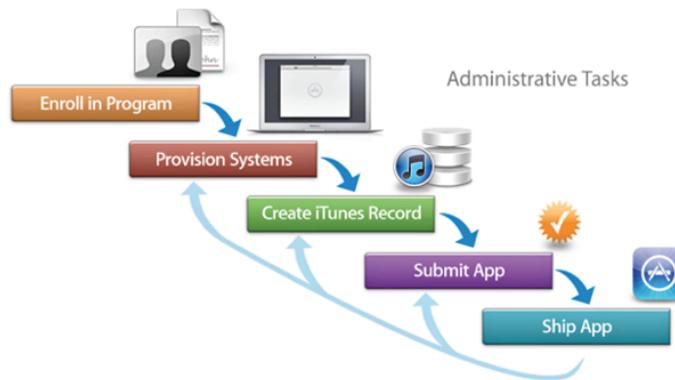
### Production consideration: Crash and Error Reporting

Crash and error reporting is a requirement not only for the development of your application, but also for testing and production. There are quite a few crash-reporting tools available including AppDynamics, Crashlytics, Crittercism, BugSense, HockeyApp, and TestFlight. All of these tools have are capable of reporting fatal errors in your application to help developers track down the root cause of bugs. The problem with crash and error reporting is that it only tracks issues after they have affected users. Both the Apple App Store and the Google Play Store provide basic crash reporting metrics. Even though this is convenient for developers, it is no replacement for good testing.

### Production consideration: Analytics & Instrumentation

Smart executives are data driven and mobile applications can be a plethora of business intelligence. When it comes to instrumentation, the earlier and more metrics you have, the better. Analytics and instrumentation are crucial for making informed and smart decisions about your business.

Who is your audience? What platforms and devices do they use? What user flows are the most common? Who do users abandon? Where are my users?

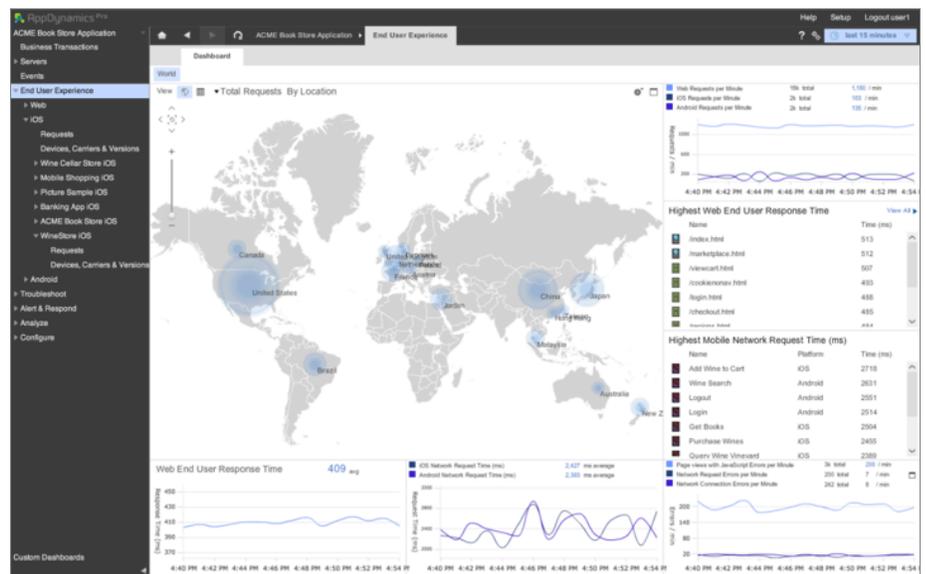


Tracking important demographics of your audience like operating systems, devices, carriers, application versions, and geography of users. These metrics allow you to target your limited resources to where they are needed most. There are quite a few analytics platforms built for mobile including Google Analytics, Flurry, FlightPath, MixPanel, KissMetrics, Localytics, and Kontagent.

All of these tools will give you better insights into your audience and enable you to make smarter decisions. There are important metrics to track like total number of installations, average session lifetime, engagement, and geography of your users. Once you have basic user demographics you can use MixPanel or KissMetrics to track user activity with custom events. The more instrumentation you add to your application the more metrics and customer intelligence you will have to work with

**Production consideration: Application Performance Monitoring**

Application performance monitoring (APM) tools enable you to discover the performance of both your mobile and server-side applications in production. APM allows you to understand your application topology, third party dependencies, and the performance of your production application on both the client-side and the server-side. Modern application performance management solutions like AppDynamics track crashes and errors, the performance of the mobile application and correlates performance to your backend platform, and provide rich user demographics, and metrics on the end user experience. With modern business reporting tools you can evaluate the health of your application and be proactive when performance starts to deteriorate.

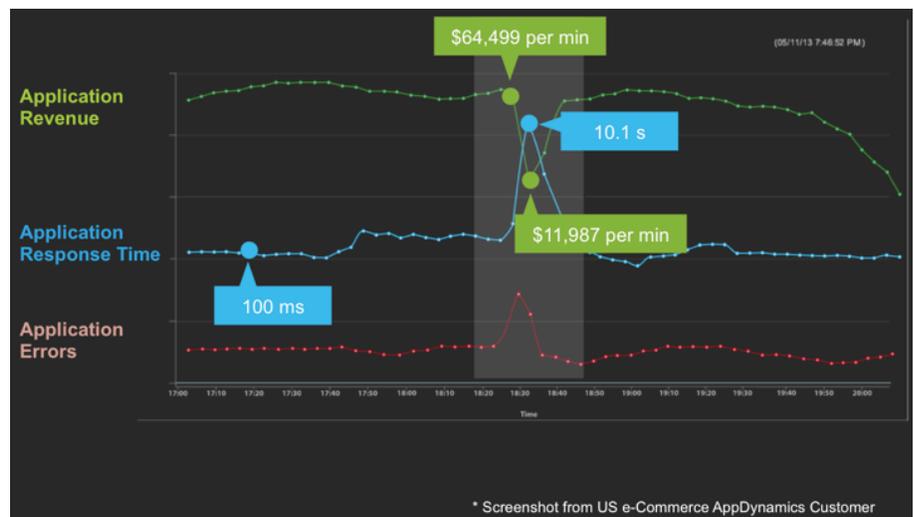


End User Monitoring allows you to understand the application experience from the eyes of your real users. There are quite a few solutions to monitoring the end user experience in the market place. AppDynamics, Crittercism, New Relic, and Compuware allow you to instrument your application and gain visibility into production performance problems.

## Phase 5: Optimize your business revenue

### Business consideration: Real-time business metrics

Once you have launched a successful mobile experience you need to understand how that experience effects your business. If you have a business critical application like the Apple Store Checkout application or the Fedex Package Management application your business is dependent on performance and use of your application. You can gain valuable insight into your business if you track and correlate the right metrics. For example, how does performance affect revenue or what is the average price for a checkout transaction. Understand your core business metrics and correlate them to your mobile experience for maximum business impact.



### Business consideration: Monetization

Your plan is to retire off this application so you need to have a monetization strategy. The most common ways to make money from applications are Pay to play (charge a fee for your application), freemium (offer a free and pro app), offer in-app purchases (like levels), and finally advertisements. All of these strategies require precision execution, but some strategies work better for specific types of apps. Experiment with multiple strategies and do what works best for you.

## Moving forward

It is no longer enough just to have a presence on the web. In fact more and more companies are going mobile first. The mobile landscape is constantly evolving and the mobile market is seeing continued growth year over year. Having a mobile presence isn't enough – you need a lot of planning in order to make a truly compelling mobile experience that's rewarding for your end users (and your business). For more information on mobile trends and monitoring your live mobile apps visit [www.appdynamics.com](http://www.appdynamics.com).

Read our blog post to learn more: [www.appdynamics.com/blog/category/mobile](http://www.appdynamics.com/blog/category/mobile)

## About AppDynamics

AppDynamics is the next-generation [application performance management](#) solution that simplifies the management of complex, business-critical apps. No one can stand slow applications—not IT Ops and Dev teams, not the CIO, and definitely not end users. With AppDynamics, no one has to tolerate slow performing apps ever again

Visit us at [www.appdynamics.com](http://www.appdynamics.com).

Try it FREE at  
[www.appdynamics.com](http://www.appdynamics.com)