iTexico | Smarter Software Development
Innovation, Near and Now.

8 Steps to Create and Execute Your Mobile Strategy

End to End Mobile Development with Multidisciplinary Software Teams

www.itexico.com

8 Steps to Create and Execute Your Mobile Strategy

End to End Mobile Development
with Multidisciplinary Teams

iTexico | Smarter Software Development
Innovation, Near and Now.

Index

Eight Critical Steps to Secure Your Smartphone Strategy

The evolution of mobile computing has been a bullet-train ride to a future that many of us could not have foreseen. Was it not only yesterday that we were still exchanging thought fragments via interoffice memorandum & flipping through catalogs or –perish the thought– going to a bricks -and- mortar location to purchase a product?

Our progression from PC to laptop to "dumbphone" to smartphone has been swift and paradigm - changing. Consumers who have first eschewed setting foot into a store, and more recently, aban - doned the home computer, are now expecting that their mobile, internet-connected device serve as a primary portal into a world of product acquisition - or offering value as an employee of an enterprise.

If this "harsh reality" feels like the sting of a Band-Aid being ripped off, then one could say that the typical reaction of marketers – to employ the easiest, cheapest and quickest fix – would in fact be like using Scotch Tape to close an incision.

Believe it or not, there is a difference between a slap-dash, triage-style mobile solution; and one that has been pondered thoughtfully & developed strategically. And your consumer? They certainly know the difference.
Without a mobile strategy, businesses have experienced catastrophic failures, including:

• Adoption problems caused by uneven and inconsistent user experience that differ from platform to platform

• Incongruous, use-inhibiting designs implemented by multiple studios or various departments working independently of each other

• Frustration caused by a lack of a common support and maintenance strategy to support the portfolio of applications

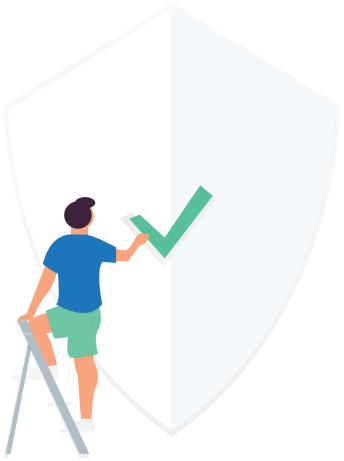
• Missed development deadlines or incomplete work caused by business leaders who underestimate the complexity of developing, deploying and maintaining the app

Fortunately, you don't have to "mobilize" your enterprise just for the sake of being on mobile. Our recipe for a mobile strategy consists of eight steps you can take right now to plan, prioritize and map
out a roadmap that will result in true, and lasting benefits.

Step 1 - Focus on value

Here is where you ask the key question about what you'll achieve from your new mobile strategy. Are you simply looking to add value to your user's journey? Do you want to reduce your operational costs or reduce your risks? Or is it as black and white as simply increasing sales through the added convenience of a mobile platform? Is it a combination of all of these factors? Once we list and rank these priorities, we have the needed starting point that will ensure your map is both purposeful and actionable.

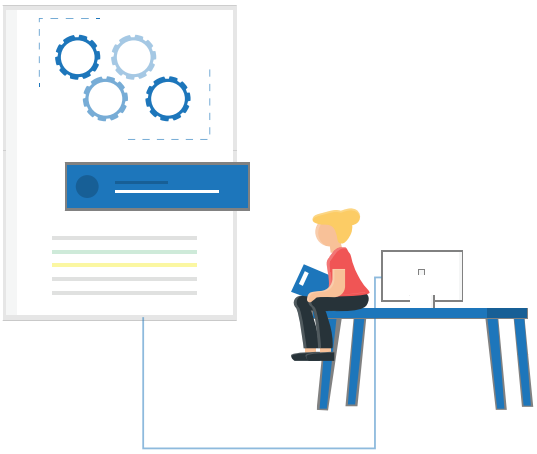
Step 2 - Develop use cases



Here is where we will more specifically pinpoint the key benefit to your mobile user. A great app is clearly defined by the way it addresses the consumer's most critical need. Companies that have clearly defined use cases for their apps include Starbucks (easing transactions), Zillow (marketing homes), Evernote (extension of a desktop platform), Facebook (content consumption) and Yammer (productivity). At this stage, you'll also want to define details like frequency of use, performance considerations and device resources that the app will utilize.

Step 3 - Target Distribution, Devices & User Personas

Who in your customer base should absolutely download the new app, the moment it's available? The use case we defined in Step 2 should be a big help in determining this. Once we define the personas – e.g. moms with kids, teenagers, or realtors – we can make some decisions about how they will interact with both app and device, and plan our development and



marketing strategies accordingly.

Step 4 - Develop your app

Now, we're ready to build – and to address the key technical decisions:
should it be a native app – downloadable from an application store? Should it be built to spec for a particular device?
Do we want it to be a scaled version of our website, using responsive design across platforms? Or do we want a hybrid?

Step 5 - Understand platform considerations

The Enterprise needs to make a host of platform choices to develop a support strategy for their portfolio of Applications. Examples of this include Test Automation to reduce the cost and complexity of testing across multiple devices and form factors, as a means of preventing bugs and improving time to market; crash detection to pinpoint the factors that trigger App crashes in the field; performance monitoring and testing to ensure app speed is as desired; analytics to measure user behavior including – if necessary – A/B testing and cohort analysis; integration of advertising and other monetization strategies; and infrastructure testing to ensure whether you can handle a doubling or tripling of traffic without a hiccup. This "support infrastructure" around App is a huge cost factor, and is also a large determinant of whether you can compete with the leaders in your category.

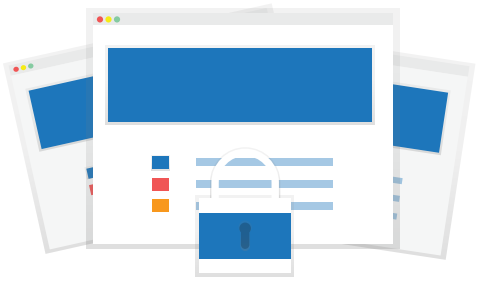
Step 6 - Assess Your In-House Development Skills & Budgets

Do you have the human and financial resources to support your mobile strategy? Budget considerations are certainly clear during the buildout phase, but have you allocated for the type of maintenance and testing delineated during Step 5? If you are supporting multiple languages, do you have the personnel to interact fluently with native speakers? Now is the time to determine whether you wish to hire to these capabilities

or contract with subject-matter experts.

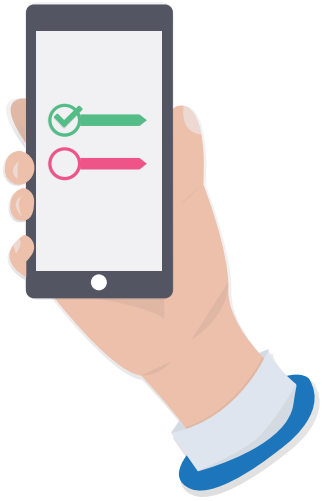
Step 7 - Ensure your consumer's security

Have you found – and plugged – all of the holes that can let nefarious people penetrate your armor? Often, hackers can even exploit a mobile weakness to bring down an organization's entire technological backbone.
Understanding – and planning – for the specific security needs of your company and your consumers is a key consideration.



Step 8 - Provide Maintenance & Support

Though we are constantly testing and analyzing the app, there will always be something for which we have not planned. Apple may make changes to its operating system or app store rules, and we may need to react quickly to avoid obsolescence. A subset of users may uncover a bug that we didn't know about. A solid plan for maintenance, support – and the next edition of your app – is absolutely critical to protect your investment.



Step 1
Focus on Value

Hopefully if you are reading this, you are convinced that allowing individual departments to hire their own vendors – and then hoping to reconcile the resulting mish-mash of platforms, user experiences and back-end services – will not be the best way for you to create a mobile strategy that will stand up to the vigors of today's competitive landscape.

The first step in formulating an Enterprise Mobile Strategy should be to focus on value. Indeed, the first raison d'etre of any strategic mobile effort should be to identify the key benefits the Enterprise can derive from its mobility efforts.

Make a wish list

The first step is to gather an inventory of all the existing and wish-list mobile applications across all departments. Closely examine this portfolio of apps and app requests. Be sure to add to this apps that your competitors have. Go down the list and decide what value each brings to the department or Enterprise.

Ideally, you can translate this value into a dollar amount for each application. Do you want to reduce your operational costs? Do you want to reduce certain compliance - related risks? Or is it as black and white as simply increasing sales through the added convenience of a mobile channel? Is it a combination of some of these factors?

To help you identify and articulate the value, here are a variety of perspectives to consider:

1. Customer Value: Is your mobile strategy designed to simply facilitate the "care & feeding" of your customer? Some goals to focus on when considering the "customer value" model of mobile development:

- Reducing the friction in purchasing or repurchasing transactions
- Timely and easy access to needed data
- Retention and loyalty applications
- Faster customer service
- Painless customer support

All of these increase the potential lifetime value of each customer, either by encouraging repeat transactions, or the "stickiness" across your customer set.

2. Reducing Operational Costs: Does mobile represent a way that you can make your company and its resources more efficient? Some solid example results of this type of strategy will promote:

- Increased employee productivity
- Reduced overall expenses
- Reduced time to capture data in the field
- Reduced reliance on manual document processing via mobile document capture

3. Increasing Sales: Can a solid mobile strategy deliver a means by which your sales team is better supported and your sales funnel is more productive? You can gain:

- Improved overall sales process, including prospecting, follow up and conversion rates
- Reduced sales cycle times
- Better educated customers, leading them to buy more/more often/other products
- More timely information delivered to your vendors and distributors
- A way to give distributors tools to sell and promote your product

4. Risk Reduction: Can mobile enablement of some tools help with regulatory compliance? Increasing workplace safety? Capturing corrective actions? Mobile solutions focused on risk reduction can promote:

• More timely accident or exception reporting
• Ease of tracking corrective action
• Enhanced capabilities to monitor and repair equipment
• More efficient means to document field repairs, lease equipment condition, etc.

5. Overall: Your company is patronized for the value proposition offered by its people, products or services. A well-conceived mobile strategy can promote:

• Greater differentiation from competitors
• Better ability to leapfrog any process by an order of magnitude of improvement (for example, can you develop a mobile solution that will eliminate paper in the field altogether?

With this greater focus on your priorities, you should easily be able to make a list of all potential mobile strategies and apps that could support each of the priority areas. Depending upon your creativity, your list will vary in length from "enormous" to "gargantuan" – this is why we need a mechanism to rank these in priority order.

Stack rank your wishlist

Developing a matrix with some ability to weight and rank the ideas is a great way to get started.
Ideally multiple stakeholders are together in a room, hammering out the criteria to rank, and informing the ranking process.
Here are some typical parameters you could use to rank and prioritize:

• ROI
• Cost of development
• Ease of development
• Time to Deployment
• Ease of Adoption
• Urgency of the Problem being addressed
• Other company-specific metrics.

Once the matrix is completed, and your priorities effectively ranked, we can create a roadmap – with a rough wishlist timeline – for what launch targets will be support your objectives.

Go above and beyond

One last exercise at this step: challenge your team to include a definition of success for each deployment, so the value of each potential app is measurable in terms that you can quantify.

As an example, if you are a loan processor developing a solution to make processing more efficient, it's more relevant to target that efficiency in quantifiable terms (for example, we want to do the work in five days that it used to 10 to days to complete).

All of the data collected at this stage -- rankings, rough roadmap, timeline, success definitions and measurements -- are your starting points. Pat yourself on the back.

You have now completed Step 1 in your new mobile journey!

Step 2
Use Case

You may have noticed by now that we are very deliberately looking at your mobile strategy through the broadest lens possible, & then zooming in to focus on the details. The first step that you've just completed will now serve as an atlas of understanding to frame our next, and deeper, explorations. Now, we need to consider the primary use case for each potential app. In our experience, as opposed to users in other countries, especially China, Korea and Japan, American mobile users tend to like mobile apps that serve one primary function.

This is the reason, for example, that Facebook separated the Messenger into its own app, or why Instagram will probably never be merged into the main Facebook app. Additionally, every user or actor tends to have their own app. Facebook Page admins have their own Pages app.

Though it is the consumer-focused apps that gain the most notice, Enterprise apps have made great strides of late – both in terms of how companies have embraced their development and deployment, and how workers have benefited from their use.
A survey found that such enterprise apps boost worker productivity by more than 34 percent. But a Forester study found that, though 33 percent of companies are planning to build Enterprise app stores, a strong majority of employees – 64 percent – criticized their company's enterprise app offerings because of a poor user experience.

This supports the notion that a thorough understanding of use cases MUST precede any app development work. As a starting point, here are some examples of some typical use cases for mobile apps – & the companies that employ these use cases:

Content Consumption

Here, the primary action the user is doing is searching, browsing, consuming, bookmarking, annotating relevant information. Check out the leading Apps that satisfy this primary use case – BBC News, Magazine Apps, Newspaper Apps, Kindle & iBooks, Newstand etc. Typically information is presented in grid, with easy search and categorization.

Content Creation

Here, the primary action undertaken by the user is creating content. For example, the user is out in the field doing an audit of expensive capital equipment, capturing corrective and preventive actions, documenting processes etc. Typically the user is capturing photos, annotating them, tagging them with geolocation, adding notes, checking off lists. The device may be expected to work in an offline mode, and when back online, submit data to a server, or generate PDF documents. Some of the Apps that satisfy this use case are home inspection apps, field audit, field service, etc.

Ease of Transactions

Here, the primary use case is helping the customer engage in commerce with the company. A prime example is Starbucks, whose app allows you tp save your favorite drink, order it and pay for it. But this is not limited to sales type transactions only – helping customers along a long information gathering process – for example selecting and buying a house (Redfin, Trulia, Zillow), closing on a loan application (Vancity), or a requesting a service call (ServiceMaster) are other examples. Uber is another prime service example, where the entire process of

calling a ride, getting an estimated time of service, paying for it and rating the service – all beautifully and seamlessly packaged into one extremely intuitive and amazing App.

Extend your Platform/Product

Though not an individual use case, per se, when primary reason for the app is for the company to be able to extend its platform or product to the mobile screen, service and form factor must be considered. Examples are many, for example Evernote, or pretty much any SaaS product. Most offer some version of their product or service accessible via the mobile.

Marketing

Here, the primary use case is for a company to increase awareness of its products. Realtor apps are a prime example, allowing a realtor company to market their inventory of homes. Movie-based video games are another example. Enterprise conference management apps could be considered a form of a marketing app. Pure marketing apps are difficult to pull off. It is important for the Enterprise to provide some real value to customers – even if that is just entertainment – for the app to have any hope of success.

Sales Support, Distributor Support and Field Sales Support

The primary use case here is to enable your sales force to conduct extended channel sales. The gamut of apps here is pretty large – and this is where Enterprise adoption of mobile usually takes off first. There is, at least in theory, a pretty strong argument for measurable ROI in these cases. Empowering your channel or field sales force to quickly calculate and offer lease or financing options, or to provide firm quotes and proposals in the field, is very attractive and allows you to shorten the sales cycle.

Communications and Social Engagement

The primary use case here is communication between various stakeholders in an organization, or an Enterprise social network. It could be informal communication like WhatsApp, Skype, or more formal Enterprise messaging apps or social networks like Slack or Yammer.

Productivity

The primary use case here is to increase employee productivity – offering ways to track time, record expenses and mileage, host virtual meetings and coordinate other corporate resources. Apps like Expensify, Freshbooks, Workboard, Huddle and GoToMeeting are good examples.

These are just some common examples. Obviously you will find many more that are more relevant to your industry and your specific domain.

Once you have identified the major user cases that apply to your app, study the best of breed in each of these. Closely examine the user experience in all these apps. Remember, it's less important for you to "reinvent the wheel" – especially for Enterprise applications.

It is important that the experience be relevant to your user, yet intuitive to use from the first launch. This is true even for Enterprise apps.

Congratulations! you are progressing quite well in the development of your mobile strategy. Let's take the next bold step by getting an even better sense of the type of user that will derive the greatest benefit from your app.

Step 3

Considerations Regarding Distribution, Devices & Users

In our previous chapter, we helped you to narrow down the use case for your mobile app. We considered some key classifying factors for your app, and got a sense for some solid mentors that have blazed the trail for us.

When targeted correctly, a mobile app – whether focused to consumers or for Enterprise users – becomes an indispensible and reliable tool. Today, only 20 percent of app users – a historic low – abandon an app after a single use, according to a Localytics study.

This trend is abetted by the amount of advance research that goes into matching the right features, with the right platforms, and the right audiences.

For us, the next logical – and critical step – in preparing to develop our work plan is to answer some key questions that support this notion -- regarding whom, specifically, we want to download the app and how they will benefit from it – things like:

1. **What frequency of use do you foresee?** Will this app be used hourly, daily, weekly or intermittently? Will it include some notification/reminder scheme to prompt users to launch and interact with it? Is it crucial that the user engage with it on a habitual basis, or is this an "only when needed/prompted" kind of app?

2. **What kind of performance is needed?** Will this be a "lightweight" app that has minimal content and information handling? Or will you need to upload or stream video and other rich content? Will you need the app to have a large offline cache on the user's smart device?

3. **Are there any device-specific features you will need to use?** Does the app need to use the camera, accelerometer, compass, Bluetooth or other hardware built into the phone? Or will it be dependent upon software interactions with things like iBeacons, Contacts, Calendar and Passbook? Will you need to integrate the GPS for things like Geofencing and other location-based services?

Or, is the experience best actualized on a dedicated, single device? As an example, if the CXO will need to interact with financial metrics and generate graphs, perhaps the app is best formatted in landscape or widescreen, for an iPad or other tablet.

4. **What are the user personas of those who will be using the app?** How familiar is the user with smartphone interaction? How would you define the target audience – are they teenagers? Moms? Your sales team? Building inspectors? Aggregating these user personas – in the form of a "segmentation" study or report – will help you understand the level of complexity that will be tolerated by your user. Also, at this point, consider who you expect to be your early adopters, and consider a focus on them that will allow you to gather actionable feedback for immediate tweaks, and future iterations.

Having an idea of these kinds of requirements will inform in future a lot of the platform considerations you need to account for – as you consider your mobile platform.

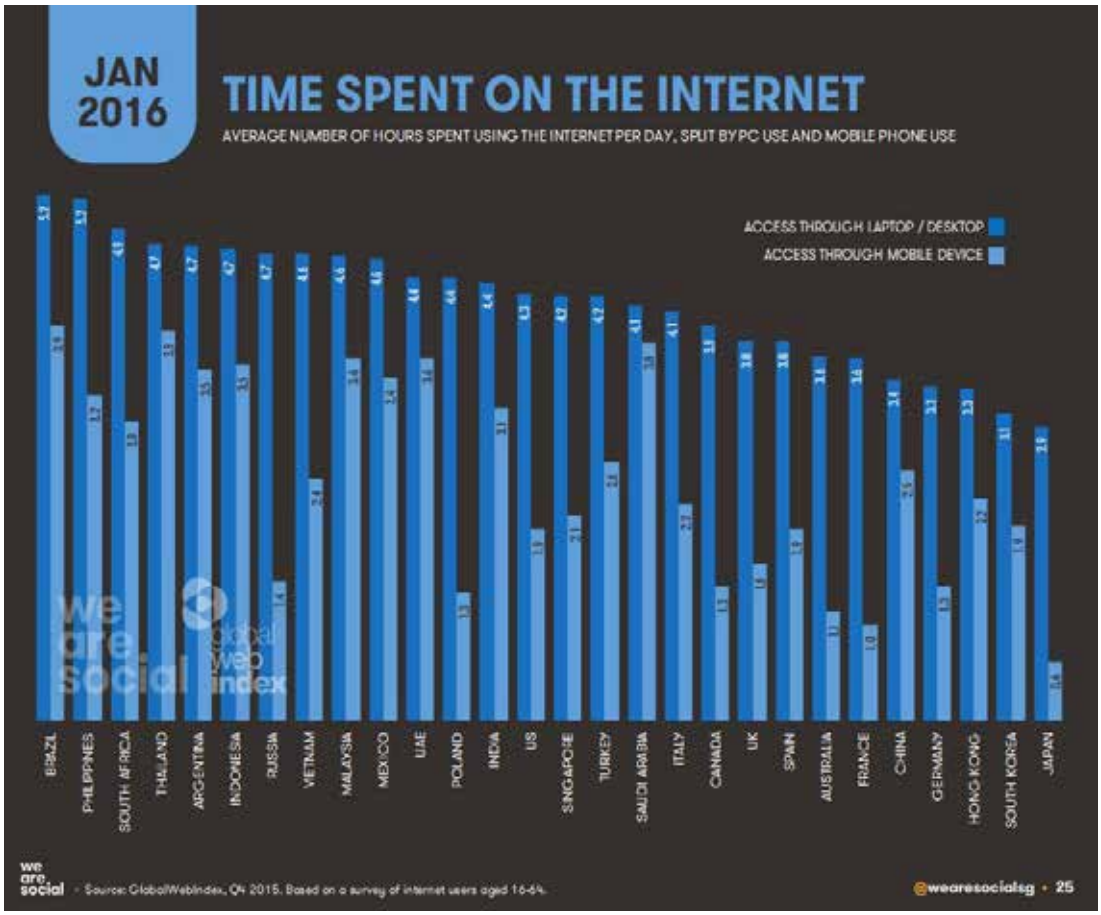
Go above and beyond

If you can get the buy-in from management, consider putting together a Company wide mobile experience, brand and style guide.

That can be a company-wide reference for all departments – to reduce the diversity of user experiences, and user interfaces, especially in employee-facing apps. These documents delineate the look & feel, navigation patterns, menus and other details across platforms for all your applications.

Even if you ultimately engage a third-party developer to build out your apps, you still want to guide them and give them boundaries & constraints. Otherwise, you risk enabling these resources to apply their own creative spin on the product and ending up with something quite different than you conceived.

At this stage, we've arrived at the launching pad for app development. In the next step, we'll explore in detail the myriad technical considerations that allow us to bridge the worlds of concept and reality.



Step 4

Building Your Application

Now that you have targeted your audience and prioritized considerations regarding the distribution of your mobile app, you likely have a sense of who is going to be using your app, on which devices, and how often. The next step is building the app to satisfy these needs.

This step is a bit more technical – but don't let that scare you. We will use the information gleaned the work in the previous chapter when specifying your vision to navigate the path toward app development, and ultimately – launch!

At this stage in the game, one of the first things you will want to decide is whether you want your app to be a native application – developed for use on a particular platform or device; a web app – a client-server software application in which the UI runs in a web browser; or. a hybrid application – developed using web technologies and then put inside a native container.

Let's explore these in a little more detail.

Native Apps

Native apps are developed and coded in a specific programming language to run on a specific platform or device, such as iOS (iPhone, iPad) or Android operating systems. They typically provide fast performance, are very reliable, and generally have access to a mobile device's built-in features such as a camera, calendar, location-based services and address book. In addition, many of these apps are designed to function without an Internet connection.

As you can imagine, a native app is more expensive to develop because it is tied to one type of operating system, requiring the company that creates the app to create duplicate versions in other programming languages that work on other platforms. These programming languages include:

- **Objective-C**, which is a general purpose object-oriented programming language mainly used by Apple for the OS X and iOS operating systems.
- **Java**, a venerable, general-purpose computer programming language used for Android that is concurrent, class-based, object-oriented, and designed to have as few implementation dependencies as possible.
- **C# (pronounced C-sharp)**, a multi-paradigm programming language used in Windows platforms which incorporates simple, modern, general-purpose, and object-oriented capabilities.

Cross-platform development tools – such as Appcelerator Titanium – also exist to help developers write code that can be adapted "on the fly" for multiple environments.

Web Apps

A web application, or web app is a client-server software application in which the UI runs in a web browser. A web application generally differs from a pure website, though this may be a technicality only evident to app and website developers. Content typically is shared between platforms, using a "responsive design" to reformat the information for the varying sizes of displays. Integrated "web applications" – which handle user inputs and interactions – generally run remotely, or independently of the static content; and require a processing and data storage solution on the backend.

Familiar and widely used frameworks for web apps include:

• **PhoneGap** – a free and open source framework for quickly building cross-platform mobile apps using HTML5, Javascript and CSS.
• **Sencha Touch** – a UI JavaScript library that can be used by Web developers to develop user interfaces that look and feel like native applications on supported mobile devices.
• **jQuery Mobile** – a framework that allows for the design of a responsive web site or application that will work easily cross-platform.
• **Dojo Toolkit** – an open-source modular JavaScript library designed to facilitate the rapid development of cross-platform, JavaScript/Ajax based applications and web sites.

Hybrid Apps

Hybrid apps begin their lifecycle through web-based development tools, then are placed within a so-called native "container," such Adobe PhoneGap, which, in effect, run the web application code and package it in a way that allows a device's infrastructure to recognize as an app. This allows for the hybrid app to leverage native device elements such as the camera, contacts, location services, and more.
Additionally, hybrid mobile apps can incorporate native UI elements for use in some situations.

Now that you have addressed the specifics of what you want your app to look like, how you want it to run and on which devices or platforms, and how you want to invest time and money for development - rejoice! Things can start moving forward quickly now. Our next step will walk you through platform considerations – specifically, how to develop a support strategy for your portfolio of applications, from test automation and performance monitoring, to advertising and monetization strategies.

RATINGS COMPARISON OF NATIVE VS WEB VS HYBRID APPS

No	Functionality	Native app	Web app	Hybrid app
1	User Experience, User Interaction & personlization	★★★★★	★★★★☆	★★★★☆
2	Responsive Design	★★☆☆☆	★★★★★	★★★★★
3	Monetization	★★★☆☆	★★★★☆	★★★☆☆
4	Time to Market, Approvals in App-stores	★★★☆☆	★★★★★	★★★☆☆
5	Prototyping Options	★★★★★	★★★★☆	★★★★☆
6	Development Cost	★★★☆☆	★★★★★	★★★★☆

Source: http://www.knowstack.com/web-app-vs-native-app-vs-hybrid-app/

Step 5

Platform Considerations

In our previous chapter, we helped you answer key questions regarding application use & performance expectations to help you understand the technical underlying decisions steering the application development process.

We covered key elements for Native Apps, Web Apps and Hybrid Apps, while highlighting the pros and cons of each.

Now, we are ready to take a closer look at platform considerations – namely, the Enterprise tools and protocols you'll need to manage, support and update your portfolio of applications to ensure they remain relevant, viable, competitive – and functional.

Let's break down some components of a typical support infrastructure, and how each component works to foster long-term success for your mobile strategy:

Automated Testing

Consider this the software of softwares – a package that runs in the background to provide dynamic control of the execution of testing schemes, and then to compare actual outcomes with predicted outcomes and earlier test runs.

An automated testing suite is the best way to increase the effectiveness, efficiency and coverage of your overall software testing program. Once fully optimized, your automated testing program will result in a reduction of cost and complexity of testing across multiple devices and form factors, as well as offering a reliable means of preventing and fixing bugs – all of which will improve the time to market.

Crash Detection

When a computer program or application stops functioning properly, we commonly refer to it as a crash. It is generally the result of single or multiple device instructions running incorrectly.

Pinpointing the cause of the crash, on which device the crash occurs, what action resulted in the crash, and what line of code is causing the failure is absolutely critical to recovering from the issue.

While it is true that all apps will crash at one time or another – you will most certainly want to minimize not only the frequency of crashes, but also the time it takes to diagnose and fix the cause of the crash and prevent future problems.

Performance Monitoring and Testing

Mobile applications offer a direct connection between your business and your customer. Delivering first-rate customer mobile-first experiences can be critical to customer satisfaction and to maintaining your competitive advantage.

Application and backend performance monitoring and testing will tell you if your app is running at desired speed, if there are delays, and where these delays occur (ie: A specific chunk of code? Problem with the network? Backend issues? Device?, etc.).

Performance monitoring and testing will give you the tools to remain ahead of user complaints by spotting performance issues proactively.

Analytics and User Behavior

If you're reading this, you probably already appreciate the important role that data and analytics play in ensuring that the problem your app solves remains relevant.

Though of the utmost importance for consumer-facing apps, every app should have an analytics suite capturing statistics on user behavior.

App developers – and in turn, marketers, function best when they have actionable insights about what your users are doing, what feature (or features) are gaining most utilization, and where they are abandoning the use of your app – if that becomes an issue.

User behavior analytics can also help businesses detect insider threats, targeted attacks or financial fraud through advanced cyber-threats.

Adversting Integration

In order to ensure a consistent brand message across both traditional and mobile
marketing channels – and to employ promotional methods that reinforce both – you will want to consider in-app ads that forge a defined link between the app and your Enterprise's overall marketing efforts.

In thinking about what kinds of ads you are planning to serve – i.e. third party ads, self-promotional ads, or both – you may also want to include partnerships with whom cross-promotions can be of mutual benefit.

Companies with a large application portfolio, or with new applications that can be promoted through existing successful campaigns, are great candidates for these types of partnerships.

If your application is consumer-facing, you will need to decide your monetization strategy as it applies to goods or services offered by your company. Will you offer in-app purchases, house ad support, Freemium (use of the app for free with in-app purchase options), or links to purchase goods or services outside the App?

Deployment Platforms

You will also want to consider how you will make your app available to its intended user
base. Your deployment platform can be a link distributed from within the Enterprise or a downloadable available through any of the consumer-facing app stores.

Regardless of the selected approach for your Enterprise, you can consider Mobile Application Management (MAM) solutions such as Apperian, Air Watch, Mobile Iron, etc. to control access to both your internally developed and commercially available mobile applications.

Backend Infrastructure Testing and Support

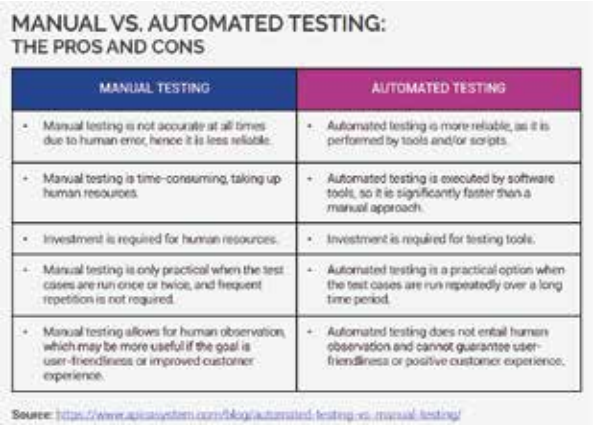
As we noted at the outset of this chapter, mobile applications pose a particular risk for crashing – a result of their continued exposure to large volumes of unpredictable traffic.

These spikes can strain your infrastructure – moreso than web or desktop applications – as a result of location and time dependent usage patterns and the pinging from server to server for information.

You will want to know that you have a backend support infrastructure that can handle the doubling or tripling of traffic without skipping a beat.

Preparing for next-level success

In this chapter, we have introduced you to myriad platform considerations that will inform the ideal strategic mobile pram that suits your enterprise. While assessing usage needs and creating a solid support infrastructure in such a dynamic manner may require a larger slice of the budget, it can also be a key determinant of whether or not you can successfully compete with the leaders in your category.
Our next step will help you examine those human and financial resources that fuel your mobile strategy – taking into account creation-to-launch budget considerations and development personnel needs.



MANUAL VS. AUTOMATED TESTING:
THE PROS AND CONS

MANUAL TESTING	AUTOMATED TESTING
• Manual testing is not accurate at all times due to human error, hence it is less reliable.	• Automated testing is more reliable, as it is performed by tools and/or scripts.
• Manual testing is time-consuming, taking up human resources.	• Automated testing is executed by software tools, so it is significantly faster than a manual approach.
• Investment is required for human resources.	• Investment is required for testing tools.
• Manual testing is only practical when the test cases are run once or twice, and frequent repetition is not required.	• Automated testing is a practical option when the test cases are run repeatedly over a long time period.
• Manual testing allows for human observation, which may be more useful if the goal is user-friendliness or improved customer experience.	• Automated testing does not entail human observation and cannot guarantee user-friendliness or positive customer experience.

Source: https://www.apicasystem.com/blog/automated-testing-vs-manual-testing/

Step 6

In House Development, Skills & Budgets

Now that we have covered the technical aspects of App development and addressed the Platform considerations you will encounter as you move forward, we are ready to help you turn inward, to assess the availability of both development skills and resources within your company to power a positive outcome.

Understanding whether your Enterprise has the human and financial resources available to support your mobile strategy is a critical first step. With larger Enterprises, we often see a temptation to rely upon internal IT personnel to build and manage its own mobile business applications – thinking that this approach will allow greater control over the app environment, enhanced security, and the freedom to customize apps for specific devices and needs.

In addition, organizations securing a mobile strategy for their business sometimes think that, because application development is a huge competitive differentiator, it might be too mission-critical to entrust to a one-time third-party app developer.

Though we certainly believe that our expertise can conform to virtually any type of industry, market segment or development need, we also understand that the digitalization plans of many Enterprises may limit our participation to specific strategies or tactics.

On this notion, it's helpful to take a closer look at the pros and cons of in-house development, along with budgetary considerations specific to your organization, to gain better insights about what you may be able to do now, as well as what ramp you would need to grow your in-house capabilities in the future, to ensure the long-term viability of your App portfolio.

Attributes that favor in-house development:

• In-house talent is readily accessible onsite, and as needed.
• The Enterprise can have greater visibility into the development process.
• Communication is streamlined within the Enterprise.
• Existing IT staff, by their nature, have a greater understanding of the needs of the organization than would an outside contractor.
• Ongoing app maintenance and trouble-shooting can be prioritized.
• Investing in your team's skills prepares you for future development projects, at a long-term reduced cost.

That said, there are some obstacles and ongoing challenges as well.

Of the myriad considerations regarding the cultivation of in-house development talent, we believe that the "learning curve" challenge is probably the most consuming. Learning to code requires an investment of time and mental focus; and training and cultivating a new developer from within your organization is huge consideration that offers no guarantee of a great app at the end of the day.

However, depending on your long-term mobile strategy and application goals, the cost of that risk can be comparable to the cost of a development contractor.

Plus, developing and managing a custom mobile application introduces a number of additional budget complexities for consideration. Referring back to Step 4 of this book, you'll recall that a Web Application is going to be much less costly than a Native App, with a Cross-platform App falling somewhere in between.

Also, while budget considerations are very clear during the build-out phase, have you allocated sufficiently for the type of maintenance and testing delineated during Step 5? How long before you expect to see ROI after launch? Will you be able to hire additional staff if needed for maintenance and support?

All companies change and evolve, as do the needs of clients and consumers. You will want to ensure that your mobile strategy or custom application doesn't become restrictive or less valuable for your enterprise over time, by ensuring you've accounted for the continued cost of updating the platform and adding new features.

Now is the time to determine whether you wish to hire to these capabilities or contract with subject-matter experts.

Even if you do ultimately decide to engage third-party developers to build out your apps, you will still want to guide them and give them boundaries and constraints to remain consistent with your organization's brand and long-term marketing strategy.

Now that we have taken a look at personnel and budget requirements for your mobile strategy, we are only 2 short steps away from "Go"!

In Step 7 we are going to outline key security considerations and highlight strategies to protect your company and consumers.

Step 7
Security

You and your Enterprise have made huge strides on your path to App development, and have invested time and money along the way. Now it is time to ensure the security of your Enterprise, as well as that of your consumer.

Most users typically don't consider security and data privacy when they are interacting with a game on mobile device, or enjoying the ease of paying for goods or services via mobile app.

They are under the assumption that the app is from a reputable company, an app store, or from their employer, that the issuing party has taken precautions to protect the user - so what could possibly go wrong?

Well - any number of things, actually.

With the use of networks sharply on the rise, mobile applications are vulnerable to variety of threats. According to a 2014 survey by Cenzic, 96 percent of tested applications have vulnerabilities, and in another survey in the same year by Cisco, 50,000 network intrusions are discovered each day.

The various paths hackers can potentially take to exploit a mobile weakness can bring down an organization's entire technological backbone. To make matters more complicated, many of these threats go undetected until they reach the server.

As you can see, understanding – and planning –for the specific security needs of your company and your consumers is a key consideration.

Let us now take a closer look at some of the more common security concerns, and the solutions to address them.

Insecure Data Storage

Many consumer-facing apps today store sensitive personal data, such as usernames, email addresses, and passwords. In an alarming revelation - some well-known companies recently confessed that their widely-used apps were storing this type of data in clear text.

The problem with storing data in clear text is that it allows anyone with access to the phone to see passwords and usernames, as well as geo-location tracking points and credit card numbers, just by connecting the phone to a computer.

And since it is quite common for users to employ the same username and password across systems, the very real potential exists for other accounts to be compromised as well.

It is paramount to design mobile apps in such a way that passwords and sensitive account numbers not reside directly on a device, and if they do — to securely store them in an encrypted manner, and marked to disallow backup, or for the data cache to be wiped every time the mobile device reboots.

Encrypted offline storage solutions include:

- **Cookies:** Cookies have been around since the early inception of the Web and were originally intended to associate small amounts of data with the user.
- **Plugin-Based Storage:** Storage supported by Flash, Java and Google Gears.
- **Web Storage:** Convenient offline storage compatible for many applications that uses a simple structure of key-value pairs, such as any JavaScript object.

Weak Server-Side Controls

When an Enterprise creates its first mobile application, it is often exposing existing systems that had not previously been accessible from outside their own networks. That being the case — these systems may not have been tested for proper and proportionate security.

A number of back-end APIs assume that an app will be the only thing that will access it, but the truth is that these servers - be they an Enterprise server, or a third-party system - need to take specific security measures to prevent unauthorized users from accessing data, be that from hackers, or unintended invasions of privacy.

Most mobile applications that handle sensitive data connect it back to a server, rather than storing it on the device. Therefore, you want to ensure that the transit of the information is safe, and that nothing be vulnerable to interception on an insecure Wi-Fi connection. This can be achieved through offline cache encryption and SSL certificates.

Unintended Data Leakage

Businesses are often hungry for the kind of personal information that mobile applications collect, because such information can often help them to personalize their marketing strategies and business goals. However, it is essential that the gathering of personal information and sensitive data not compromise the consumer's privacy in any way by the business, or by any third party. For example, widely-reported media accounts conte nd that the NSA tapped popular smart phone apps to gather huge amounts of personal data. That rendered the app(s) to be considered "leaky", meaning they didn't protect the consumer's data, which included usernames and passwords, as well as age, gender, location, and possibly credit card or bank information.

Broken Cryptography

There is no quicker way to mishandle mobile encryption than for an Enterprise to create and use its own encryption algorithms or protocols. It is best practice to use the most up-to-date algorithms that are deemed strong by the security community, and to use state of the art encryption APIs whenever possible within mobile platforms.

As we have stated — technology is in a constant state of change, and as a result encryption algorithms can just as quickly become obsolete. Without good encryption, sensitive information is vulnerable to hacking.

There are a number of encryption levels available, with the major differences between them being the complexity of the encryption. As a rule: The more complicated the encryption - the more secure your data will be. The flip side to this is that you must be prepared to match that complexity with equal processing power to decrypt the data.

In a nutshell:
More security = More demands on hardware.
The more popular the app, the greater the threat to hacking.
Invest in good encryption, if you want your app to be considered secure.

Security Decisions via Untrusted Inputs

Not only is data leakage a concern, but since a mobile application has the ability to accept and collect data from several resources, it is vulnerable to attackers who may try to modify inputs such as cookies or environment variables, where authentication and authorization are based on these input values. This would allow anyone to bypass your security.

A well-known example of this happened with the first wave of wide-spread

smartphone use – a vulnerability in the very first iPhone OS enabled hackers to listen to phone conversations when those phones were connected to insecure networks. You concern for both businesses and consumers!

The lesson learned: If your mobile application has openings to accept data from external sources, it is vital that it encrypt or secure all inputs used to build the app. Check every inlet for security issues, including the GPS, sensors, camera, and even the platform operating system itself.

The biggest risk with offline storage is that often browsers are accessed my multiple users, such as in homes or Internet cafes. Though not all of these users possess malicious intent, the risk of them viewing the sensitive data of other users remains high.

Once you decide on the level of security you need for your business and customers, it is time to secure the intellectual property of your app development team.

Application Source Code Protection

As your business grows and your mobile application portfolio expands, your Enterprise will want exercise control over its creative content. This will ensure that App developers will retain the right to choose how their ideas are distributed to the general public.

Without protection, innovators forfeit the right to decide whether their code is dedicated to the public as "open-source,", licensed, or kept as a trade secret. Innovations in software can be secured under various categories of intellectual property, which afford developers protections that are granted by law:

- **Copyright**, which protects a particular tangible expression of an idea. In App development or software, that could include source code, object code, and UI.
- **Patent**, which protects the idea itself.
- **Trade Secret**, which covers proprietary information used by a company for economic advantage over competitors. Unlike copyrights and patents, there is no requirement to file or register an application with a government entity.

Use multiple forms of protection to best secure your App and software innovations, and to retain full control of distribution. Once you have found – and plugged – all of the holes that can let nefarious people penetrate your armor, you're nearly there!

Hooray! In Step 8, we will help you prepare to provide maintenance and support for your mobile application that will be critical to protect your investment.

Step 8

Maintenance & Support

We have arrived! The eighth — and final — step in your mobile strategy development. After all of the hard work along the way in preparing a mobile app for launch, you will want to protect your investment by making sure you have a solid plan in place for maintenance and continued support for your business and your customer.

Maintenance of your application is just as important — if not moreso — than the launch event. Bug fixes, upgrades and improvements are central to developing and ensuring customer satisfaction and loyalty.

In the same manner that your Enterprise has carefully considered and allocated a budget for an app's development and launch, you will want to exercise similar prudence in allocating resources for maintenance. Bear in mind — it isn't unusual for maintenance to consume upwards of 20 percent of the original cost of development, on an annual basis.

Another key factor in your app maintenance costs will be which platform you chose to develop your app. If you decided to develop a Native App for each platform, you will need to be prepared for inflated maintenance costs, as you will have to do updates, fixes, and enhancements separately for each device's specific platform.

If you are following the procedures we outlined in this book, you will be dedicated to a constant cycle testing and analyzing the app. However, with the lightning-fast pace of technology and its ever-changing landscape, there will always be something for which your Enterprise had not planned. There may be changes to a smartphone operating system or the app store rules. Or a subset of users may uncover a bug that was unanticipated.

In any case – your Enterprise will want to act quickly to avoid obsolescence.
Maintenance can be broken down into 4 major categories:

Corrective Maintenance

This would include anything that modifies the app after delivery to correct discovered problems, i.e.:

- Bug Fixes
- Compatibility issues with Web browsers
- Security patches to address discovered security concerns

Adaptive Maintenance

This would cover the modification of the app after delivery to maintain its usability in a changed or changing environment, i.e.:

- Modifications to a web site to better function with a newly emerging Web browser.
- Modification to comply with changes made by third parties, such as new API releases made by popular social media applications like Facebook, Twitter, or Instagram.
- Increases in capacity to handle traffic surges to a web site.

Perfective Maintenance

Perfective maintenance is any modification to an application after delivery to improve its overall performance or maintainability, i.e.:

- Changes to servers or networks to make a website faster.
- Changes to software code to make it less expensive to maintain or modify.

Preventive Maintenance

This would refer to any modifications to the application after delivery to detect and/or correct latent faults before they become effective faults, i.e.:

• Patches and software tests, similar to those made to websites on or before the dreaded December 1999 to prevent near apocalyptic system failure on Y2K.

Final Thoughts

As you can see, the development of an ongoing App support plan is critical for the protection and the sustainability of your mobile strategy. The cost of not having a good maintenance plan ranges from the gradual decline of the application, to the total loss of your target consumer base and your financial investment made into the work. At a minimum, your user will want reassurance that you will quickly identify faults and implement fixes.. However to remain relevant, maintenance also includes the development of new features and functionality to enhance the user experience or to increase the capabilities of the application.

Finally, applaud yourself for having the insight to consider a comprehensive mobile strategy. Many companies encounter "paralysis by analysis" — but the cost of doing nothing can often be much greater.
And once you plunge ahead, don't be afraid to experiment or to fail — so long as you are prepared to learn from your mistakes. Iteration and cheap experimentation are the keys to learning what works and what will not work for your Enterprise.

Ultimately, the eye needs to be on the prize. If so, the payoffs can be huge.

Contact Us:

General Inquiries

(512) 551 2773
(512) 535-1650
info@itexico.com

Sales

(855) 602-1949
(855) 501-0100
sales@itexico.com

https://itexico.com

iTexico | Smarter Software Development
Innovation, Near and Now.