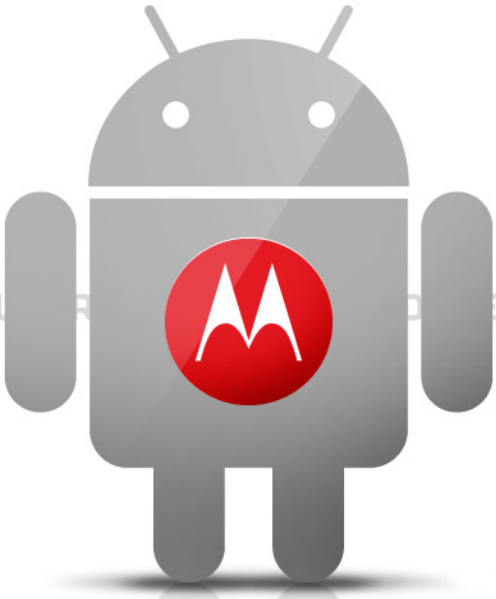
A GUIDE TO MOBILIZING YOUR APPS

BEST PRACTICES FOR ENTERPRISE ANDROID™ DEVELOPERS



MOTODEV
FOR ENTERPRISE

developer.motorola.com/enterprise

INTRODUCTION

Enterprise mobility continues to gain momentum as it engages employees, boosts productivity, and encourages cloud collaboration. A recent IDC report[1] found that more than 50 percent of IT departments plan to spend at least 10 percent of their annual budgets on mobility. And more than 15 percent are planning to spend at least a third of their funds — representing the bulk of their innovation resources.

If your company has made the commitment to enterprise mobility, you may be wondering how designing and developing for mobile differs from desktop development, what in-house experience and knowledge can be transferred, and what skills will need to be supplemented.

In this guide, you'll find best practices for developing mobile apps for the Android platform — from planning to development and deployment. We'll show you how mobile apps and their development are different from desktop apps and what you can do to close the gap between the many skills you already have and the new ones you'll need.

TABLE OF CONTENTS

AN OVERVIEW OF ENTERPRISE MOBILE APP DEVELOPMENT

Mobility brings unique opportunities to enterprises along with some unique challenges for in-house developers. There are top-line considerations in mobile app development that you'll want to keep in mind when creating your first enterprise app for Android.

SECURING CORPORATE INFORMATION

Enterprise mobility raises key data security issues for CIOs, such as how to enable mobile devices to securely and privately access resources behind the company firewall and how to authenticate and authorize mobile users.

Most enterprises enable remote computers to access corporate data stored behind the firewall using virtual private network (VPN) software which is also supported by most mobile platforms, including Android. The drawback with VPN on a mobile device is that all IP network traffic gets routed through the VPN server. So, even that funny online video — when viewed from a mobile device — will demand encryption and transmission resources that may result in device battery drain. It is therefore important to consider VPN alternatives, such as a web-front-end to corporate data, or a remote terminal view into the corporate network (as in Citrix products), either of which can enable access to corporate data through controlled port access and tunneling.

When developing in-house mobile apps, you may even want to consider a hybrid of direct access and web-front-end access to corporate data.

Enterprise developers should also be aware that devices such as smartphones can be easily misplaced and/or stolen. It is vital that any data that your corporate application accesses is stored in a secure manner, either by local encryption, or by wiping it off the device as soon as a session ends. Either way, design the mobile application so that it is not the primary location for that data. The data should also be located behind the corporate firewall. This will ensure that a lost device can be remotely wiped without overall corporate data loss.

For your first mobile enterprise app, it's wise to select a project that does not require behind the firewall access. Mobile access behind the firewall should be an IT project in its own right.

Learn more about how to provide employees secure access to the corporate network from their favorite mobile devices in the MOTODEV article, **Secure Connectivity**.

Learn more about securing your corporate information and network while allowing mobile access in this MOTODEV webinar: **Authentication, Certificates, and VPNs.**

INTEGRATING MOBILE SOFTWARE WITH ENTERPRISE SYSTEMS

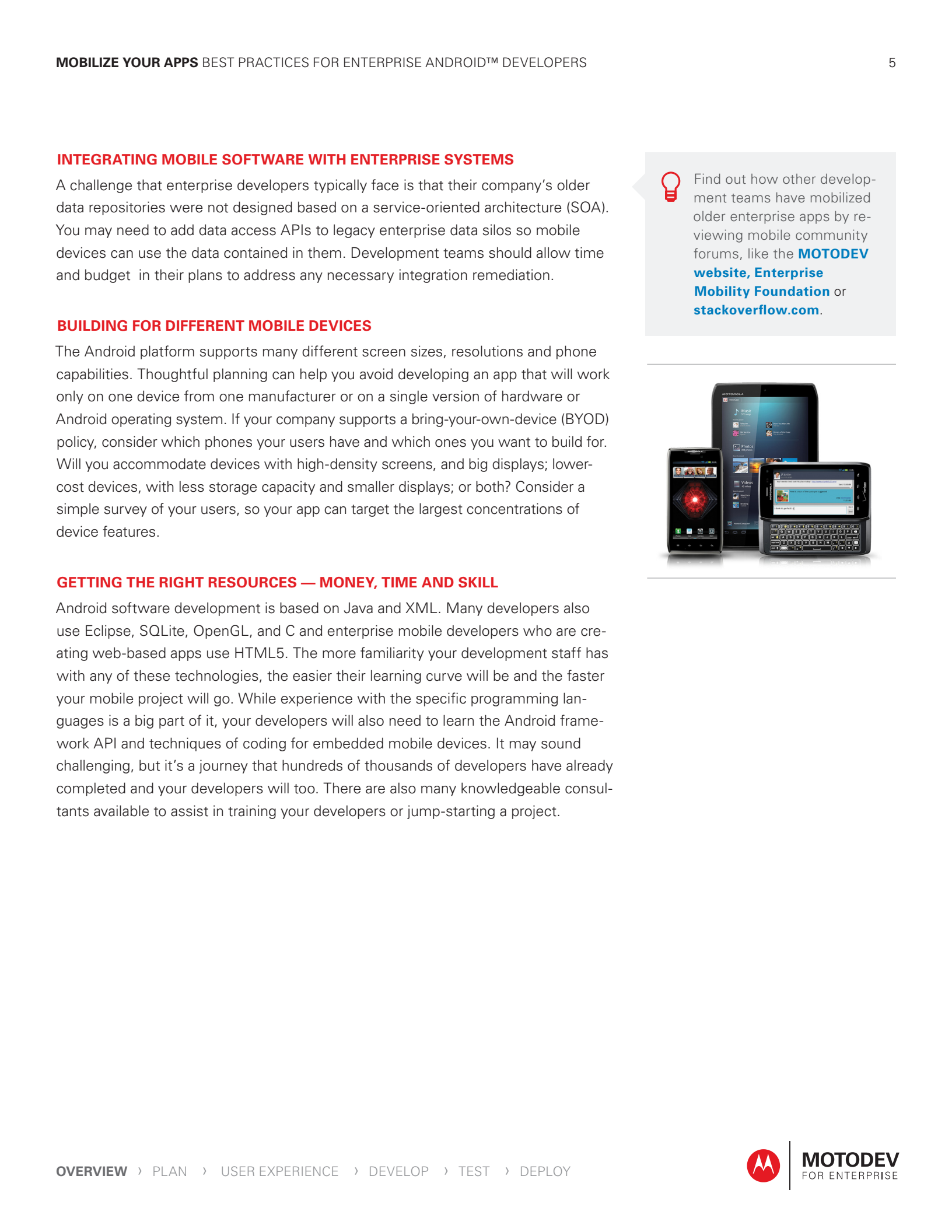
A challenge that enterprise developers typically face is that their company's older data repositories were not designed based on a service-oriented architecture (SOA). You may need to add data access APIs to legacy enterprise data silos so mobile devices can use the data contained in them. Development teams should allow time and budget in their plans to address any necessary integration remediation.

BUILDING FOR DIFFERENT MOBILE DEVICES

The Android platform supports many different screen sizes, resolutions and phone capabilities. Thoughtful planning can help you avoid developing an app that will work only on one device from one manufacturer or on a single version of hardware or Android operating system. If your company supports a bring-your-own-device (BYOD) policy, consider which phones your users have and which ones you want to build for. Will you accommodate devices with high-density screens, and big displays; lower-cost devices, with less storage capacity and smaller displays; or both? Consider a simple survey of your users, so your app can target the largest concentrations of device features.

GETTING THE RIGHT RESOURCES — MONEY, TIME AND SKILL

Android software development is based on Java and XML. Many developers also use Eclipse, SQLite, OpenGL, and C and enterprise mobile developers who are creating web-based apps use HTML5. The more familiarity your development staff has with any of these technologies, the easier their learning curve will be and the faster your mobile project will go. While experience with the specific programming languages is a big part of it, your developers will also need to learn the Android framework API and techniques of coding for embedded mobile devices. It may sound challenging, but it's a journey that hundreds of thousands of developers have already completed and your developers will too. There are also many knowledgeable consultants available to assist in training your developers or jump-starting a project.



Find out how other development teams have mobilized older enterprise apps by reviewing mobile community forums, like the **MOTODEV website, Enterprise Mobility Foundation** or **stackoverflow.com**.

PLANNING

The phases and steps in the mobile app development lifecycle are similar to their counterparts in the server and desktop software development lifecycle, so you can plan your mobile app in the same way, but with the following best practices in mind.

START OUT BY KEEPING IT SIMPLE

Keep your first mobile app project straight-forward so that you can focus on learning the development process. Consider something simple but useful such as a company phone directory, conference room map, or cafeteria menu app.

If you already have an application that is programmed with HTML5, that would be an excellent starting point for a first mobile app. It could be delivered to your users quickly and lay the groundwork for your next mobile initiative.

MEET A NEED OR SOLVE A PROBLEM

Just because there's a desktop version of an application, it doesn't necessarily mean there's a need for a mobile version. Before investing time and effort into app development, determine the business value that mobility will add.

Finding a project that will advance your organization's goals in a significant way may require you to focus your attention on business processes, and partner with a business domain expert in your enterprise. Together, you can identify a process that can be improved through some capability of mobile devices; e.g. the camera, location pin-pointing, bar code recognition, off-the grid communication or processing. Then, match those capabilities to a business workflow that can be made faster, more efficient, more accurate or more effective.

RECOGNIZE THAT THE USER IS KING

In addition to ensuring that there is a solid business need for a mobile version of an app, you must also build the app in such a way that your target users will want to use it.

For any enterprise mobile app initiative to be successful it is important to understand the needs of the user base. There may be several unique groups within a user base, and each of these groups may have specific requirements for mobile apps. Are you

The top three applications categories mobilized in 2011 were:

- Email, calendar and contacts
- Basic intranet access from mobile devices
- Field service scheduling and dispatch

Source: Mobile Enterprise, "The Top Ten Mobile Apps in the Enterprise"

To help you choose the best development path for your initial mobile development projects, watch the MOTODEV webinar: **The Enterprise Dilemma: Native vs. Web Apps**

targeting engineers with technical backgrounds or sales people? Different users will want different interfaces, content and functionality.

If you're unsure of your target user base, human resources applications are great because they typically target all employees in a company. An app for entering time and expenses or checking vacation time would be useful for everyone.

You should also determine the types of devices your users have, so you can plan and design your apps for different screen dimensions and operating system versions. You'll also need to consider the backwards compatibility of your app and which OS versions you're willing to support in deciding which device capabilities to take advantage of.

SCALE DOWN

If you're creating a mobile app, the screen sizes you'll be developing for could range from a 40-inch HDTV to an 11-inch tablet to a 3.5-inch smartphone. It might be unusual to create an app that targets a fixed location HDTV as well as roaming mobile devices, but the Android framework provides considerable support to architects who want to build that kind of system. The Android framework lets an app provide alternative resources, such as image files in several dpi resolutions, with the right one for the target hardware chosen at runtime.

Despite the advances in mobile performance and hardware technology, mobile devices are not configured in the same way that desktop computers are. Today's smartphones have a much smaller memory footprint than a desktop environment and while many phones have a gigabyte of memory, it's still a finite resource that you'll need to keep in mind. Your app must also account for possible mobile network interruptions and battery life limitations, which can affect app responsiveness and usability.

When you develop apps for mobile devices, you must consider the following:

- Limited screen size
- Intermittent network connectivity
- No backing memory
- Limited-duration power supply
- Restricted processing bandwidth

Discover tips for designing for both tablets and handsets:
Supporting Tablets and Handsets

Learn how you can write a responsive app despite hardware limitations:
Designing for Responsiveness

USER EXPERIENCE

User experience design is an important success factor to consider throughout the entire development lifecycle. A good user experience is critical to driving adoption and usage of your mobile app.

GET MOBILE AND THINK BEYOND THE DESKTOP

Mobile app development is not just about making an enterprise app accessible and optimized for mobile devices. Because of the popularity and pervasiveness of mobile devices, users expect to be engaged and delighted by their mobile apps — even their enterprise apps. Think about how you can use the unique features of mobile devices, such as the camera and GPS, to give your app exciting new functionality that is not available on the desktop version.

Learn more about how you can make the most of Motorola's **business ready device features**

NAVIGATE IN NEW WAYS

On a desktop or laptop, web applications often rely on click-through navigation with hyperlinks to new content. The smaller screen size on mobile devices can make "small zone" hyperlinks less effective as users can easily tap the wrong link and become frustrated.

Divide a long form with many input fields into a series of screens that users swipe through horizontally, and save user input at the end of each screen. We call this UI pattern Form Cards.

Include new ways to navigate content:

- Use the action bar
- Leverage fragments for multipane layouts on large screens that adapt to single-pane layouts on smaller screens.
- Take advantage of touch and gestures for navigation.

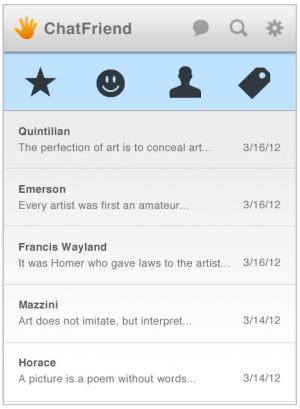


Figure 1
The action bar typically displays the app name and icons that access features.

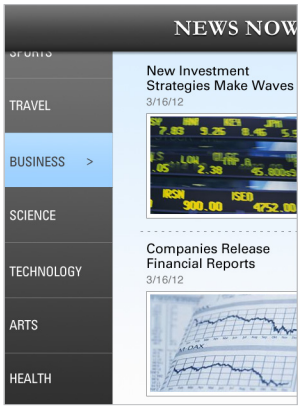


Figure 2
Independently scrollable fragments can display lists, image grids, and other content.

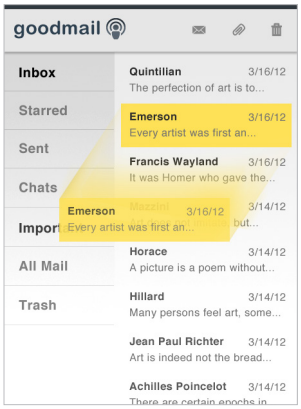


Figure 3
Touch screens are natural and intuitive, because they mimic the way users interact with real objects.

Instead, Android mobile apps offer new ways to navigate through content by means of an action bar — which sits at the top of the app screen and hosts icons for quick access to application features — and fragments, which are adaptable to a multipane layout or a successive single-pane layout. So rather than tapping on links, mobile users can reach new content or more application features in many engaging, natural ways. Users can swipe horizontally or vertically, scroll through lists, or tap an icon in the action bar.

IDENTIFY A USER EXPERIENCE LEAD AND SOLICIT USER FEEDBACK

As most software developers are not user experience designers, it is advisable to engage a user experience lead for your project and to solicit user input throughout the app development lifecycle. You are likely to move between observing scenarios, clarifying requirements, defining solutions, and developing implementations — repeatedly. It's important to have a rapid prototyping and development cycle, where you engage users and solicit feedback early and often. This approach can be adapted for either an agile or a traditional development environment.

One way to incorporate additional user experience expertise for your project is to utilize user interface (UI) patterns, available on http://developer.android.com/design. These patterns offer UI solutions that are easily recognizable to users and are free to you.

> "You are not the user.
> And neither is your boss."
>
> -Suzanne Alexandra,
> Android UX Design Expert

For more user experience guidance, read the MOTODEV whitepaper: **Beyond the Desktop: User Experience Design for Enterprise Mobile Apps**

DEVELOPING

It is in the development phase where you will find the most similarities between creating desktop applications and creating mobile apps. Here are some best practices that will help you succeed where there are deviations.

MAKE THE MOST OF WHAT YOU KNOW

You can use the same kind of tools to develop your mobile app as you use for developing a desktop app. Your integrated development environment (IDE), compiler, linker, modern programming language, debugger, version control solution and more are valid, so all of your software design and development knowledge still applies.

Iterative development — an approach you may already use for enterprise application development — is not only still applicable but highly encouraged. It is often best to get a release out quickly to capture users' attention and then roll out new versions with patches, fixes and new capabilities. If you try to build the "end-all, be-all" version the first time, you may never deliver anything.

NOW APPLY THAT TO EMBEDDED SYSTEMS

Because you can't actually develop apps on mobile devices (yet), you'll need to develop on a desktop instead. This means you're developing on a separate kind of system than the one you're ultimately running your application on — so you're effectively operating it by remote control. You don't have perfect access and control over the over the embedded system (the mobile device) in the way you do on a desktop. There's a new set of skills for mobile app development that must be learned because it requires you to work one step removed.

ADD IN TOOLS SPECIFIC TO MOBILE APP DEVELOPMENT

There are many tools and resources available for Android app development.MOTODEV Studio offers a set of plug-ins for Eclipse, the industry-standard IDE for Android development. This free offering significantly reduces the steps (from 20 to just 6) required to develop an Android app. It has unique capabilities that help both new and experienced Android app developers, including the following:

- A database inspector, which helps you update relational points in your app's database, so you can, for example, easily reset your test data

- Code snippets, which are small amounts of code that can be copied and pasted into your app code to perform a specific task, such get a reading from the GPS
- Wizards, which can help you quickly create a new view from scratch by asking questions and generating much of the new view based on the answers

Find out more and download **MOTODEV Studio for Android**

TESTING

You've spent a lot of time planning and developing your mobile app and now everything you've developed needs to be tested — and under changing conditions. All the devices and operating systems you decided to support need to be tested with your app. This, combined with the different characteristics of the devices, can generate a large testing matrix.

RECREATE THE USER ENVIRONMENT

The level of interruption in mobile devices is significant compared to desktops. Networks can drop and pick back up in seconds or not at all. Connectivity can switch from 4G to 3G to WiFi. Phone calls, text messages and other apps can interrupt your app at any time and push it into the background. When you're testing your mobile apps, it's important to recreate this highly interruptive environment so you know how the app will respond when it loses or changes resources. You can create some of these changes simply by putting the test device in airplane mode.

An additional difference between mobile devices and desktop computers is the touch screen — few enterprise PCs have that interface, so it's important to remember to test the different mobile input capabilities and features. Moreover, powering up and down is not the same between desktops and mobile devices. Once you close an application on a desktop, it is closed. But on a mobile device, you rarely close an application — you simply leave it, so powering down becomes the exercise in closure. And turning the screen off on a mobile device is not the same as powering down, so that needs to be tested as well.

ACCOUNT FOR DEVICE FEATURES

In addition to hardware variances, devices can often support both landscape or portrait view modes, or sometimes they are set to default to one or the other. If you're supporting both views, you'll need to test both views at start-up time and then during runtime when the orientation changes.

There are many more features available across Android devices that you'll want to test for in your app. Check for common feature sets across multiple devices to extend the reach of your application. For example, if your app takes advantage of a camera it will be supported on more devices than if you need a barometer.

In the case of network loss, aim to cache data locally so you can upload it when the network is restored.

"My team has a lot of experience testing the wide variety of apps available on the Google Play™ store and the number one failure we find is apps not handling network connectivity issues — absolute number one failure."

-Manager of Application Testing Services, MOTODEV

VALIDATE ACROSS DEVICES

If you are designing your app for a BYOD user base, you'll also need to test your app on a broad range of user devices. Instead of manually testing different screen configurations and densities, you can use tools such as MOTODEV's App Validator to verify your resources against all the different screen densities. This testing resource can also run a permissions check on your app so that you are not over or under declaring permissions, and can validate application resources and declarations.

Motorola contributes to the Unified Testing Initiative's **Android Unified Testing Criteria** — the industry's first comprehensive set of testing criteria designed to help developers consistently drive quality into Android applications.

Read about testing insights from one of MOTODEV's Android Technology Evangelists: **App Validator and Developer Sanity**

Download it now: **MOTODEV App Validator**

DEPLOYING

With new desktop applications, you can choose your roll out schedule and install your software on your users computers at the time you choose. But with mobile apps, there's a shift of responsibility. You must depend on your users to select, download and install that application on their device. Follow these best practices to accelerate and boost adoption of your mobile app.

DECIDE THE BEST WAY TO SERVE UP YOUR APP

Determining how you want users to download your app is important for two main reasons. First, you want your users to be able to easily download it. Second, you need to think about how you want to manage updates. Particularly if you use iterative development, you will have multiple versions of every app that you create and bugs to fix along the way.

There are many options for deploying your app. You can e-mail the Android application package file (.apk) file to your users or post the app file on a web server and send out a URL link to it. The users can then download it from their mobile devices. While these are the simplest methods, the disadvantage is that you'll need to repeat the process for every update. But for a small user base or for a small number of apps, this may be the best option.

Alternately, you can offer your app through the Google Play™ store, which offers the advantage of auto updates or update notifications. You can limit the installation of your app to only devices with a valid corporate account within your domain. However, this should not be confused with securing your app for your enterprise. Another option is creating your own app store, which has the advantage of auto updates but requires more work upfront. You can minimize the upfront investment with app store solutions from vendors such as 3LM and MobileIron. And if you plan to create a large number of apps, this may be the option that makes the most sense.

Choose the deployment method that best suits you and your users:

- Email or file manager
- Web server
- Google Play store
- Private enterprise market

BE NOSY — USE ANALYTICS

Now that you have deployed your app, you'll want to know if people are adopting it and how to improve the next version. It's very beneficial to track the user, app version, date and possibly the device type for future analysis. This information is key to understanding usage demographics and gauging the severity of problems when they arise.

You can simply track the data in a Google™ spreadsheet every time a person downloads your app. Note that you cannot get this kind of granular information from the Google Play™ store.

Another way to use analytics in your app is to code the app to report statistics to the developer team about its use. The team can then check to ensure its features are working as designed. You can use the same kind of tools as you do when developing a desktop app.

RULE YOUR DOMAIN WITH MOBILE DEVICE MANAGEMENT

Mobile device management can lock down phones similar to the way you lock down PCs in the office. It's part of industry standard best practices to deploy mobile device management, which can require that users install a particular security app on their device if they want to use your app. The security app may require a PIN every time the user wakes up the mobile device, helping to further prevent security breaches of corporate information systems.

Listen to this MOTODEV podcast for information on **Deploying Apps to Your Mobile Workforce**

MOTODEV FOR ENTERPRISE IS HERE TO HELP

The MOTODEV for Enterprise program is designed to make it easy for you to get started developing Android applications for your company and to support you throughout the development lifecycle. An advantage of developing for an open technology platform like Android is that you can benefit from the experience and expertise of a large vibrant development community. As you begin executing your mobile application strategy, you'll find a myriad of Motorola resources available to you through the MOTODEV for Enterprise website, as well as pointers to Android community resources.

Need more of a view into Android? Or are you ready to start requirements planning? Find detailed articles for your next step here: **Android App Development: Wondering Where to Start?**

developer.motorola.com/enterprise

1 "The State of Mobile Enterprise Software in 2011: An IDC Survey of Applications, Platforms, Decisions, and Deployments," IDC, September 2011

MOTODEV FOR ENTERPRISE