



Your Perfect IT Partner

Mobile Application Development:
Challenges and Best Practices

Design | Development | Testing

An increasing number of both mobile devices and potential applications are forcing developers to overcome obstacles through the use of sound practices.

The plight of a mobile application developer these days is a challenging one. On the one hand, development in this space is vibrant and full of opportunities; a spectrum of new devices, from smartphones to tablets, is redrawing the boundaries of what users can do. On the other hand, this new landscape also brings new development questions – including, what devices to target, how to create simple yet effective applications, and how to secure the data that is uploaded and downloaded.

In particular, the so-called trend of the consumerization of IT weighs heavily on enterprise mobile application developers. This trend encompasses many facets. Increasingly, corporate users are accessing enterprise data from mobile devices which may be their own or may be deployed by their internal IT department. That means developers may not know what the target platform is, requiring either a cross-platform or multi-platform development effort.

But the consumerization of IT also presumes an element of ease of use. Corporate users, from the boardroom down, increasingly demand that applications be as easily understood as the applications they use online or for their personal computing needs. As a result, developers need to craft applications that are not only straightforward, but also lack a lot of menus and options that require screen real estate that isn't available on most smartphones.

As if those parameters weren't enough, there is the increasing demand for organizations to interact with customers via online applications on smartphones. Whether the industry is finance, transportation, retail or others, organizations are looking for ways to cement the customer relationship through mobile delivery of updates, offers, and alerts. Newly available in the customer relationship arsenal: so-called disposable applications that are time-sensitive, relating to specific events and promotions.

As there are many application scenarios that developers face, there are an equal number of elements that they must consider in mobile application development, including data access, security, offline capabilities, and backend integration. For mobile applications to be as useful as possible, they must transparently synchronize data with backend systems.

There are also many development environments. Developers can also choose from either native development tools for each of the major mobile devices and platforms (including Apple® iOS, Android™, Microsoft® Windows® Mobile and Microsoft® Phone 7, Symbian®, RIM® BlackBerry®, Java®, Linux®, and Meego™) or cross-platform environments PhoneGap™ and Appcelerator® Titanium).

What kinds of mobile development environments are available, and what are their pros and cons? And which best practices give developers a better chance to achieve success?

Addressing the Challenges of Enterprise Mobile Development

When developers sit down to map their goals for the application, they must determine how they are going to address the aforementioned challenges. The decision is based on the usage scenarios they're trying to accommodate. Not surprisingly, many of these challenges overlap. That's why developers must look at each one in context.

Data Access

Where is the enterprise data users need to access? Should it be accessed through a Web browser, or through a native application running on the device? In some cases, the data might exist solely in a backend database. That requires connectivity, but the state of wireless coverage is such that, beyond one's own office (or sometimes a well-covered metropolitan area), you can never assume a network connection. To address this, developers have to determine whether some data will be stored in the device's native file system or accessed solely through a browser (in which instance the device will most likely store no data).

For instance, do you want a salesperson to be able to download updated inventory data onto a mobile device and head out into a territory with spotty connection.

strength? That requires a synchronization capability to help ensure that once the salesperson is back within range, they can update again.

Most operating systems can accommodate the disappearance of a network connection, as well as the handoff between a broadband wireless connection, such as 3G or 4LTE, and a local Wi-Fi connection. But the application must understand when a connection is dropped, so that it can roll back to its lastknown data.

Security

Every mobile device manufacturer understands the importance of security, and offers accommodations for it. For developers, however, that means each operating system has its own custom architecture.

Sometimes different mobile platforms provide varying degrees of support for securing data stored on the device. At the same time, not every mobile application needs the same level of security, and users dislike multiple log-ins to access data on the device itself or back-end data. However, the responsibility for protecting data falls squarely on the application developers' shoulders, so they must incorporate encryption into the application down to the file system level. back to its lastknown data.

Encryption helps to ensure that data cannot be pilfered from the device, even if it is lost. But at the same time, developers should also consider other options for preserving and protecting data. These include programming backup capabilities for ensuring data is archived on a periodic basis, as well as creating remote "kill" or "wipe" capabilities so that the disk can be erased if it goes online after it's reported lost by the authorized user.

User Experience

This is especially important in cross-platform scenarios where screen sizes may not only differ but offer the option of portrait versus landscape. How is the screen real estate best used? What features should be available? Developers have to remember that users on the go are usually looking for specific information, and determine how to make that information as easy to find as possible. They also have to differentiate between internal and external users. A consumer application requires clarity and simplicity, and frequently offers a "gotta-have wow" factor. Applications for employees, in keeping with the precepts of consumerization, should also be clear and simple productivity tools.



Mobile Development Options

In order to address these challenges and concerns for developers, tools vendors have crafted multiple development environments, including both native tools and cross-platform tools. Native tools are designed solely for use with a particular operating system, while cross-platform tools give developers the flexibility to create an application that will run across multiple mobile devices.

Native Development Tools

These enable developers to create applications that run on specific platforms, such as Apple® iOS, Android™, or Microsoft® Windows® Phone 7. The decision to use one or the other relies on how deeply developers want to link the application with the underlying operating system, as capabilities in one operating system may not be available in another. In order to take advantage of specific capabilities, developers may target the operating system and the associated development environment to create an application with those features.

One advantage of using native development tools: applications will run more smoothly on whichever mobile devices use that operating system, which can represent a vast market. If a company has standardized

a vast majority of users on one or two mobile platforms, in fact, it may be better to develop native applications for that majority of users. That let developers take advantage of platform functionality. The alternative is developing a cross-platform application that serves everyone, but has more limited functionality.

One disadvantage of native applications: there may be stringent requirements for admission into public app stores. Apple Inc., for instance, requires that developers submit iPhone mobile digital device applications for testing within Apple to facilitate such compatibility.

Cross-Platform Development Tools

These cross-platform tools generally fall into two categories themselves: Mobile Enterprise Application Platforms (MEAP) or Mobile Consumer Application Platforms (MCAP). The two sets of tools are not as differentiated as they were originally, though, because some of the MEAP vendors – including Antenna Software Inc. and Kony Solutions Inc. – are also offering MCAP toolkits.

In many ways, these two options resemble the choice between build-it yourself or a full-fledged toolkit in other segments of the application development world. For instance, MEAPs tend to have more full-fledged development environments, with a wider variety of traditional tools such as graphical user interfaces, version control, and workflow. They tend to have more integration tools and gateways to third-party services (such as Facebook and Twitter), as well as better technical support capabilities.

In addition, because MEAPs focus on the enterprise segment, they strive to incorporate stronger security capabilities, taking into account that the applications will be used to access back-end corporate information.

The MCAP category includes such tools as PhoneGap™, Appcelerator®, and AppMobi®, and leans more toward the open-source world. They incorporate three key technologies: HTML5, Cascading Style Sheets (CSS), and JavaScript. Using open-source capabilities allows developers to create mobile applications at a lower cost, but also requires them to do a greater amount of work on their own, and do it with a command-line rather than a graphical interface app stores. Apple Inc., for instance, requires that developers submit iPhone mobile digital device applications for testing within Apple to facilitate such compatibility.

In order to incorporate a browserbased application using a crossplatform tool, developers can wrap it in a native wrapper format, basically embedding a browser screen within the native application which then renders the HTML5/CSS/JavaScript page.

Which application development strategy developers choose really depends on the application itself. Will it be browser-based, with little or no data saved on the mobile device? Does it require capabilities native to the operating system? Does it require security features, or support capabilities, that require a MEAP? Or can it be created using an MCAP? Once developers, in conjunction with line of business representatives have answered these questions, they'll be ready to begin.

Mobile Development Best Practices

No matter what the answer to the foregoing questions, some commonalities apply to development efforts. The best results come from applying the following best practices in the design and deployment of mobile applications. Some of these, of course, relate to any application development effort, but most take into account the vagaries of mobile development.

Methodology

Determining the design methodology is key to mobile application development, especially in a cross-platform environment where multiple efforts may be underway simultaneously. Even if the development toolkits are different, developers should verify that they can use common testing tools, project management and reporting.

Development Skills

Developers should confirm that their team has the proper breadth and depth: breadth in the needs of application area being targeted, and depth on the particular platforms being used.

Design

Developers should take into account the most appropriate architecture for mobile applications. The best method is to develop a layered application, where the functionality remains consistent across platforms (while still conforming to each.

platform's look-and-feel), with modular pieces plugged in underneath. Within those modular pieces, retain identical business rules and application logic.

Creating re-usable platform components can help speed up application development. At the same time, though, especially with cross-platform development, developers must determine which features they will need to add. For instance, in native development, the operating system can notify applications about such events as message arrival and power levels. In a cross-platform environment, developers will need to add such feature.

API Management

Developers should institute a strict policy regarding use of components across platforms, which facilitates maintenance and extensibility across applications on different platforms. For instance, even though the web services library on Android™ and Apple® iOS will be implemented differently, the functions and their underlying code are the same.

Security

By definition, mobile devices are vulnerable. They upload and download data wirelessly, in potentially insecure locations. They're small and can be misplaced. Think carefully about security, not only incorporating the platform's inherent security capabilities but also using other

tools such as encryption for sensitive data. (Remember to balance this with power consumption). The ability to remotely wipe data from a lost device is also crucial. On the reverse side, data preservation precludes the creation of an easy or even automatic backup mechanism so that users don't have to think about backing up stored data.

Testing and User Experience

Application testing (especially with real users) is often overlooked in development schedules, but the fact that mobile users tend to need information on a time-sensitive basis means that applications should be more reliable than desktop applications. Applications should be tested not only for usability, but for power consumption to verify they're not battery hogs. They should be tested with multiple browsers and on multiple carrier connections to confirm that they accommodate both online and offline usage.

A key part of testing is the user experience, which can help make an application more successful by maximizing acceptance and usage. With a short amount of time to develop and limited screen real-estate within which they have to communicate usability, developers must devote time to researching the most effective text and graphics. For global developers, the challenge is enhanced by the need to create graphics that are globally recognizable as well as culturally sensitive to create graphics that are globally recognizable as well as culturally sensitive.

API Management

Because mobile applications do not run on an always-connected environment where IT can monitor clicks and other usage, it's crucial to incorporate analytics features that track how users interact with the application. This will not only help identify ongoing technical support issues, but also reveal whether users are interacting with the application in the fashion developers expected. Analytics can also show which features aren't used, so that they can either be removed or made easier to access. Developers should also incorporate capabilities such as crash logs to track what's gone wrong on devices as they're being used out in the real world.

Feedback Mechanism

The flip side of automated application analytics is a feedback mechanism for users. Don't rely solely on automated capabilities, but develop a capability—whether through e-mail, social networking or other means (even through the application itself)—for users to report bugs and offer insight into what they like and dislike about the application. This will not only help reduce technical support issues, but also provide insight for application updates. Consolidating this feedback will show which features or capabilities the most users are demanding.

Conclusion

Turning Challenges into Opportunities

Sustainable application development and delivery excellence requires the implementation of five steps :

1. Optimize development and testing efforts by accurately identifying which platforms to target, whether for internal or external deployment. This facilitates a better return on development and testing investment.
2. Employ consistent and reliable development practices across all leading platforms. This facilitates consistency in look-and-feel of applications, reusability of code components, and more-easily programmed updates.
3. Collaborate extensively. The application development team should incorporate coding, testing, graphical design, and user experience skills to help applications to fulfill users' needs.
4. Tailor distribution mechanisms appropriately. This involves creating capabilities for both initial downloads through managed application stores, periodic updates, and integrated device management for security purposes.
5. Track emerging technologies and maturing standards. With mobile technology still advancing, developers should look for ways to make applications perform even more efficiently, whether in terms of speed or features.

For more information on how SNT Infotech can help you achieve your mobile application development goals, visit www.sntinfotech.com/mobile-application-development-companies-in-india.html

Contact Us



If you need to contact SNT Infotech Pvt. Ltd., please use the information below to get intouch.

By Phone

Our telephone numbers are
Call Toll Free: 1800 200 4313
Land Line: 0120 4109397

By Email

You can email us at
info@sntinfotech.com

Head Office

SNT Infotech Pvt. Ltd.
101, 1st Floor,
Alpha Tower Sector-9,
Vasundhara, Ghaziabad
(U.P) 201010



About SNT Infotech Mobility Services

SNT Infotech Pvt. Ltd. is focused on enabling its clients to achieve breakthrough growth throughout the rapidly changing mobile ecosystem. SNT Infotech Mobility Services offers five mobility services including consulting, software services – applications, software services – devices and platforms, managed services, and business integration services. These are designed to help organizations embrace business to employee (B2E), business to consumer (B2C), business to business (B2B) and machine to machine (M2M) business opportunities. SNT Infotech offers mobility software services across a wide range of industries and platforms, including Android™, Apple® iOS, Blackberry®, Linux, Meego™, Symbian, Windows® Phone and Windows 8.

About SNT Infotech

SNT Infotech Pvt Ltd was started in 2006 at Ghaziabad, Uttar Pradesh India with a pool of talented creative designers, software programmers and marketing managers.

SNT Infotech Pvt. Ltd is a ISO 9001 :2008 Certified professional web design company providing affordable and personalized web designing, website development, website maintenance, web hosting and web marketing / Search engine optimization services.

Our mission encompasses the following areas when working towards your objectives:

We are committed to understanding the needs of your organization when delivering our services. We fully appreciate the demands of business and will work with you to understand your exact requirements whilst taking care of all the issues for you so there will be no 'hassle' involved. You can be as involved or as detached from the process as you like. We will happily meet face to face to discuss and develop your preferred designs and to understand your needs.

For more information on SNT Infotech Pvt. Ltd. Mobility Services please scan the 2D barcode.

