Best Practices in Enterprise Smartphone Development

Rhomobile White Paper

This white paper was written by Adam Blum, Founder and CEO of Rhomobile

12/16/2010

Best Practices in Enterprise Smartphone Development

Usage of modern smartphones, as exemplified by recent powerful devices such as Apple's iPhone and Google's Android devices, has exploded recently. The usage on these devices is dominated by local native apps (versus mobile web pages). Over 7 billion apps have been downloaded from the iPhone App Store. There are over 200,000 apps on the iPhone App Store and over 100,000 apps on the Android Market.

The success of local native apps over web apps raises some new questions in how to build optimal mobile apps. Most of the information technology industry has focused on how to best build web apps (mobile and otherwise) over the past 15 years. In effect these are truly centralized in their logic and information architecture. Depending on your perspective, local, rich client, native apps are a new development in the industry, or they can be thought of as a resurgence of the truly distributed computing model. In either case it is worth examining how best to approach what, for most developers, is a new development problem – how to build optimal mobile apps[1].

There are several emerging guidelines that developers are following when building native smartphone apps. Most of these are quite distinct from best practices in either web apps, mobile or otherwise. It's also worth noting that the guidelines discussed here are really for informational smartphone apps (for enterprise or consumers). These guidelines do not apply for games, for which an entirely different set of best practices applies.

Leverage Device Capabilities

Rule number one in smartphone development is to fully exploit the device capabilities. Smartphones are more than small laptops. All modern smartphones ship with a wide array of "senses": sight (the camera), hearing (microphone and voice phone connections), touch (as in touchscreen), and a sense of location in space (GPS). The best smartphone apps take advantage of these capabilities.

These apps make use of multi-touch to zoom in and out, GPS to determine their location, and the camera to record images and videos. In addition, these apps

[1] Note what we won't be doing here, however, is making the case for native, local, mobile apps versus web apps. This paper assumes that you are already interested in building native, local, mobile apps (the predominance in usage of native apps versus mobile web apps is already quite well established).

integrate with other capabilities of the phone such as the phone dialer and the contacts database.

Task Oriented

The best informational apps are task oriented and use those powerful device capabilities to do something with your smartphone that cannot easily be accomplished with desktop or web apps. Modern enterprise apps, such as SalesForce CRM, allow you to not only keep track of all of your customer data, but also study all of your current product line, examine feedback from your customer base, and produce complicated sales report all in one application. Users of the modern smartphone, however, are not looking to have access to all of those functions in one app.

Modern smartphone users want to complete tasks. Therefore you should break down your comprehensive back ends into small but powerful task driven apps. Using this SalesForce CRM example, you create one app that gives your sales team access to their opportunities and leverages native smartphone mapping to give them turn by turn directions to their next appointment. Or, you might create another app to give them access to a report builder that uses voice recognition technology to let them add notes to their reports on their way back to the office while their interactions are still fresh in their mind. A third app might just give them access to customer feedback reports and include the camera capability so they can take a picture of the customer's defective product or interesting product use while on the customer's premises.

Context Sensitivity

Because of the latency necessary to navigate in mobile, it is best to take users directly to the most likely task that they may want to perform in the app. Given that an app can perform many tasks or manage multiple business objects, this "guess" of the appropriate task may or may not be correct. But, getting it right most of the time will create a more usable app.

In our CRM example the app could take the sales rep user right to the account that they wish to edit based on their location determined via GPS. An app can also take the user back to the last screen viewed when the app was last used.

Avoid Typing

Modern smartphones are dominated by "soft" keyboards. And when they aren't soft, the keyboards are limited and small. The most usable apps use many clever techniques to minimize typing. There are many methods available to do so:

- Use far more select wheels and radio buttons than most web apps do, even at the cost of flexibility or more app data maintenance.

- Use the device’s knowledge of location for filling in geographic information.
- Use predictive text techniques such as Swype.
- Use voice recognition, camera, video, barcode scanning and other powerful device capabilities to collect information.

Support All Devices

To reach the maximum potential user community, app developers need to be able to target all popular smartphones. There are a variety of techniques for building single apps that work on all smartphones. These include cross-platform frameworks, which are burgeoning in popularity. With modern frameworks, developers can develop using their web skills and the browser component to render user interfaces on the underlying SDKs (e.g. the UI WebView control and Objective C for the iPhone) while maintaining access to the device.

Synchronized Local Data

If the app is an informational app, the best practice is to allow that information to be accessible whether or not the user is online. There should be some way of ensuring that the necessary information is pulled down from any backend application servers, kept current based on changes, and resynchronized back up to the central server when appropriate. Historically, apps that only support “live data access” get far less usage, especially when the app supports creating, updating and deleting data.

Synchronization solutions can either rely on the device polling or on some form of a push solution. Polling-based sync drains device battery life and results in stale data. Push-based synchronization is better and can now be done as modern smartphones now support push natively in the operating system (iPhone, Android and BlackBerry now all have native push capability).

Offline Usage

Beyond data synchronization, which allows offline usage of critical information, there is the general problem of making sure that apps can be used efficiently and completely when offline. Data synchronization solves part of the problem; however, in general, the developer should pay attention to make sure that the app’s most important operations can all be performed whether or not the user is online. Not all apps ensure this, but the best ones emphasize this ability.

Handle Varying Metadata

Perhaps the subtlest problem with the reemergence of *truly distributed local applications* running on modern smartphones, is handling changes to the backend applications schema. With a web application, however, when the data schema changes, the application's full stack is changed: the database, the business logic (usually in the form of a web applications controller) and the interface (the web app's view).

With smartphone apps running locally on a remote device, it's difficult for the central application developer (generally of a web app) to ensure that all remote smartphone apps are updated. Thus, there is usually a need for smartphone apps to handle changes in the underlying schema of the app or database that they connect to in a more sophisticated way.

This problem has been a big hindrance to the widespread availability of smartphone apps for enterprise applications on the smartphone "app stores." The few developers that have been successful have handled this with some custom way to communicate metadata changes to the remote smartphone app. This is an area of emerging research and discovery among smartphone app developers.

Conclusion

In general smartphone app usage is rapidly expanding. There are a new set of emerging guidelines for how to build the best possible smartphone apps that app developers should carefully consider before building their app.

Rhomobile's Rhodes framework and RhoSync app integration server make it easy to follow the guidelines outlined in this paper. Whether or not you use our products, addressing these issues will result in the best breed of smartphone apps. For more information about how Rhomobile can help you achieve your mobile development goals, please email us at sales@rhomobile.com.