

MOBILE DEVELOPMENT

COHERENT SOLUTIONS WHITE PAPER

by

Maksim Belov,
Chief Technology Officer

November 2012



INTRODUCTION

Mobile application development is exploding. Consider that by the last quarter of 2012, about 35 million applications were being downloaded daily around the world. Research firm Gartner Inc. estimates that by 2015 applications for smart phones and tablets will outnumber PC projects by four to one, and Forrester Research predicts that the mobile app market will grow to \$54 billion by then.

Just about every business wants to keep up and cash in on this developing industry. Whether it's to attract and engage customers, or to help employees be more productive, mobile apps are the IT project du jour, and

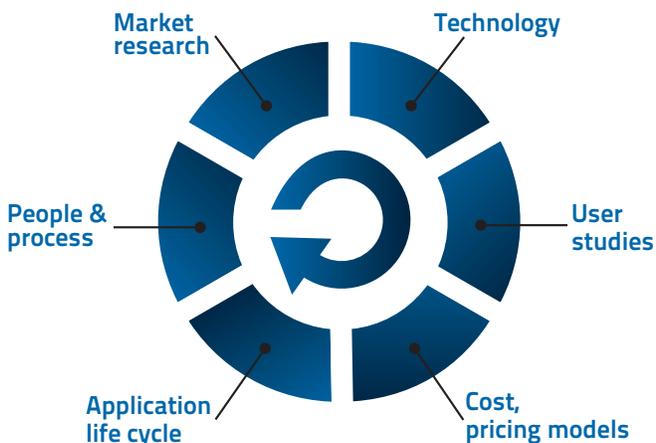
developers are being inundated with orders for new apps. While the technology is there, success is not a guaranteed outcome for all these new initiatives. This white paper discusses this brave new world of mobile app development and addresses:

1. What technical people need to know to evaluate the readiness of a mobile app strategy.
2. The current and future state of technology for mobile app development.
3. Coherent Solutions' methodology and experience with developing successful mobile apps.

COMPONENTS OF A VIABLE BUSINESS STRATEGY

The definition of a good strategy will differ from organization to organization. Larger, more mature organizations typically require research and business case documentation before moving forward to develop a new mobile application. This documentation may include market analysis and competitive research, as well as technology evaluations and pricing models. Once these have been addressed and objectives determined for the app, such organizations may go through prototyping, user studies, exit strategies and more.

For smaller, more nimble companies targeting new and emerging markets, it may be enough to identify a specific market need or gap in service. The ability to move fast is a competitive advantage for these companies, and understanding their own core competencies and markets is generally endemic within these organizations. However, these more agile companies need to have thoroughly defined their business objectives for the mobile application. The goal of this whitepaper is to provide technical people with tools and knowledge for determining whether a mobile app proposal presented by a business is based on a well-defined strategy and is ready for development.



Assessing mobile app objectives & readiness

In the excitement to get into the game with a cool new app, the most important question regarding readiness of the mobile app is sometimes – surprisingly – not asked. That is, what does the business expect the new mobile solution to do for customers, prospects or employees? Follow-up questions to this include: Does the business want to reach and acquire new customers? Engage and retain existing ones? Does the business even have existing customers? If so, is this app going to wow them with something they cannot do or access with some other mobile app?



The mobile solution proposed needs to offer something more than currently available apps. Another front-end for a blog/professional community, or a tool to post to users' social media accounts is not compelling on its own. It's like all the other hammers on Home Depot's shelf. If it doesn't offer something new or of value, why should a user download it?

Mobile technologies open up a new frontier for innovation, allowing applications to follow users and not the other way around. This is enabling truly useful smart features. User experience can be enhanced via simple yet powerful capabilities based on knowing things like the user's location, spatial context, current interactions,

and schedule. With this information, developers can create features ranging from sophisticated reality augmentation scenarios, to more simple cases such as reporting the device's last location as the battery is getting low.

Enterprise apps are a bit of a different animal. The desire to adapt to a new solution is built into employees' jobs – if they want to retain those jobs. For instance, mobile information workers will quickly embrace apps that help them access and capture current, accurate information, as well as collaborate with colleagues and customers in real time. The desire to be more productive is assumed.

That said, technology and features are still important to enterprise users who want and expect a rich experience. For instance, while penetration of social networking tools like Yammer and Jive is still fairly low in the workplace, acceptance of new interaction patterns will grow within organizations and their mobile product strategies will eventually need to include a social interaction piece.

As in early web development, some mobile apps have been created as a result of the insertion of mobile devices into daily life. Good examples include Instagram and Flipboard. These mobile-first, or mobile-only products deserve special consideration. With no ability to map to more traditional, richer web experiences, these apps must deliver top notch, intuitive performance. These applications often require more complex back end support to create the rich experience required at the user interface.

Employees and customers are increasingly attached to their handheld devices, and most would prefer to use one device for both their professional and personal lives. The proliferation of BYOD (Bring Your Own Device) in the workplace presents new security issues for business. Mobile device management and security in an enterprise environment should be a critical component of the strategy.

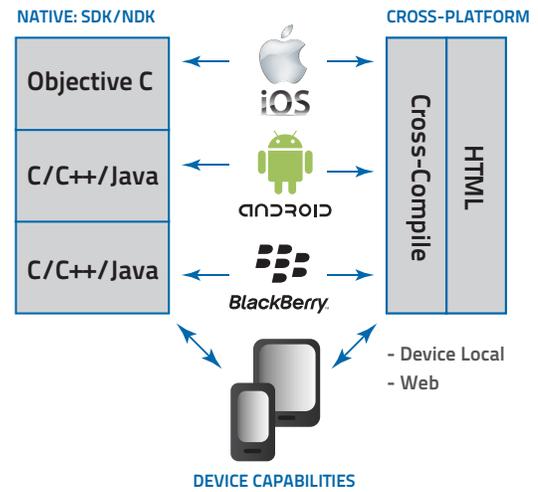
WORKING WITH CONSTANTLY EVOLVING TECHNOLOGY

Mobile application development is such a dynamic space encompassing a wide range of technologies that developers need to stay abreast of technology updates as they are happening. Gartner's Hype Cycle for Emerging Technologies provides a good measure of where some of the most anticipated technologies are currently, as well as where they are headed and in what timeframe. Developing a mobile app requires evaluating risks and rewards across a spectrum of variables that could change and vary the outcome. Looking at Gartner's Hype Cycle it is clear that introducing an app with technology at the peak of inflated expectations is likely to result in some disillusionment on the part of users before the technology, and the app, plateaus into productivity. Large organizations with deep pockets may be able to ride this out, but small organizations risk losing their customer base in this trough. Less well-funded organizations need to measure the risk of introducing trendy but unproven new technology with the need to get to market fast, and the possibility of great reward.

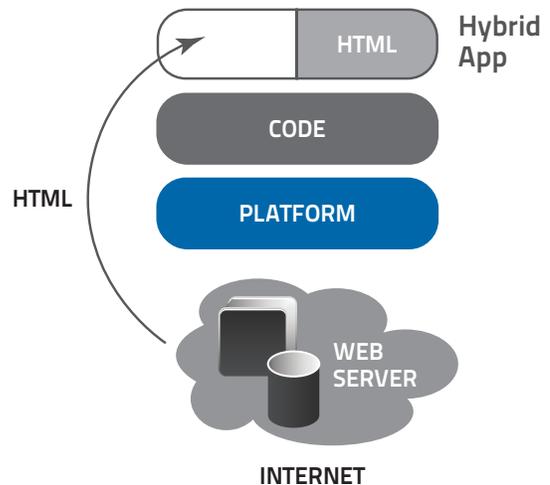
The day of the iPhone 5 is a good example of this scenario. Tech pundits were predicting that this version would include Near Field Communication technology. After all, NFC was past the peak of inflated expectations and more and more Android phones are including it –though without too many practical applications, as concerns about its security loom. Apple dislikes being second in technology, preferring to invent their own versions – usually with a defined purpose that serves the market strategy for their devices. If a newly introduced app relied on NFC in this generation of the iPhone, the entire product would be invalidated.

Native vs. cross-technology app development

Mobile applications can be created as native or cross-platform solutions. Native development utilizes Software Development Kits designed to support each platform, or device, on which the app will reside and operate. Cross-platform apps rely on cross-compiling framework for translating portable source code into platform-specific native code, or using an HTML/JavaScript-based solution to represent the UI as HTML.



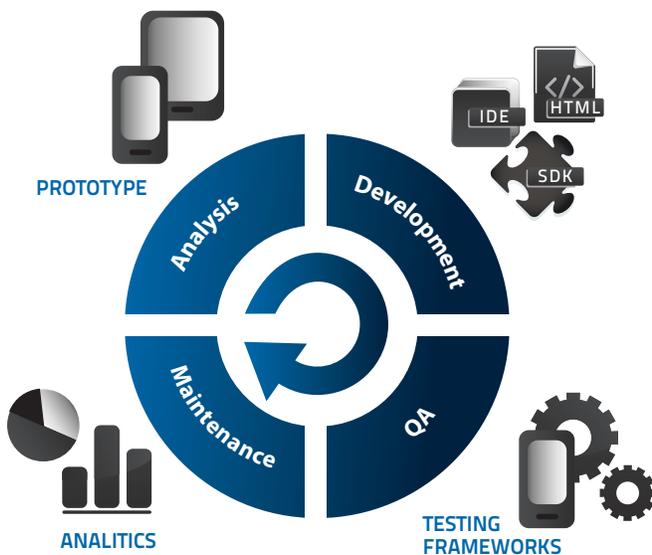
HTML 5 has been a driver in cross-platform development. Its runtime is supported by major mobile browsers and provides rich UI, data semantics, storage and communication capabilities. HTML applications may be designed as pure HTML apps with content coming from a server, or as hybrid apps combining content packaged within the app's resources along with native UI and logic.



Each of these approaches has benefits and drawbacks. Native apps have direct access to device capabilities, such as GPS, accelerometer, compass, camera, address book, etc. With these and other functions residing on the mobile device, the app performs quickly and reliably. There is no need for internet connection and no waiting or threat of losing connectivity when launching an app. Native apps rely on the device's standard protocols, providing users with a familiar experience using recognizable icons and prompts.

A recent survey of open source developers found that over half use native SDKs as these are established and proven development tools. The richness of tools to cover the full development lifecycle is a major contributing factor when choosing a technology. These include, but are not limited to:

- UI rapid prototyping and design
- Debugging
- Source code management
- Team collaboration
- Support for device emulation with hardware platform capabilities (location, accelerometer, multi-touch)



The major benefit of cross-platform development is in saving time and money to develop the app. Developers can use languages that may be more familiar to them—such as Java, C#, HTML/JavaScript/CSS—across all supported platforms, with the majority of the code written only one time for all device types.

As cross-platform solutions have matured, most of them provide access to many native platform capabilities. So non-native apps do not necessarily equate to a lack of access to these features. Making logic portable across different platforms also makes supporting different form factors with each platform easier. For example, the PhoneGap framework exposes a JavaScript-based Application Program Interface (API) to a hybrid application to interact with:

- Accelerometer
- Compass
- Camera
- Address book
- Notification mechanisms
- Media playback
- Local storage

Note the last bullet. The ability to remain functional offline is no longer limited to native or cross-platform compiled apps. Hybrid apps can do this through the capabilities of the hosting framework and mobile web apps get local storage support through the HTML 5 standard.

Selecting the technology platform can be influenced by product strategy as well as technical considerations. For instance, cross-platform development is often the choice for UI prototyping, even though the UI can be somewhat unpredictable across devices and browsers. But the time savings realized by only writing code once (in many cases) more than makes up for the greater amount of user testing. And for mobile web or hybrid apps, content is served from the web, where updates can be handled at the source.

Performance considerations

The inherent strengths that cross-platform apps bring to the table play against them in connectivity and performance. While these apps are conveniently isolated from platform and device-specific implementation of the networking stacks and memory management, they also lose control over these facets of the app. This results in a limited ability to handle network timeouts and communication errors. Direct app performance may also be impacted by the quality of code generated from a compiler, or limitations exposed by a third party accessing native capabilities. The latter is the case for the JavaScript engine on the iOS platform where only mobile Safari has access to the new Nitro JavaScript engine. The rest of the app may only use older, slower versions of the engine. While Apple’s official reason for this is security, protecting the ecosystem and market, as well as encouraging native development on iOS, are also significant drivers for their policy.

If there is one thing that can negate an excellent user experience and features, it is poor battery life. Power and memory management should top any list of factors when choosing a development path. A good app optimizes its use of local device hardware resources such as:

- Networking stack
- Location services
- Accelerometer
- Storage

It is also a best practice when coding to avoid:

- High CPU utilization
- Constant polling
- Unnecessary screen updates

Power and memory need to be aggressively managed from both hardware and software perspectives. Cases that verify power consumption should be included in the test suite and executed on a regular basis. Additionally, network connectivity, device form factors and capabilities, and mobile OS compatibility should also be tested. This is especially important on the Android platform where hardware and software fragmentation is so significant. Android test strategies should not only identify all of the scenarios and risks, but also a proper toolset to allow testing across the wide variety of device types and setups. It can really be inefficient to manually test applications across several hundred possible devices and configurations.

Usability

For many years the challenge of mobile app development was in cramming all the user interface functions into devices with small screens, limited buttons and no keyboards. Today it is widely recognized that the real challenge is presenting the right amount of information and controls at the right time. This is almost never equivalent to squeezing in as much as possible.

Thus, good utilization of precious screen real estate and limited input capability is the goal. The vast majority of users interact with apps in a dynamic or at least an informal environment, with many external distracters.

A good app allows users to devote laser-like focus to the task at hand. Even if a screen can fit a large number of UI primitives, the ability to focus on individual tasks makes an app friendlier. The iPad's success is partially due to this strategy. Even though the larger screen allows for productive multi-tasking, users can choose to focus. The creative use of real-world patterns – like turning a page, pinching or stretching items, or arranging information in layers that mimic a stack of paper – enhances the feeling of working in the physical world. Apps that include the creative use of interactions around multi-touch, gesture support and more will score high on usability.

At this point it is worth reminding readers that while technology can enhance a user experience, it should follow rather than lead the application. It is imperative to first identify the app's business objective. Once that is established, interesting technology that can provide a new kind of user experience can be applied. Some mobile first apps, like those relying on GPS capabilities, might seem contrary to this notion because technology made the application possible. But, in fact, the organization's use for the application—tracking people or packages in real time, mapping routes in progress, etc.—is the lead objective. The technology makes the business purpose possible, but it serves that purpose.

Security

A lot of mobile applications being developed today are extensions of existing enterprise systems, made available to employees on the go. These apps do a wonderful job of extending the company's important functions or business intelligence to users beyond the confines of the brick and mortar workplace. Most of the data stored in these systems is proprietary or at least sensitive in nature, requiring organizations to develop security processes that extend to mobile access. Further complicating security is the move toward more BYODs (Bring Your Own Device) in the workplace as employees prefer to use just one smart device for their professional as well as personal lives. With less control over the devices, corporate IT departments need new tools for assuring security on these far-flung devices.

Some of these include:

- Sandboxing of application data.
- Local encryption of data on devices.
- Use of PINs to protect access to an app, or after a period of inactivity with the app.
- Transitioning of an app into background mode when not being directly accessed.
- Providing the ability to remotely wipe data on a device.

It may seem that at least of these issues were addressed during Web 2.0 cycles, but care and forethought should be taken when securing information for mobile devices. The unique nature of the devices as well as the applications themselves calls for careful scrutiny of this issue with every new project.

Analytics

Application developers should utilize mobile analytics to understand how users acquire and use mobile apps. While developers need not present specific strategies for structuring data collection and analysis at the time of a project's proposal, mentioning this capability is a good practice. Data collected should cover the full range of user experience, from installation to usage scenarios to uninstallation. Data tracked must also be compliant with the app's privacy policies, as well as standards and regulations such as HIPAA.

App Distribution

Finally, we address the state of mobile app distribution. Specific distribution policies vary between platforms, with iOS being the most restrictive and controlled. The only way to deploy a native or hybrid app to an iOS device is through the App store. This requires packaging the app for publishing and going through a submission/review process administered by the platform vendor. Native apps for Android and other platforms may also go through their respective store distribution systems, but also allow ad-hoc distribution schemes. Mobile web-delivered apps are free from constraints imposed by application stores.

Like the initial release of an app, updates are subject to the same publishing constraints. Only exclusively web-delivered apps can offer updates directly from their servers. Mobile web applications provide significant advantages in that area because of the seamless and frictionless application distribution method and ability to easily present different UI to different users based on a defined strategy for scenarios like A/B testing and optimization.

Larger enterprises are beginning to develop their own internal app stores for employees to access. This gives them control over user experience, intellectual property, security and release/update cycles. Ever restrictive, iOS deployment requires that an enterprise get a license from Apple to create its own app store, but other platforms are more open for enterprises. A typical enterprise solution goes beyond app distribution and provides a full set of mobile device management features.sds

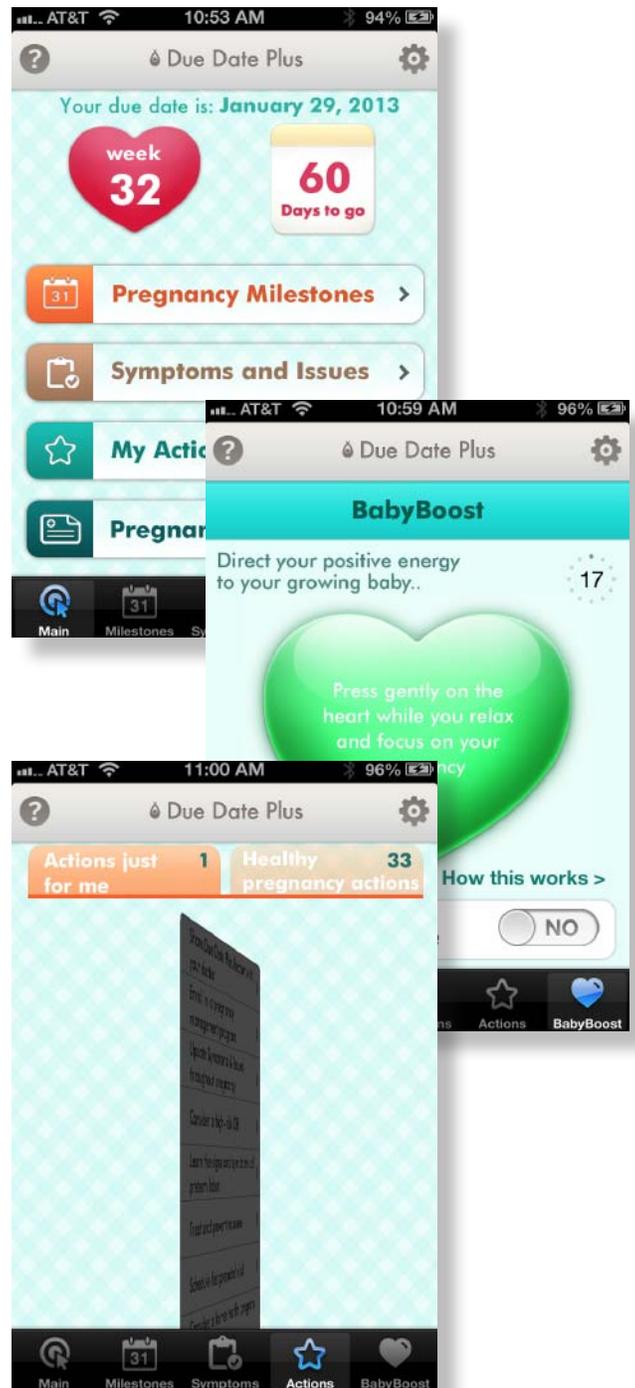
User interface

With such a high priority on user interface in mobile apps, we have created a separate process for developing this feature. Working with subject matter experts and project stakeholders our team:

1. Defines the overall look and feel of the UI.
2. Creates a prototype for key user interaction scenarios
3. Determines different screen types based on navigation maps and wireframes.
4. Designs the main screen types and form factors of supported devices and target platforms.

Multiple alternatives may be developed as appropriate and screen designs are created to take advantage of new UI paradigms associated with touch/gesture interface support and making use of Responsive and Adaptive Design principles where appropriate.

At Coherent Solutions we utilize familiar tools, such as Adobe Photoshop or Axure, for screen design and prototyping, while continuing to add new, mobile-specific solutions that allow designers to quickly and efficiently create designs that support multiple form factors.



SUMMARY

As the newest frontier for application development, mobile smart devices offer nearly limitless potential. Device attributes like GPS, multi-media, ubiquitous connectivity, and more make everything from high-impact gaming graphics to augmented reality solutions to complex, enterprise-grade systems possible. No wonder so many developers and enterprises are eager to jump on board with cool new apps aimed at promoting products and services, engaging customers and increasing productivity. The challenges for technical workers developing these apps include:

- Developing reliable paradigms for assessing app objectives and readiness.

- Staying abreast of evolving technology that constantly enhances and alters the playing field for development.
- Finding and vetting reliable suppliers of technology.

Coherent Solutions has been developing mobile applications for over 10 years, keeping up with changes and new technology tools throughout that time. Our experience has given us insight into these issues and we welcome your inquiries. We invite you to learn more about Coherent Solutions' software architecture, application and mobile development, and quality assurance services by visiting: www.coherentsolutions.com.

Minneapolis, MN

612-279-6262

www.coherentsolutions.com

New York, NY

212-225-8282

Minsk, Belarus

+375 17 294 51 61

