WHITE PAPER • MARCH 2018

ca technologies

The Science Behind Five-Star Mobile Applications

A guide to leveraging analytics for customer-centric mobile apps

CA App Experience Analytics

Table of Contents

Executive Summary

Challenge

Building a successful mobile application may seem like performing magic. There are currently more than 5 million mobile apps, and tens of thousands of new ones are released each month. Many fail utterly in the market, despite the large investments companies make in developing them.

Given the stunning size of the mobile application market, as well as anecdotes about seemingly worthless apps drawing huge investments, one might conclude that building a successful mobile app requires luck more than skill and careful planning.

Opportunity

Mobile success is not magic. Mobile developers can and should adopt a systematic, analytical approach to building mobile applications. Success in the mobile market is not the result of luck, despite appearances. It is the fruit of careful, deliberate attention to customer needs, and an application development and management strategy that enables those needs to be met.

Benefits

An analytics-based mobile development strategy is important not only for standing out from the crowd in a tremendously competitive market, but also for avoiding shortcomings in mobile app delivery that could harm your business's image and lead to a loss of customers.

This white paper explains how companies can create five-star mobile apps and avoid making usability, functionality and performance mistakes in mobile application delivery. It begins by outlining the obstacles that hamper the development of successful mobile applications, then provides tips and strategies for implementing an analytical approach to mobile application delivery.

SECTION – 1:C

Why Mobile App Development Is Challenging

About 25 percent of the mobile apps the typical user downloads will be used only once. The vast majority of mobile apps last fewer than 90 days on users' devices before they are deleted entirely. Two-thirds of mobile apps fail to achieve more than 1,000 downloads in the first year after their release.

Why is the outlook for mobile app success among consumers so dim? It's not for lack of investment. Companies typically spend around $200,000 or more developing a mobile app.

The failure of most mobile apps is instead the result of the following challenges, which make it difficult to design mobile applications in ways that please users.

Providing the Right Services on the Right Platforms

In many cases, mobile apps are developed alongside Web or PC apps that provide similar functionality. Developers sometimes make the mistake of trying to include all of the same services in each version of the app. This approach does not always work because it can lead to mobile (and wearable) apps that are too complex or that have features that are a poor fit for the screen sizes, input methods and other hardware characteristics of the devices.

In other cases, developers may fail to take full advantage of the hardware features that are offered on one type of device but not others; for example, a mobile application might be able to leverage the GPS functionality that is built into most mobile devices in order to provide a better customer experience, whereas the PC version of the application cannot do this because most PCs lack GPS hardware.

Instead of attempting to shoehorn as much functionality as possible into mobile applications, developers should carefully plan which features make sense based on the types of devices they are targeting, as well as the way their users prefer to interact with the devices and software.

Maintaining a Consistent User Experience

Even if developers choose not to offer all of the same features on all of the device platforms they support, they must nonetheless strive to deliver a consistent user experience across different platforms. Many customers switch routinely between different types of devices. A poor user experience on just one device can drive them away from all versions of an application—and lead to negative feedback about how the mobile version of an application failed to deliver the usability of its PC-based equivalent, for example.

Large Number of Operating System/Browser Combinations

There are tens of thousands of different types of mobile devices, as well as hundreds of different versions of Web browsers that might run on a mobile phone. To make matters even more complicated, mobile operating systems and browsers are updated on a constant basis.

Attempting to support all the possible browser and operating system combinations that customers could potentially use is simply not feasible. Developers must choose to focus on certain types of software environments to target and ensure that their software runs as expected in those environments.

Choosing the right browsers and operating systems is a challenging task. The most popular software configurations among one demographic group may be very different from those preferred by another.

Multiple Development Models

There are multiple approaches to developing mobile applications. Some mobile apps run natively on the device's operating system. Others are Web apps that are accessed through a browser. In some cases, mobile apps are hybrid apps that combine both native and Web-based functionality.

Deciding which development model or models to adopt for creating a mobile application is challenging. It involves evaluating the ways in which your users can best interact with your software, as well as technical considerations, such as which types of programming languages and libraries developers can use to implement and maintain the application most effectively.

Mobile Device Mobility

Mobility is a core feature of mobile devices, but it also creates special challenges for building effective mobile software. Application designers must deliver a consistent, reliable user experience on devices that are constantly moving, experience significant fluctuations in network bandwidth and have limited data storage and energy resources.

This challenge is made greater by the fact that most consumers lack the expertise to determine the root of performance problems in mobile applications. If a mobile application fails to deliver an excellent user experience due to a problem such as poor network connectivity, users will be quick to blame the company that developed the application, even though it has no control over the reliability of the mobile network they use.

SECTION – 2

An Analytical Approach to Mobile Development

Responding effectively to these challenges requires more than a haphazard approach to mobile development. Mobile application designers and developers must systematically evaluate what their users want, how they interact with devices and which types of technical challenges the application must be designed to handle in order to deliver an excellent user experience.

In other words, mobile developers must take an analytical approach to app development. An analytical approach can enable an application to receive high ratings from consumers, delight customers, keep users coming back and help your company to stand out from competitors.

An analytical approach involves the following strategies and practices:

Systematic Understanding of What Users Want

Mobile developers can't rely on ad hoc user feedback in forums or bug reports to determine what users want in an application. They must instead collect detailed, systematic data that allows them to determine which features and functionality users expect.

They can achieve this insight by "peeking over the shoulder" of users with tools that visualize user touch and gestures within applications, provide video playback of user sessions and so on. This data should be collected on a continuous basis and made available constantly to developers in order to help plan new application features and updates based on user behavior.

Understanding of What Users *Don't* Want

Determining which features users don't want in an application is just as important as identifying those they do. Developers need to be able to trace which features in an application are not frequently used so they can remove them if users find them unattractive, or improve them in the case that a usability problem is keeping users away from them. Maintaining functionality that users don't want leads to bloat, inefficient use of developers' time and potential software security risks.

Tracking of Customer Dropoff

Most customers who abandon an app never take the time to tell developers why they left. Even among those who do leave feedback, the information—which usually takes the form of generic statements such as "confusing interface" or "slow to load"—is often too ambiguous to provide useful insights to developers.

For this reason, application developers should rely on software tools that allow them to determine exactly when and why users stop using an app. They must be able to track, on a screen-by-screen basis, what happens before the application is closed or the user stops interacting. With this insight, developers can target particular aspects of the app for improvement.

Detailed App Debugging Information

When a mobile application crashes or underperforms, developers need detailed, context-aware data about why the problem occurred. They need to know not just which parts of the application code were associated with the crash, but also how environment factors such as the user's location, network carrier and device hardware profile may have contributed to the problem.

This allows developers to get to the root of performance problems quickly and ensure that certain groups of users are not saddled with performance issues resulting from environment factors that an application can't control.

Omnichannel Visibility

In order to deliver a consistent user experience across all platforms, developers require an omnichannel view of the user journey. In other words, they need to know when and why users switch from one type of device (such as a smart watch) to another (such as a phone or PC). By analyzing this data, developers can determine whether certain versions of their software are underperforming, as well as which types of platforms to prioritize based on frequency of use.

Understanding of Device Software Configurations

As noted above, there are a vast number of operating system and browser combinations that users might run on their devices. Rather than attempting to support all possible combinations, which is highly unfeasible, or choosing the most popular overall configurations, development teams should determine which software environments are most popular among their target users.

To understand why this decision is so important, consider how Apple iPhone mobile device usage varies between different groups. The market share of the iPhone in the United States is around 33 percent, but it is only 2 percent in India. iPhone use also varies significantly between genders, with women being more likely than men to use an iPhone.

Because the types of devices consumers use can vary widely, developers should be able to collect and analyze data that shows which devices consumers are using to access their services, as well as how service performance varies between different types of devices.

Continuous Feedback

We live in a DevOps-dominated world. Most organizations strive to deliver software to users on a continuous basis. In order to ensure that continuous delivery chains do not outpace the ability of developers to keep up with user expectations, user feedback must be collected and shared with the development team on a continuous basis.

The importance of continuous feedback can be easy to overlook. The DevOps movement has tended to prioritize delivery speed, affording less attention to software quality and performance. Delivery speed is one factor in providing an excellent user experience, but it is not the only one. Continuous feedback is an equally important component of a healthy continuous delivery chain—and one that plays an important role in delivering a quality user experience.

SECTION – 3

Streamlining Mobile App Analytics

Identifying the right analytics tool

When implementing the strategies described above, mobile app development teams should look for tools with the following capabilities:

- Real-User Insights: Collect data about application usage, in a granular fashion, in order to reveal how users interact with particular features in an application and how usage patterns vary between different devices and demographic groups.
- Performance Data: View end-to-end performance of mobile and wearable apps all the way to back-end systems—including mainframe and the cloud—to address incidents before they impact your user experience.
- Code-Level Visibility: Allow developers to identify root causes quickly. Developers should be able to determine whether an app performance problem was the result of poor code, poor design, infrastructure failures or something else.
- Design Visualizations: Gain insights into application design and flow through analytics visualizations that aggregate touch and gesture data. These insights will help you understand how users are engaging with your application and optimize the app's design and navigation.
- Single-Pane-of-Glass View: Provide a single pane of glass for analyzing application user experience and performance issues.

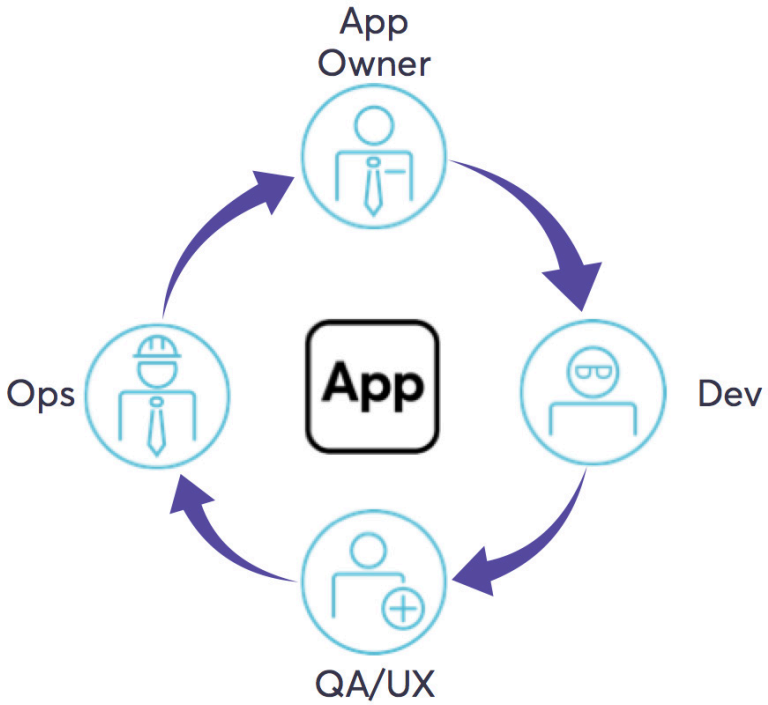
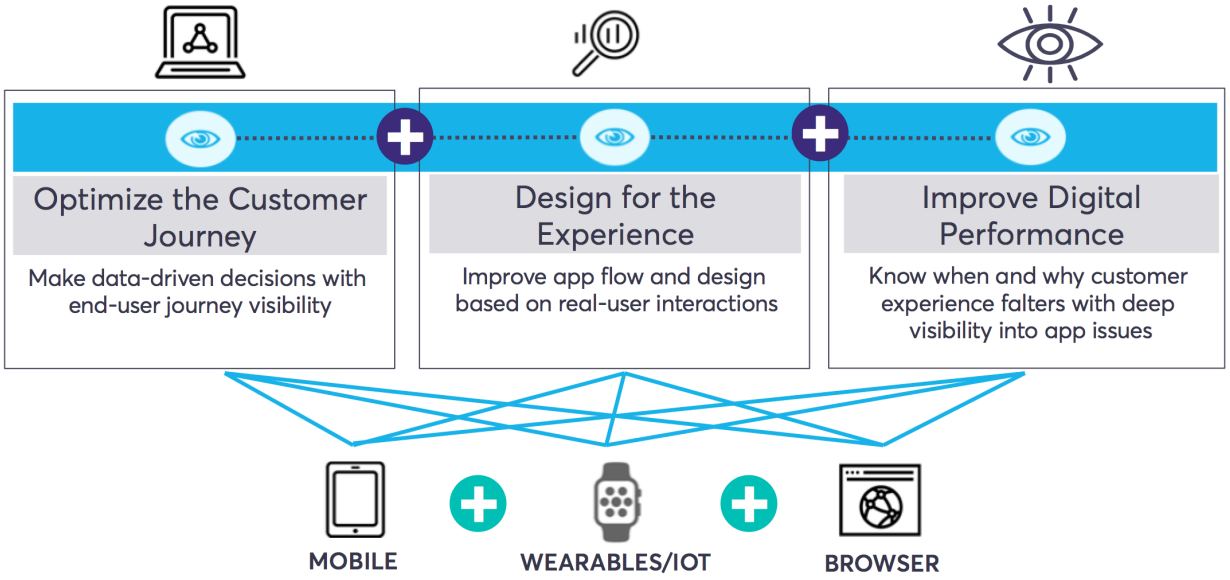


FIGURE A.
A single-pane-of-glass view facilitates easy interpretation of data and enables efficient communication between teams—a particularly important consideration in a DevOps-focused organization.

These features ensure a streamlined process for turning analytics data into usable insights when designing and improving mobile applications that earn five-star reviews. Choosing an analytics solution with these key features will enable you to:

- Optimize the customer journey
- Design for the experience
- Improve digital performance

One Platform for End-User Analytics



SECTION 4:

Conclusions

Developing mobile applications that stand out from the huge crowds that populate app stores requires prioritizing the customer experience. To keep customers happy, mobile app delivery teams must adopt a careful, analytical approach to the features they implement, the devices they support and the way they leverage feedback about user preferences and behavior.

This strategy is the only reliable means of achieving consistent five-star reviews. Without an analytical approach to mobile app development, your company's investment in mobile software is likely to result only in underused applications and lukewarm user feedback.

An analytical approach to mobile development becomes possible when software delivery teams adopt tools that provide detailed data about user behavior and preferences and make the data available via intuitive, efficient interfaces.

CA App Experience Analytics provides the insights you need to deliver a five-star app experience for your users, enabling you to:

- Optimize the customer journey and make data-driven decisions
- Design for the experience with analytics and visualizations to improve app flow and design based on real-user interactions
- Improve digital performance and provide the insights to know when and why customer experience falters with deep visibility into mobile app issues.

FIGURE B.

The overview screen provides an at-a-glance view of application usage, performance, and crashes and errors, so you can quickly evaluate your customers' experience and drill-down into application issues.

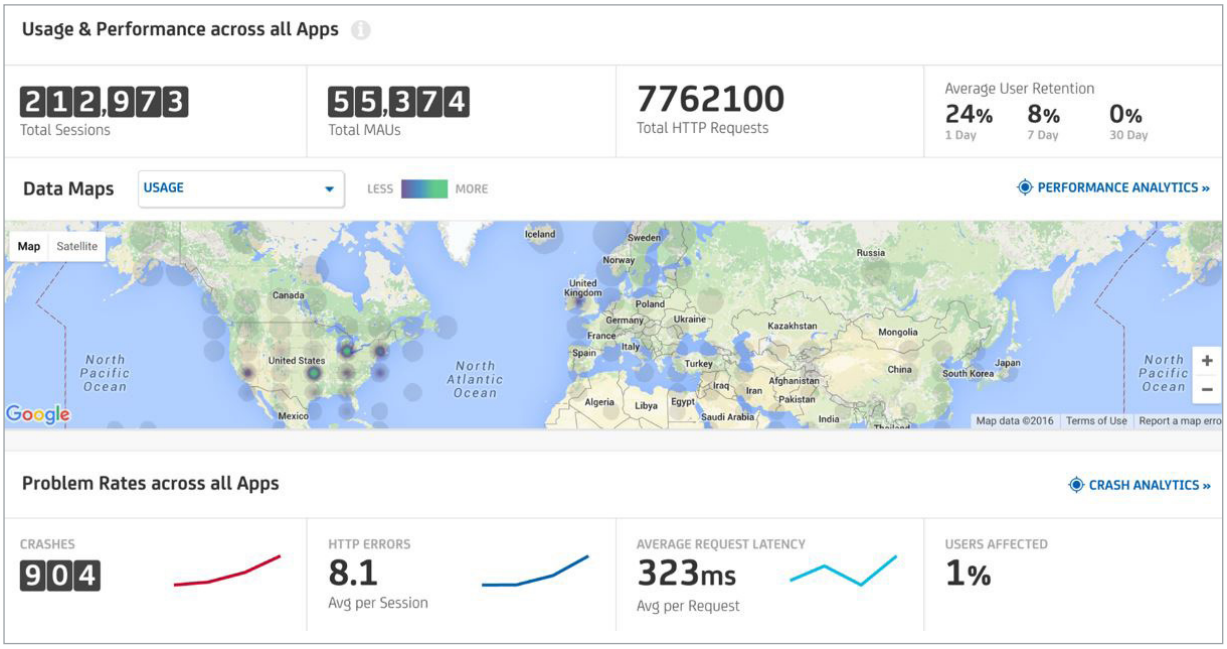
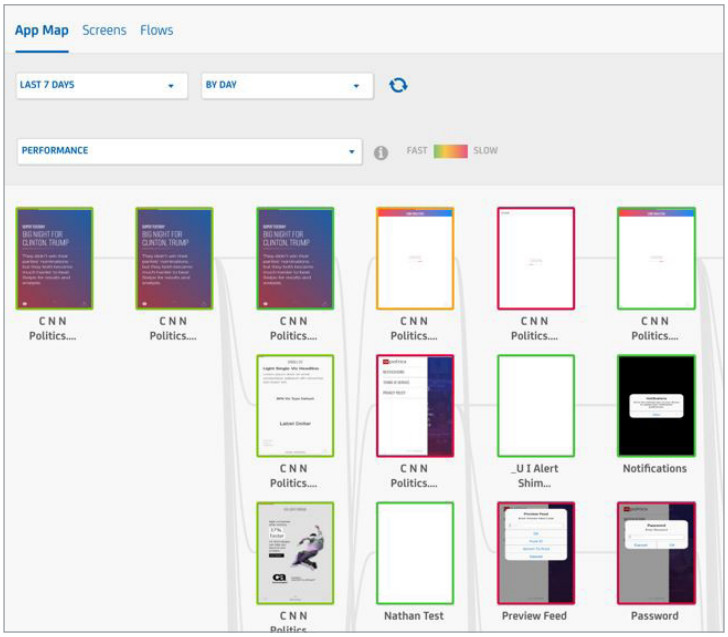


FIGURE C.

App flows provide detailed insight into how customers are navigating through your app and displays app issues and drop-off rates by screen.



"CA App Experience Analytics definitely has a more advanced visualization than anything we've used before, and it's easier to get an understanding of what the data means."

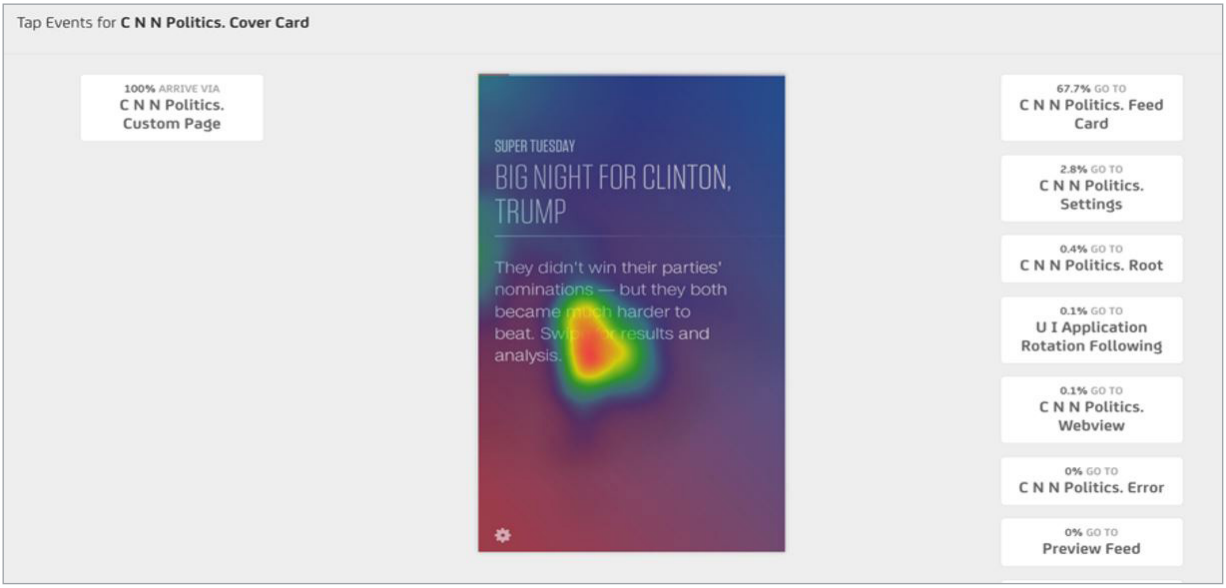
Chris Kilroy,
Director of Client Integration and Activation, CNN

FIGURE D.

App session views allow you to view individual sessions and watch video session playback with animated touch and gesture visualizations so you can understand exactly what your user experienced as they navigated through your mobile app.

FIGURE E.

App screen drill-downs provide touch and gesture visualizations so you can optimize app design based on real-user interactions.



Try CA App Experience Analytics today by [signing up for a free trial](#).

Connect with CA Technologies

CA Technologies (NASDAQ: CA) creates software that fuels transformation for companies and enables them to seize the opportunities of the application economy. Software is at the heart of every business, in every industry. From planning to development to management and security, CA is working with companies worldwide to change the way we live, transact and communicate—across mobile, private and public cloud, distributed and mainframe environments. Learn more at ca.com.

ca technologies

200-341451_0218