



12 October 2012

White paper

# Best Practices in Mobile Application Testing

The basic frame of software testers' mindset is attuned to check documentation, functionality, stability, API and performance and make sure that the software application is secure from all vulnerabilities and check for any other related things surrounding the software piece of code. However, mobile application testing adds a different dimension to the traditional software testing in the sense that the testers have to think about the mobile functions based on usage patterns, heterogeneous devices available in the market with different screen sizes and form factors adds to the complexity of the mobile testing.

This whitepaper talks about the best practices that exist at Calsoft Labs pertaining to mobile application testing focusing on mobile apps for iPhone, Android, Windows Phone and mobile web apps. These practices have evolved over a period of time with our experience in working with global top enterprises and ISVs that improved the testing activities and provided a quick turnaround time reducing the cycle time.

## Functional Testing

Every new feature developed for a mobile device needs to be thoroughly tested. Functional testing is an important aspect when it comes to mobile app testing. Based on the developed test cases, mobile testers should do manual and automated testing. At the beginning of the test phase, a tester must test the mobile app manually using "black box" strategy to see if the functionality provided matches to customer expectations and work as designed. Besides traditional software testing defined in text books, like clicking a button, submitting a form, playing around with different screen elements mobile testers must perform more functional and device-specific testing. Today, the market is flooded with smart phones that have a touchscreen requiring multi-touch gesturing to interact with them and provide motion, tilt and gyroscope sensors.

A mobile software tester must be sure that the app will work with all these specific device functions if they are used within the app. The sheer number of different mobile devices means that it is not possible to cover all of them during testing, so testers need to focus on key areas of their app during functional testing.

Besides the entire manual testing process, it is really important to have good test automation in place for mobile applications. Every code change or new feature could affect existing features and their behavior. Usually there is not enough time for manual regression testing, so testers have to find a tool to perform automated regression testing.

Currently, there are a lot of mobile test automation tools on the market, both commercial and open source, for each different platform like Android, iPhone, Windows Phone 7, BlackBerry, and mobile web apps. Depending on the development strategy and infrastructure, quality assurance experts need to find the test automation tool that best fits their environment.

From an Android perspective, there are open source tools like *Robotium*, *Robolectric*, *Roboguice*, *MonkeyTalk*, *Monkeyrunner*, *NativeDriver* and *Calabash* for Android. The Android test automation tool *Robotium* has become the de facto standard in the open source world, as it is able to simulate real user interaction on real devices.

iPhone test automation tools include *KIF (Keep It Functional)*, *UIAutomation*, *MonkeyTalk*, *Calabash* Frank, Zucchini, and many more for iOS. It is however a standard practice to choose test automation tool that uses the same programming language as the production code. If the test and production code are written in the same language, it provides a number of benefits for both testers and developers, as it makes it easy for them to do collaborate on the app development.

To summarize the best practices of Functional Testing:

- Test the app as a “black box” and try to break it.
- Open every screen of the mobile app and change the position from portrait to landscape mode and back again as most errors are discovered this way.
- Don’t forget to test device-specific functions, like sensors and communication interfaces.
- Write test automation scripts for mobile apps using a standard tool
- Choose a test automation tool that fits into the company strategy and infrastructure.

## Non-Functional Testing

Another important area of mobile testing is the non-functional requirements testing of a mobile app. Usability plays an important role for any mobile application. The ease and comfort with the usage is a huge factor in the success of a mobile app. Hence, the first test to be performed during the early development phase has to be a “user centric” usability test. These should be carried out by alpha users or work colleagues.

Also it’s vital to test the mobile apps behavior and performance during interruptions like incoming phone calls, SMS, MMS, tweets or other notifications.

The key Non-functional requirements to be tested in a mobile app are:

- Usability Tests
- Compare performance levels between the released and the new version of the app(for upgrades)
- Check how the app reacts to incoming calls, SMS, MMS, or tweets
- Validate the battery drain of the test device
- Testing the app’s memory usage
- The ease with which the app is Installed and deleted
- Testing mobile apps for version upgrades
- Compatibility and Localization testing for mobile apps

- Testing the app on different carriers and network connections like GPRS, Wi-Fi or LTE

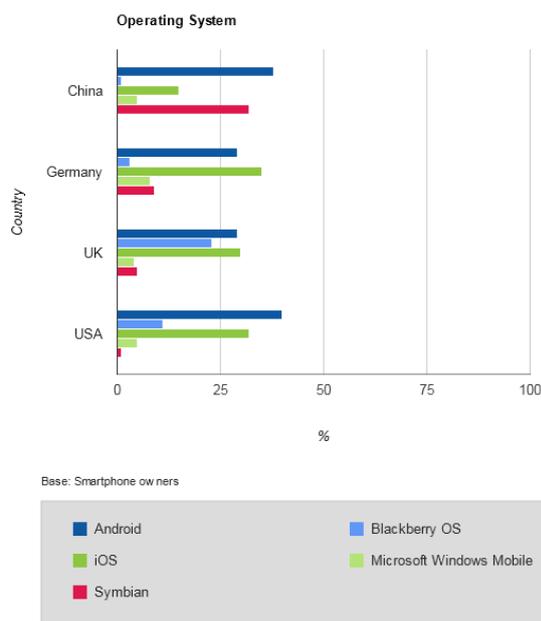
## Choosing the right devices for Testing

Gartner estimates that there will be 1 billion smartphones and 1.2 billion mobile workers by 2013 and large enterprises expect to triple their smartphone user base by 2015. Also, it's a well known imperative that Mobility is driving the consumerization of IT with billions of downloads from Apple app store and Android market place.

With all the seismic and tremendous growth happening in the mobility segment, it becomes a herculean task for an Enterprise to zero in on the right device for testing your application.

An industry research finds that there are 11 OS, 49 manufacturers and 8000 devices. As things stand at the moment, there are two big players on the mobile device market: **Android** and **iOS**. But there are several other platforms that are used fairly often, depending on the geographical region. These are Windows, BlackBerry, webOS, SymbianOS, and feature phones.

The chart below shows mobile OS penetration in different countries.



Source: <http://www.thinkwithgoogle.com/mobileplanet/en>

Nearly every platform has different device vendors who sell smartphones with different hardware and software specifications, as well as customized user interfaces. For example, in the Android world there are vendors like Samsung, HTC, ASUS, LG, Motorola, Sony, Huawei, and many more. This is a serious case of device fragmentation and it is really difficult to find the right devices for testing.

Thinking about mobile websites is another challenge that can be really painful, due to the various mobile browsers such as Safari, Opera Mini, Dolphin, Android and RIM native, Google Chrome, Firefox, Internet Explorer 9, and some other feature phone browsers.

So, do the following solutions really work out?

- Use the latest devices and browser versions
- Buy every device on the market
- Use simulators or emulators for all the devices

A better approach to solving the heterogeneous device problem is to group the devices and browsers depending on their hardware and software specifications with a weightage given to each groups. Each group will have devices assigned based on the platform and vendor. Some of the parameters for grouping the devices are:

- **Smaller screen mobile devices** with a slower CPU, small RAM and lower screen resolution
- **Mid-range devices** with average/optimal CPU, RAM and processor speed with good screen size and better resolution
- **High End Devices with** dual/quad core CPU, higher RAM and greater screen resolution with all the latest software installations

Note: It is always advisable not to use simulators and emulators as the test results on them drastically differ from the testing on real devices.

## Combine Tools

As mentioned before, testers must choose an optimal automation strategy and plan for mobile apps testing in order to make sure that any code changes do not affect core functionality.

Another best practice is to combine best testing tools and integrate them into a continuous integration server in order to execute them from a central place. Developers need to write unit tests for their code to be sure that each small component of the app is safe and works as expected.

In addition, it is very useful to use tools like *Robotium* or *Keep It Functional* to implement end-to-end integration tests.

## Know Your End Customers

Mobile testing is more effective, if the developer and testing team have a better understanding of the target end users of the apps that use them. One of the means by which this information could be mined is via the app store of the specific vendor (Apple App Store, Google Play Store, Windows Marketplace, and so on).

If the team already has multiple apps in one of the stores, they will receive information about customers' preferences about devices, software versions, languages and carriers used.

When it comes to mobile web pages, there is no app store to provide such user data. Hence, it is advisable to get the information about the 'user\_agents' (devices, browsers) used within the mobile website.

With the aforesaid knowledge, the development and testing teams can optimize and decrease the amount of development and testing effort required for the various devices and software versions.

## **ABOUT Calsoft Labs**

Calsoft Labs provides specialized concept to market Product Engineering and embedded design and engineering services both established and early stage product and technology companies in select market segments – ISVs, New Media Companies, Networking and Datacom OEMs, computer hardware manufacturers, semiconductor companies and consumer electronics companies.

### **USA**

2953, Bunker Hill Lane, Suite 203, Santa Clara, CA 95054.

Ph:+1 925 249 3000 - Fax : +1 925 249 3031

### **INDIA**

Robert V Chandran Tower, B3, # 149, Pallikaranai,  
Velachery Tambaram Main Road, Chennai – 600 100

Ph: +91 44 4282 9000 - Fax: +91 44 4282 9012

No. 196, Bannerghatta Road, Opposite HSBC,  
Arekere Circle, Bangalore 560 076

Ph.: +91 80 4034 3000 Fax: +91 80 4034 3111

### **FRANCE**

40 avenue André Morizet, 92514 Boulogne-Billancourt, France

Phone : +33 (0)1 46 08 70 00 - Fax : +33 (0)1 46 08 70 10

*© 2012 Calsoft Labs. All Rights Reserved.*