White Paper

The Mobile Testing Checklist

A guide to help investigate mobile testing solutions, and trigger structured discussions and decisions

opentext™

Market Dynamics

No matter the industry, mobile consumers judge their mobile experience on the speed of response and the quality of service they receive. The complexity is hidden behind user-friendly application interfaces, leading to a feeling that everything is simple and should work flawlessly. Every product and project manager must rethink the way they approach the mobile app lifecycle—inclusive of products, processes, and business models—and focus on the user experience. Mobile app developers and testers understand that selecting the right solution for—beyond mobile app testing—user experience testing can be a daunting task. One aspect we forget is that hours of work go into dust.

Mobile testing is inherently more complex than traditional software testing:

- More complex because mobile apps need to survive ongoing and continual changes of the run-time environment, new devices, new operating system releases, etc. Any and all of these can expose multiple problems with existing apps, which then need to be updated to maintain the status quo. Active apps demand constant care even if the functionality does not change.
- More complex because of our users' freedom of choice, their context, and expectations.

The ground reality is that mobile app testing tool evaluation should not be merely reduced to scripting or functional testing. A balanced approach to assess tools with a user experience mindset is an imperative. Selecting such a solution is a three-phase activity comprising user experience analysis, weighing the tool's features vis-a-vis requirements, and proof of concept (POC) development. Here is a recommended practice:

Step 1: Focus on the User Experience

The first step is to put together a way of defining the various aspects of the user experience, however subjective it might be. Mobile teams have to define a set of user experience metrics that reveals something about the interaction with the app.

Table 1 gathers the various aspects of the user experience you have to take into account in your investigation.

Attribute	Description	How It Applies to Mobile Apps
Functional suitability	A set of attributes that satisfy stated or implied needs	Mismatches between offering and expectation can cause frustration, confusion, wasted development effort, and ultimately lose users
Performance efficiency	Capability to maintain performance under stated conditions for a stated period of time	Speed (and latency) matters tremendously and face impediments across all three layers: device (CPU, memory, battery, and storage), network conditions, and back-end performance. Amazon calculated that a one-second delay in the loading time of its website could result in $1.6 billion in lost sales annually.

Continued on the next page

Attribute	Description	How It Applies to Mobile Apps
Availability	Back-end services are ready for use	Web servers, networks, databases are key parts of the essential infrastructure to support the mobile moments. Issues damage brands, lower revenue and employee morale. An unprecedented increase in mobile traffic caused BestBuy to crash on Black Friday (2014). Also, they may need to be available when third-party login authentication mechanisms are involved, such as a Facebook logon service or a payment service such as PayPal.
Security	Coverage and visibility of security control between layers and interacting components	Insecure apps are likely to be discovered, and when they are, their provider loses the trust of the users. The Ashley Madison breach in 2015 is one of the more widely publicized examples of data collection gone awry.
Usability	A set of attributes that bear on the effort needed for use	Mobile apps cannot afford steep learning curves; after all, many users will only give you one shot before abandoning an app. Even more persistent users are unlikely to appreciate apps that have awkward features (as compared with challenging puzzles or games).
Accessibility	Users have preferences or disabilities that affect how they interact with technology	Our apps also need to improve so that they work effectively with the various accessibility features provided by the platforms, e.g., to change the color scheme, the font size, or to have the device read out information on the screen.
Portability	A set of attributes that bear on the ability of software to be transferred across environment	Porting from one platform to another can be a major undertaking, particularly for native apps. Portability from one device model to another varies in complexity; for instance, non-trivial Android apps may have custom implementations to work well on Samsung devices.
Maintainability	Robustness facilitating the time to adapt	Apps need ongoing changes to keep the app viable, particularly because their environment continues to change as new operating system releases and new devices become available. In addition, changes include modifications to fix problems and add new features.

Table 1: User Experience Attributes

Step 2: Scorecard the Top 10 Capabilities

Organizations typically approach vendors with a list of 10 motivators, which vary from one company to another, one project to another.

The purpose of this document is to help you identify weaknesses and strengths for each vendor, inclusive of OpenText™.

Requirements	Definition
Lab management	Breadth of lab management functionalities, e.g., devices, app, users
Mobile technology	Ability to support the mobile technology, e.g., real devices, OS, gestures, events
Manual testing	Ability to accelerate manual testing, e.g., in-the-lab, crowd testing, path to automation
Functional testing	Ability to automate functional testing, e.g., record, run, replay, report
Performance testing	Ability to assess the end-to-end performance prior to release, e.g., test across all tiers
Security testing	Ability to provide an end-to-end security analysis of the app, e.g., storage, session management
Mobile analytics	Ability to capture real-time information, e.g., app and user behavior, app store feedbacks
Mobile monitoring	Ability to monitor the performance and availability post-release
Enterprise readiness	Ability to comply with enterprise requirements, e.g., market coverage, references, support, training
Licensing model	Ability to provide a flexible and relevant licensing model, e.g., term options, variety of price points

Lab Management

Breadth of Lab Management Functionalities	OpenText	Other Vendor	Comments
Deployment			
Does the vendor support hybrid deployment, e.g., on-premises installation with cloud services?"	Yes		
Does the vendor support a distributed private cloud?	Yes		
Does the vendor provide an enterprise-grade public cloud, e.g., OpenText™ LoadRunner Cloud?	Yes		
Device Management			
Real devices	Yes		
Emulators, e.g., Genymotion, Android SD	Yes		
Remote access (incl. reboot)	Yes		
RDP performance	Fast		
Immediate swap	Yes		
Disable Wi-Fi	Yes		
Disable carrier data connection	Yes		
Automated provisioning/enrollment	Yes		
App Management			
Support for Web, hybrid, and native apps	Yes		
Automatic distribution of apps on targeted devices	Yes		
Central view of all app versions	Yes		
Insight into the app behavior in production, e.g., performance, stability	Yes		

Mobile Technology

Ability to Support the Mobile Technology	OpenText	Other Vendor	Comments
iOS			
Supported versions	Yes		
Ability to support new versions	Yes		
Ability to fully interact with apps without instrumentation	Yes		
Ability to include system apps within a test case	Yes		
Support for gestures, e.g., tap, double tap, swipe, pinch, touch-and-hold, press, rotate, scroll, home screen jump	Yes		
GPS injection	Yes		
Events, e.g., call/SMS simulation	Yes		
Real cellular reception (beyond in-the-lab Wi-Fi)	Yes		
Capture and simulate network conditions	Yes		
Android			
Supported versions	Yes		
Ability to support new versions	Yes		
Ability to fully interact with apps without instrumentation	Yes		
Ability to include system apps within a test case	Yes		
Support for gestures, e.g., tap, double tap, swipe, pinch, touch-and-hold, press, rotate, scroll, home screen jump	Yes		
GPS injection	Yes		
Events, e.g., call/SMS simulation	Yes		
Real cellular reception (beyond in-the-lab Wi-Fi)	Yes		
Capture and simulate network conditions	Yes		
MADP Support			
IBM Worklight/Workbench	Yes		
SAP	Yes		
Sencha	Yes		
Microsoft Xamarin	Yes		

Manual Testing

Ability to Accelerate Manual Testing	OpenText	Other Vendor	Comments
In-the-Lab			
Screenshot grabbing	Yes		
Auto-authoring	Yes		
Screen annotation	Yes		
Full report, e.g., snapshots, video recording, device vitals	Yes		
Manual to automation script	Yes		
In-the-Wild			
OTA build distribution	No		
In-app bug reporting	No		
Team management	No		

Functional Testing

Ability to Automate Functional Testing	OpenText	Other Vendor	Comments
Record			
Cross-OS and device scripting	Yes		
Full object identification, e.g., system-level objects, spy, breadth of properties, repository	Yes		
Visual analysis, e.g., OCR, image matching x-path/regex	Yes		
Descriptive programming	Yes		
Checkpoints	Yes		
Settings control	Yes		
Conditional flow	Yes		
Replay			
Execution management and status			
View running scripts			
Automatic restart on crash			
Integration with Existing Testing Tools			
OpenText™ ALM/QC/UFT One/UFT Developer	Yes		
Selenium	Yes		
Appium	Yes		
Calabash	Yes		
Continuous Integration			
Jenkins			
Bamboo			

Performance Testing

Ability to Automate Performance Testing	OpenText	Other Vendor	Comments
Generic Requirements			
Include all three tiers in a test: real devices + virtual users, network conditions, back end	Yes		
Cross-OS and cross device record and replay	Yes		
Single script for functional and performance	Yes		
Device vitals under load	Yes		
Integration			
OpenText™ LoadRunner Professional	Yes		
OpenText™ LoadRunner Enterprise	Yes		
OpenText™ LoadRunner Cloud	Yes		

Security Testing

Ability to Provide an End-to-End Security Analysis of the App	OpenText	Other Vendor	Comments
Assess the Mobile Client			
General requirements: Filesystem, memory, run-time tampering, input validation	Yes		
Source code and binary analysis	Yes		
Inter-app communication	Yes		
Evaluate the Mobile Network Traffic			
Transport layer security	Yes		
Data stream and malware analysis	Yes		
Host communication enumeration	Yes		
Analyze the Web Server			
Mobile Web app vulnerability assessment	Yes		
Mobile SOAP1 or REST2-based Web service testing	Yes		
Static analysis of any backend code	Yes		

Development and DevTest

Ability to Support Development Activities	OpenText	Other Vendor	Comments
Virtual routing of remote devices to workstation	Yes		
Plugin for IDE, e.g., IntelliJ/Eclipse/Visual Studio	Yes		
Remote debugging capabilities	Yes		

Synthetic Monitoring

Ability to Monitor the Performance and Availability Post Release	OpenText	Other Vendor	Comments
Generalities			
Reuse of performance script	Yes		
Real-user monitoring (Web apps)	Yes		
Synthetic transactions on real device	Yes		
Single and multi-step monitoring	Yes		
SLA tracking and reporting	Yes		
Back-end diagnostic	Yes		

Enterprise Readiness

Ability to Comply with Enterprise Requirements	OpenText	Other Vendor	Comments
Market Presence			
Installed base			
Geographical account			
Financial strength			
Support			
24x7 support	Yes		
Online help	Yes		
Self-ticket management	Yes		
Phone support	Yes		
Implementation and Training			
Online training	Yes		
On-site training	Yes		
Partner and MSP ecosystem	Yes		

Licensing Model

Ability to Provide a Flexible and Relevant Licensing Model	OpenText	Other Vendor	Comments
Generalities			
Open source	No		
Freemium	No		
Trial	No		
Pay-per-use	Yes		
Variety of price points	Yes		
Subscription licensing	Yes		
Perpetual licensing	Yes		
Flexibility	Yes		
Scalability	Yes		
Ability to change/add new devices	Yes		
Simple product activation	Yes		

Step 3: Bulletproof Your Primary Selection

Once you shortlisted one or a couple of tools, you could develop a POC using the trial version and verify the vendor's claims. Make sure you provision enough human resource and time to support the POC. Here is the recommended approach.

1. Preliminary activities
   - **Define the purpose, goals, and objectives**
   - **Establish success criteria**
   - **Define the scope, duration, and risks**
   - **Create work plan and deliverables**
2. Conducting the POC
   - **Setup infrastructure: simplest deployment that can scale to production needs**
   - **Connect mobile devices: use appropriative devices and assets ("same as" production)**
   - **Evaluate testing scenarios (record/replay, remote execution, continuous scenarios, etc.—per scope)**
   - **Raise questions as you go and collect responses/feedbacks**
3. Evaluating the POC
   - **Outline discovered issues/challenges**
   - **Evaluate in accordance to the success criteria**
   - **Drive conclusions**

Learn more at
UFT Digital Lab
www.microfocus.com/opentext

Connect with Us

OpenText CEO Mark Barrenechea's blog

opentext™