**MOBILE APPLICATION TESTING**

Mobile Application Testing is a process by which a developed application of a mobile device is tested for its functionality, consistence & usability. It can be automated and manual. It almost goes through the normal testing process but the testing techniques may vary for mobile devices.

Mobile Testing needs to be performed on variety of software platform, versions, hardware, network condition and what makes it more complex are the various devices, fast-changing mobile OS, the frequent introduction on new devices, and unpredictability of simulators.

* ***Different range of mobile devices*** with different screen sizes and hardware configurations like hard keypad, virtual keypad (touch screen) and trackball etc.
* ***Different mobile operating systems***like Android, Symbian, Windows, Blackberry and IOS.
* ***Different versions of operation system*** like iOS 5.x, iOS 6.x, BB5.x, BB6.x etc.
* ***Different mobile network operators*** like GSM and CDMA.
* ***Frequent updates*** – (like android- 4.2, 4.3, 4.4, iOS-5.x, 6.x) – with each update a new testing cycle is recommended to make sure no application functionality is impacted.

Types of Mobile Testing

When we talk about Mobile Device Testing, it can be categorized broadly in two segments;

1. ***Hardware testing/Mobile testing:*** This is to assure the quality of mobile devices. The device including the internal processors, internal hardware, screen sizes, resolution, space or memory, camera, radio, Bluetooth, WIFI, etc. This is sometimes referred to as, simple “Mobile Testing”.
2. ***Software testing/Application testing:*** The applications that work on mobile devices and their functionality are tested. It is called the “Mobile Application Testing” to differentiate it from the earlier method. Even in the mobile applications, there are few basic differences that are important to understand:

**a) *Native apps***

**b) *Mobile web apps***

**c) *Hybrid apps***

**Checklists for mobile application testing**

**Cross-Platform Compatibility**

Your android application may run fine on Oreo (Android 10) but the same might not be the case with Marshmallow(Android 6.0). This is where cross-platform testing comes into the picture.

You should ensure to test your application on all the possible combinations of operating systems, screen sizes and resolutions that the app users might be using. This ensures that the app will behave uniformly across all devices.

**App Integration**

A mobile app interacts with many features – both in-app as well as mobile device based. The app should be compatible with all the possible settings and configurations. These may include screen resolution, orientation, different versions of operating systems, camera, motion sensor, etc.

**Application Type**

A mobile app generally falls under three categories – native, web, and hybrid.

i. Web app testing : Optimized for mobile browsing and server-driven, the web apps should be tested for UI to ensure that all the elements of the app are aligned for different screens and resolutions. Since the mobile apps have limited storage space and power supply, test for memory utilization as well as battery usage. Avoid excessive usage of JavaScript  – this might drain the mobile’s battery.

ii. Native and Hybrid app testing : Although driven by different technologies, native and hybrid apps share similar functionality, and hence, their approach to testing. Functional testing is the key player here  – run functional tests to validate key aspects such as screen orientation, gestures, compatibility, connectivity, performance, and interruptions such as calls and push notifications.

**Front-End Testing**

Front-end testing validates the features available on the UI of an app – testers check the functionality of the mobile app’s GUI. This includes testing the menus, dropdowns, navigation buttons and gestures, forms, and other features that are used by the end-user.

Front-end testing will ensure that the app under test will perform on different browsers and devices. Perform unit, regression, performance, integration, and acceptance tests on the app for comprehensive front-end testing.

**Back-End Testing**

Back-end or database testing deals with the server-side of the mobile app. Today, most of the mobile apps are API-driven. Data is populated via REST APIs displayed in the UI. Use a proxy to track the app’s requests and responses. Perform SQL, performance, and security testing to prevent deadlock, data loss, etc.

**Storage**

Mobile users want to have all their important apps on the go. However, the devices do not have enough storage to fulfil the need of installing high-resolution apps.

Keep an eye on the app size with every release. You should not be in a situation where the end-user might not download the app only because of its large size.

**Network Compatibility**

Run the app on different network conditions to measure its performance under weak signal and data transition.

**Functional Testing**

This mobile testing type aims to verify that every function of an app is working exactly as required. It focuses on testing user flows within the app, as well as each feature that comprises each user flow.

It includes :

App installs and launches correctly

Users can sign-up and log into the app easily

Text boxes, buttons, menus and icons function accurately

Push notifications render correctly and appear at the right intervals

Any transactions or purchases should happen seamlessly

**Interruption Testing**

This form of mobile testing checks how an application responds when faced with an unexpected interruption.

some common interruptions that should be considered while testing are:

Incoming or phone call when an application is running

Incoming message or SMS when an application is running

Low battery when an application is running

The device plugged in or out of charging when an application is running

Device shutting down when an application is running

OS upgrade occurring when an application is running

Loss and restoration of the network while an application is run.

**Operational Testing**

Any mobile OS and desktop OS provides in-built back-up and recovery operational functions that save or recover all files or doc of mobile devices or applications that had been lost due to some reason.

Operational testing is used to test that the particular back-up and recovery process is working properly and responding as per the requirement.

**Memory leak Testing**

Test to ensure that each application of the mobile device is using optimized memory for processing.

Due to memory leakage the processes might slow down while transferring the file or in-between accessing any application mobile device or application might force stop automatically.

**Usability testing**

Usability testing is used to test the mobile applications in terms of usability, flexibility, and friendliness.

The testing process makes sure that the mobile app is now easy to use and offers a suitable user experience to the customers.

**Performance Testing**

It test how an application performs under various cond itions.

A few things that performance testing should verify:

Device performance: Installation and log-in time, battery consumption, memory consumption, etc.

Network performance: Delays, errors, pauses in receiving digital information or rendering network-activated features

API/Server performance: Speed and formation of data transfer from back-end to front-end

**MOBILE APPLICATION CHALLENGES**

The Goal of testing any application is to make sure that the application is best in Quality and the performance of the application is optimum.

QA team has to make sure that the ample functionality of the Mobile app is tested in all the verticals.

The key Mobile Testing Challenges that are faced in testing the mobile app can be classified as below:

* ***Mobile Devices and OS combinations***
* ***App Categories***
* ***User Interface***
* ***Technology***
* ***Security***
* ***Automation***

Real device or Emulator

After making a clear mindset with the list of devices and operating systems you want to use for your testing, now you need to choose if you want to use a physical mobile device or emulator for your testing purpose. Both physical device and emulators have their own set of advantages and disadvantages.

A real device offers more reliable real-time result as the testing is carried out on the same device as the intended user. It also allows the tester to test behaviour during an incoming call or message. But it also has its own set of drawbacks, real devices come with a cost and need to be maintained. Sometimes it is also difficult to get your hands on some device.

On the other hand, emulators can be obtained for free (open source) and can also be connected with your IDE for testing purposes. But it also has its own set of problems, it is slower than a real device, it doesn’t show behaviour such as network connectivity, battery, overheating, app interaction with other apps.

So, as we have seen both the real device and emulators have their own set of advantages and disadvantages. It is advisable to use them together. A real device can be used to test specific conditions that cannot be tested by emulators and all other test cases can be executed on the emulator.

***Advantages of physical/real device:***

*1. It offers better and clear results*

*2. Network connectivity and its dependency on the application can be tested effectively*

*3. Can offer real-time result*

*4. Can allow the tester to test the interaction of different components together and its impact on the application.*

*5. The tester can test application performance during a multi-tasking while making a call or during roaming.*

***Disadvantages of physical/real device:***

*1. Physical device comes with a price tag*

*2. It is difficult to procure and maintain a huge number of the mobile device for testing.*

***Advantages of Emulators/Simulators:***

*1. Most of the emulators are available for free of cost. This is the biggest advantage when compared to a real physical device which always comes with a price tag.*

*2. It helps developers in performing step by step debugging of the application.*

*3. It can help in testing mobile web application*

*4. It allows testers to capture screenshots of the failures.*

***Disadvantages:***

*1. Emulators and simulators are slower when compared to the real device. It observes lesser response rate than the real devices connected to any network.*

*2. It has certain limitations while mimicking user interactions. These limitations may lead to false result output. So, emulators can be less reliable than a real device.*

**Test scenarios for Mobile application**

* Ensure the app has been launched by downloading and installing it for use.
* Verify that the mobile app display is adaptable to the device screen and also ensures all menus on the app are functioning.
* Verify that the text on the mobile app is readable and clear.
* Check that the app display is adaptable and amenable to the various display mode (i.e. landscape and portrait).
* Verify that the app does not stop the functioning of other apps on the mobile device.
* Verify that in the play screen, the back key allows the app to go back to the start-up screen.
* Check that the app still operates as intended, if the device resumes from inactive mode or the lock screen.
* Check whether the app reminds the user to save setting changes or changing of information before moving to other activities on the app.
* Verify that the on-screen keyboard appears immediately the user attempt to enter a text.
* Check if the app behaves as designed if the mobile device is shaken.
* Verify that the app still functions as designed when “battery low” notifications appear on the screen.
* Check that the app goes into the background when on call.
* Check that the app still operates as designed when a message or notification pop-up from another app such as Facebook messaged, Instagram, etc.
* If the app comes with a users’ settings features, check if the app changes when some form of change is affected by the user.
* Check the [**Performance**](https://www.testrigtechnologies.com/service/performance-testing/)of the app on the different internet networks such as 1G, 2G, 3G, or 4 G networks.
* Check that the app operates as intended when the device is connected to the internet through WiFi.
* Check that the app still operates normally when there is an incoming call or SMS.
* Check that the app is adaptable to different mobile platforms or OS such as Android, iOS, Microsoft, etc.
* Check that the font size and style of the app are compatible and readable to the users.
* Verify that that the loading time for the app is not too long.
* Check that the app is still working as intended after the successful update of the app.
* Check how the app function under different battery levels and temperatures.
* Verify that the app is not draining too much battery.
* Check that the app support image capturing.
* Check that the app does not log out the user before the end of a session