

# Introduction to Mobile Application Testing

Nowadays, mobile plays a vital role in an individual's life. It is analyzed that maximum people try to access websites and applications from their phones rather than opening/switching a computer system or laptops for every small task. Hence it becomes very important for a Quality Assurance engineer to maintain the quality of the applications on mobile phones by properly testing them on all possible devices. Hence mobile testing comes into the picture.

## What is Mobile Application Testing?

Mobile Application testing is defined as a process in which various applications and websites are tested on various mobile devices to enhance the quality of the developed application.

Many organizations who have moved into modern DevOps practices are also investing in converting or building new projects into Microservice-based architecture.

## What are the types of Mobile Applications?

There are various types of mobile applications that various organizations develop. These mobile applications play a very important role in deciding your mobile testing approach. So here are the following types of mobile applications:



1. **Native:** This category involves applications that are compatible with Android and iOS. The iOS applications can be tested using the Appium framework built in an IPA binary file. In contrast, the android applications can be tested using [Espresso](#) frameworks as they are built in an APK package.
2. **Web Application:** Web apps are those apps that are usually accessible with the help of browsers such as chrome, firefox, safari, etc. Such applications are tested using the [Selenium framework](#).
3. **Progressive Web Applications:** PWA is those applications that can be installed using links without visiting a play store. Example: Instagram, Twitter.
4. **Hybrid Web Application:** These applications are usually a combination of the native and the web application. They are written in HTML and CSS.

## What are the types of Mobile Application Testing?

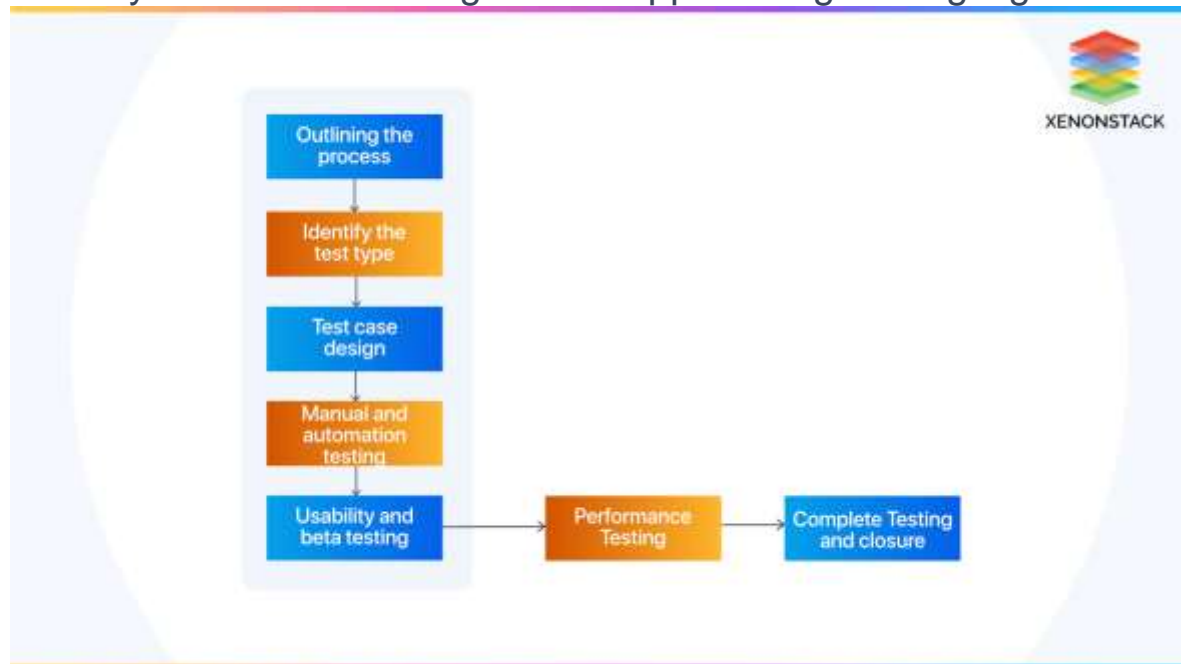
- **Functional Testing:** This type of testing verifies whether our application's functionality is working as per the requirements in various mobile devices.
- **Compatibility Testing:** This type of testing ensures that our application is working perfectly on different mobile devices, Operating systems, browsers, networks, etc. The main aim of this type of testing is to make sure that our application is compatible with all the combinations of different operating systems and browsers.
- **Usability Testing:** The main aim of this testing is to evaluate the degree of user-friendliness the application provides to its users. It helps in detecting and fixing user experience issues.
- **Performance Testing:** This type of testing is used to evaluate the performance of mobile applications under varying load and stress levels.
- **Security Testing:** This type of testing evaluates how safe and secure our application is from various security flaws and loopholes.

- **Interrupt Testing:** As it is clear from its name, this type of testing is used to evaluate the application's behavior when encountered by various interruptions like incoming and outgoing phone calls, low and full battery, message notifications, alarms, etc. On average, a user faces multiple interruptions while operating a mobile application. Hence this type of testing helps to enhance the quality of the mobile applications.
- **Installation Testing:** This type of testing evaluates whether our application is getting installed and uninstalled on our mobile devices without any major flaw or issue.

A phase in which a set of activities are performed and ensures the ultimate quality during the software engineering process that provides self-assurance and sureness to the user as well.

## What are the key factors influencing Mobile App Testing?

The key factors influencing Mobile App Testing are highlighted below:



1. **Mobile Devices:** Mobile application testing follows a device testing approach. It is necessary to test our application in

various resolutions, browsers, mobile devices, and operating systems.

2. **Mobile Simulators:** Mobile simulators play an essential role in mobile testing. A simulator is a software that simulates the behavior of an actual device, or we can say it mimics the behavior of an actual device. These simulators provide a cost-effective way to test on various mobile devices without buying actual devices.
3. **Network Conditions:** It is essential to test our mobile applications under various network conditions to make sure that our application behaves perfectly under networks like 2G, 3G, or 4G.

## What are the processes of Mobile Application Testing?

With a highly competitive app marketplace, organizations need to match and exceed their users' expectations to taste success. It becomes essential to leverage the power of mobile app testing to enhance the overall quality of the application. An effective mobile app testing involves:

1. **Outlining the Process:** The first and foremost step of mobile testing is to outline the entire process. This includes planning various activities, identifying multiple constraints, selecting mobile devices to test on, preparing a compatibility matrix, etc.
2. **Identify the testing type:** The next step is to identify the type of testing that needs to be performed based on the type of application under test.
3. **Test cases design:** The next step involves the preparation of a proper test case sheet covering all the features and functionalities that need to be tested. In this step, the tester needs to separate the manual test cases and the automation scripts.
4. **Manual and Automation Testing:** In this step, both the manual and automated test cases and test scripts are

executed to identify any major issue or loophole in our application.

5. **Usability and Beta Testing:** Now, once the application's basic functionality is tested successfully, the task is to ensure that our application is user-friendly and provides the best user experience. First, the application is tested by the in-house employees, and then it is released in a beta version for the beta testers to detect any issues or flaws.
6. **Performance Testing:** The next step is to determine the application's performance to identify scalability issues.
7. **Complete Testing and Closure:** This is the last phase of mobile testing that involves a set of final testing to make sure that our application is up to the mark and stable with no major bug or issue. In this, it is the responsibility of the QA team to perform final regression testing and check the major functionalities before giving the Signoff report and verify that the acceptance criteria are met or not.

A process that allows the user to check that the provided data, they deal with is valid or complete.

After the completion of testing, a final report is generated that contains the following points,

1. A number of bugs were generated.
2. A number of bugs were resolved.
3. Number of bugs pending
4. Different types of testing were performed.
5. Compatibility matrix of all the devices, operating systems, and browsers.

## What are the best tools for Mobile Application Testing?

The best tools for Mobile Application Testing are listed below:

### Appium

1. Appium is an open-source mobile testing tool.

2. It is used for automating both Android as well as iOS applications.
3. It supports the following programming languages: Java, Ruby, Python, C#.
4. It is examined as one of the best tools available in the market for mobile testing automation.
5. It can integrate with Continuous Integration and Continuous Development as well as with various other tools.

### **Robotium**

1. It is an open-source mobile testing tool.
2. It is developed for automating android applications only
3. The programming language that it supports is Java.
4. Robotium supports integration with Maven and Ant.

### **Selendroid**

1. Selendroid is an open-source tool used to test web applications.
2. It is also known as Selenium for Android.
3. It works on a native, hybrid, and web-based application
4. It uses Selenium as its scripting language
5. It provides support to web driver compatible languages such as Java, C#, Perl
6. It also provides us with the feature of object recognition.