# Chapter 5: Testing

# 5.1 Test Plan

**UnitTesting**  
The objective of Unit testing is to verify that a particular module of source code is working properly. Unit tests are designed to test a single class or component or module in isolation. Developers run unit tests in their own environment and only for the components they are working on.

**IntegrationTesting**  
The objective of Integration testing is to ensure that aggregates of units perform accurately together. An independent test team can perform the integration testing. 100% integration testing is not possible on real devices so we can use emulator or dev environment for same.  
Here we can perform Functional testing, Interface testing, Usability testing, performance and security testing.

**SystemTesting**  
The objective of System Testing is to identify defects that will only surface when a complete system is assembled. Here we check the application behavior with different kind of environment and hardware combination.We can perform system testing on real devices or the third party tools like perfecto Mobile or Device anywhere. It is intended to validate the application  
Here we can perform functionality testing, usability testing, security testing, installation testing etc.

**RegressionTesting**  
The objective of regression testing is to check that old functionality has not been broken by new functionality or changes made in application. Like system testing we can perform regression testing on real devices and third party tools like perfecto Mobile or Device anywhere. Here we can perform all testing types Installation, UI, Usability, Performance, Security, interruption etc. We can also automate the regression testing in mobile domain. Given below are the lists of tools which can help automate our testing.

Robotium – Robotium is a test framework created to make it easy to write powerful and robust automatic black-box test cases for Android applications. Robotium has full support for Activities, Dialogs, Menus and Context Menus.

Perfecto Mobile - MobileCloud-Automation allows users to execute automated testing on real handsets and tablets located anywhere in the world. Users can automate their testing using the intuitive web-based user interface, which allows the creation of keyword-based ScriptOnce™ scripts. Once the scripts are done, users anywhere in the world can access the devices and run the scripts.

**AcceptanceTesting**  
Acceptance testing refers to the execution of a set of test cases on the target device itself. It is intended to verify that a particular product or application meets the requirement specifications of the mobile application under test.

**UITesting**  
All mobile platforms have certain submission guidelines to follow before the application can be available commercially. Apple AppStore is very strict on guidelines. Most of the applications submitted are rejected due to small errors on AppStore. We need to verify (UI test) that application is meeting guidelines.

**UsabilityTesting**  
Usability testing is a technique of software testing, through which we validate the simplicity of the software product. Under this technique we check that a new user how easily understand the functionality and behavior of the application and in any complication, how application communicate it to the user.

**FunctionalityTesting**  
Functional testing ensures that the application is working as per the requirements. Most of the test conducted for this is driven by the user interface.

**SecurityTesting**  
Smart phones are fast replacing traditional computers. As the user base is rapidly shifting to mobiles, hackers are also shifting their attention to mobiles. Due to this trend, conducting security tests on these applications has become a necessity.

**PerformanceTesting**  
This testing process is undertaken to check the performance and behavior of the application under certain conditions such as low battery, bad network coverage, low available memory, simultaneous access to application’s server by several users and other conditions.

**5.2 Test Cases**

**1.Login page:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Test case id** | **Test cases** | **Input test data** | **Steps to be executed** | **Expected results** | **Actual results** |
|  | 1 | Test if user is able to login successfully. | correct username,correct password | 1)Enter input(correct )username and password on the respective fields 2)click submit/login | User must successfully login to the web page | (note down the results you have observed) |
|  | 2 | Test if unregistered users is not able to login to the site | incorrect username,incorrect password | 1)Enter input(incorrect )username and password on the respective fields 2)click submit/login | Proper error must be displayed and prompt to enter login again | (note down the results you have observed) |
|  | 3 | Test with valid username and empty password such that login must get failed | valid username and empty password | 1)enter the valid username in the user id and enter no password in the password field | Proper error must be displayed and prompt to enter login again | (note down the results you have observed) |
|  | 4 | Test with empty username and valid password such that login must get failed | empty username and valid password | 1)leave the username empty in the user id and enter a valid user's password in the password field | Proper error must be displayed and prompt to enter login again | (note down the results you have observed) |
|  | 5 | Test with empty username and empty password and check if login fails | - | 1)Enter nothing in the mail id and password field 2)click submit button | Proper error must be displayed and prompt to enter login again |  |
|  | 6 | Check of the password is masked on the screen i.e., password must be in bullets or asterisks | some password(can be a registered/unregistered) | 1) Enter the password field with some characters | The password field should display the characters in asterisks or bullets such that the password is not visible on the screen |  |
|  | 7 | Check if the login function handles case sensitivity | case changed username /password | 1)Enter the case changed username /password in the respective field and 2)click login button | Login must fail saying incorrect username/password |  |
|  | 8 | After logging in try to copy/cut the password and paste it on another screen(passwords are usually in \* such that its not visible on the screen) | Registered user's login id and password | 1)Enter username and password in the respective fields. Copy the password field's content(which is in \*s) 3)paste the content on another screen | password shouldn’t get pasted / password should not be visible on the screen |  |
|  | 9 | Verify account lock | Registered user's login id and incorrect password | 1)Try to login with a registered user name and incorrect password for more than 3 times | Account should be locked and access should be granted only after gettting certain assurance from the user |  |
|  | 10 | Check if on selecting back button (after logging out) if the user is not signed in | Registered username and password | 1)Login with registered username and password 2)once your are logged in, sign out of the site 3)now press back button | User shouldn’t be signed in to his account rather a general webpage must be visible |  |

**2.Registration form:**

Testing Full Name Field

* Check the Full name text field without Prefix.
* Check the Full name text field without adding First name.
* Check the Full name text field without adding Last name.
* Check the Full name text field with special characters in Prefix.
* Check the Full name text field with special characters in First name.
* Check the Full name text field with special characters in Last name.
* Check by adding numbers instead of string in the full name text field.

Testing Email Field

* Check the Email text field that has Email address without @ symbol.
* Check the Email text field that has random string instead of real email.
* Check the Email text field that has @ symbol written in words.
* Check the Email text field that has missing dot in the email address.
* Check the Email text field as “name@gmail”
* Check the Email text field as “@gmail”
* Check the Email text field as “name@gmail..com”
* Check the Email text field as “name@192.168.1.1.0”
* Check the Email text field as “name.. @gmail.com”

Testing Password Field

* Check the password text field validation
* Check the password with confirm password field