**Important set of test cases for Login Page**

Have you appeared for testing interview, or have you gone through the testing of banking or social networking site? If yes then hopefully you would be familiar with login page that is used to protect confidential information,critical system functionality or data with high integrity. Login page not only give you authority to change in to you account but also give freedom see the critical and confidential information behind the login page.

Here is the set of test cases that would help you in getting the right job with a better understanding of a login page.

* **Test with correct username and password:** This is the most basic positive test cases, while using this user should successfully logged in.If correct username and password is not helping you to login in to your application then file a bug because this shows that something is wrong with application.
* **Test with incorrect username or password:** access should be denied
* **Test with correct username and empty password:**In this case when user click on login button then a message should flash that says “Enter a password or something unexpected went wrong”
* **Test with empty username and correct password:** Again an error message should appear to enter a valid email or username
* **Verify the correct error messages like**  Incorrect combination of user name and password. If you are getting anything like Incorrect username or Incorrect password then be conscious because you application is giving half the information to hacker and your application is in great danger.
* **Verify that back button is not able to push you to your logged in page just after you logout from your specific account:** This kind of test cases invoke the flaw associated with Session management. When session is not closed just after your log-out means any one can access you account if you have opened your specific login enabled account at any time just by clicking Back button in Browser.So one way to save your account from such misconducts is to close browser when ever you log-out from your account.
* **Test a page url without login to application**: For such thing login with correct username and correct password and go to certain page, copy the url and paste the same in another browser.if you are able to open the page then this application is not in good shape to protect the user’s information because anyone can open a specific page just by entering direct url.
* **Verify the session timeout:** This is most important test case for any finance related site.Session should time out if user is inactive for few minute. This is normally a sustainability test of session. If your application is not prompting for session timeout then think once this may be issue.
* **Verify https in url for login page:**  S with Http mean secure http. If login is associated with http in url means you information to login in to application is not secure and anyone can access your information just by doing small effort. While HTTPS ensure encryption of information that is being sent to server from client end.
* **Verify ID in url while processing your request :**  keep track on ID associated with your request url and ID associated with request url should be dynamic not static otherwise this may help some hacker to nab your information.
* **Verify deletion of ID while browsing :** Go to the place where cookies are saved and try to delete cookies when you are just browsing your account and try to find out the cookies that have your username and password because as soon as you delete that cookies you should be reached to login page. If you find the same cookies then try to change the numbers in cookies and should verify what is happening, hopefully corrupt cookies that have your real id should redirect you on login page once again even you haven’t deleted the cookies.
* **Try to login when your cookies are disable**
* **Check SQL injection:** most devastating vulnerabilities to  impact a business, as it can lead to exposure of all of the sensitive information stored in an application’s database, including handy information such as usernames, passwords, names, addresses, phone numbers, and credit card details.  
  So I would suggest including this test cases if you are going to test some banking and insurance related application. Most common SQL injection that is used  ***or ‘1’=’1****, if this got executed then be ready for the loss of your important information.* If means hacker can login without any problem to system or application.
* **Verify account lock out:** I would like to include this test case with priority, if user is using 3 or some specific number of time a wrong password then his/her account should be locked out and access should be allowed after certain assurance form filling or by calling customer care. This may help user from hackers hand.
* **Verify simultaneous login to application on different browser:** I think you all would be familiar with this in daily life if you would have used railway ticket booking site.
* **Try some hit and trial username and password :** before deploying application, username and password like Admin:Admin , Guest:Guest, some username :password, author:author  should be use to test but should be denied when application is deployed.

These all test cases not only test the functionality but also test the security of the application. But I am not saying that these test cases are enough to test security of application. Because security testing is the part of testing that does not have any relation with requirement document so more and more effort is needed if we are talking about the security of an application.