**Discussion Question: Security Test Case**

George Everett, the senior developer at the Mesusa Corporation (MeCo), is working on creating security test cases, and needs your help. He only started the process and has already created these three topics for ***password creation/logging in*** suitable for a security test case.

* Password must be at least 8 characters, include capital letter, number, and special character (@#$%^&\*)
* Password cannot be the same as the login name or user's last name
* After three attempts with incorrect password, user is locked out

George would like you to use the template suggested by the 'Write Test Cases' in the Module 7 Reading above, and write two security test cases. One from the list above and one other (not in the list above) that tests another aspect of user behavior. You do not need to actually run the tests and report the results, but you do need to fill out the other information: ID, Description, Steps, Test Data, and Expected Results.

In your responses to other students, provide feedback on their proposed test cases.

***Before you submit your thread, put your name in the subject line.***

Test cases should be a standard practice when handling programming needs within a company. The Mesusa Corporation would greatly benefit by implementing the test case ID numbers 1 and 2 below. The following should be included when developing test cases for MeCo for password creation/logging.

Password must be at least 8 characters, include a capital letter, a number, and a special character (@#$%^&\*).

Test Case ID Number: 1

Test Case Description: Check the response when a user tries to create a password with an invalid character. The test includes different test data to check for at least 8 characters, capital letters, numbers, and special characters.

Test Steps:

1. Enter email address to link to account
2. Create a password for the account
3. Click the sign-in button

Test Data:

1. Email: [testemail@gmail.com](mailto:testemail@gmail.com) Password: Q2uYr3\*
2. Email: [testemail@gmail.com](mailto:testemail@gmail.com) Password: q2uyr3\*fz
3. Email: [testemail@gmail.com](mailto:testemail@gmail.com) Password: QuYr\*fztopl
4. Email: [testemail@gmail.com](mailto:testemail@gmail.com) Password: Q2uYr3fzT

Expected Results

Password creation should not be successful for any of the included test data because they all violate the requirements when creating a password for the account. The test data “a” should fail due to insufficient characters. Test data “b” should fail to create an account because no capital letter is included. Test data “c” should fail because there is no number. Lastly, test data “d” should fail because the password has no special characters.

A test not included in Everett’s list but should be is testing to see if an account already exists based on the email inputted into the signup form.

Test Case ID Number: 2

Test Case Description: Check if an account exists under the entered email.

Test Steps:

1. Enter email address
2. Create a password for the account
3. Click the sign-in button

Test Data:

1. Email: [existingemail@gmail.com](mailto:existingemail@gmail.com) Password: WaHuf12\*1!!

Expected Results:

Account creation would not be successful, and the pop-up should show that an account already exists under that email, prompt the user to sign in instead of sign up, and give the option for password reset.

**Assignment Requirements and Grading:**

1. An initial post of approximately 250 words is due by **Thursday, 11:59 p.m., CT**.
2. For the initial post to be considered substantive, it should be at least 250 words in length and fully cover the topics being presented. Single sentence definitions or responses will not be awarded points.
3. Submit your post by clicking on the **Assignment Link** above, then **Create Thread**. You must create a thread in order to view your peers' posts. Tip: Create your post in a Word document and then copy and paste your work into the thread.
4. A minimum of three (3) responses, **to the original threads of other students**, of 100-200 words each are due by **Sunday, 11:59 p.m., CT**.
5. To view the rubric grading criteria, click on the following link: [Discussion Board Grading Rubric](https://content.bellevue.edu/cst/csd/rubricdbv3.pdf)

**(50 points)**

Hey, Nima. I really enjoyed reading through your discussion post this week. I like how you added an additional test case. You really went the extra mile for MeCo. The screenshots that you included of the test cases were very clean and a smart way to compile the information. I also created a test case for the password, and I found that using different test data was the best approach to ensuring all aspects of a password requirement are included. Testing for user inactivity is vital in security protections, which is a great test to include. The previous password checker is also a good test to include.

Hi, Jessica. My only suggestion for your test ID 1 is to include a password test that does not have 8 characters, and also run tests where only one requirement is missing at a time. Otherwise, when the test data runs, it does not guarantee that each requirement has been checked. This means that all the requirements might not be met, so separating helps ensure that the code performs properly. I think your second test is a great inclusion for MeCo. Someone can expose sensitive information if they hop onto an active session on a public computer that was not closed.

Hello, Arely! I think you did a great job on your test cases for this discussion post. A password is not very secure if it contains the same information as the user name, so implementing code so it does not do this is important. For your first test case, I recommend including additional test data to make sure the program is working properly. There should be test code with Smith and smith and Asmith and Asmith and aSmith, so that no matter the case, it will still register that the first name or username is being used in the password.