**LABORATORY PRACTICAL ASSESSMENT**

**Functional Testing - Software Testing**

**Course: TESTQUA 31**

**Type: Laboratory Practical Assessment**

**Student Name: Miguel Antoin Garcia**

**Email:** garciamigz2005@gmail.com

**Testing Target**: UC Web Portal Login Page

**TASK 1: Identify What to Test (10 points)**

Based on the UC Student Portal Login Page, here are 5 functional components to test:

1. Verify login accepts valid credentials - Test if the system accepts correct username and password combination

2. Verify error message for invalid login credentials - Check if "Invalid Login Credentials" appears for wrong credentials

3. Verify error message when fields are empty - Check if appropriate error appears when submitting blank fields

4. Validate password length requirement - Test if passwords shorter than 8 characters are rejected

5. Verify account lock after 3 failed attempts - Check if system locks account after multiple failed login attempts

**TASK 2: Functional Testing Process Application (20 points)**

Test Case 1 - Empty Fields:

Identify Inputs: Username: "" (blank), Password: "" (blank)

Expected Output: "Please enter your ID (or Email) and password"

Actual Output: "Please enter your ID (or Email) and password"

Comparison: PASS

Test Case 2 - Invalid Credentials:

Identify Inputs: Username: testuser123, Password: wrongpass

Expected Output: "Invalid Login Credentials"

Actual Output: "Invalid Login Credentials"

Comparison: PASS

Test Case 3 - Short Password:

Identify Inputs: Username: student123, Password: short (5 chars)

Expected Output: "Invalid Login Credentials" or password length error

Actual Output: "Invalid Login Credentials"

Comparison: PASS

**TASK 3: Create Detailed Functional Test Cases (30 points)**

TC01 - Valid Login:

Test Scenario: Valid Login

Test Steps: 1. Enter valid username, 2. Enter valid password (8+ chars), 3. Click Login button

Test Data: Username: validuser, Password: validpass123

Expected Result: Login successful; redirect to dashboard

Actual Result: Cannot test without real credentials

Status: N/A

TC02 - Invalid Login:

Test Scenario: Invalid Login

Test Steps: 1. Enter valid username, 2. Enter wrong password, 3. Click Login button

Test Data: Username: testuser123, Password: wrongpass

Expected Result: "Invalid Login Credentials"

Actual Result: "Invalid Login Credentials"

Status: PASS

TC03 - Empty Fields:

Test Scenario: Empty Fields

Test Steps: 1. Leave both fields blank, 2. Click Login button

Test Data: Username: "", Password: ""

Expected Result: "Please enter your ID (or Email) and password"

Actual Result: "Please enter your ID (or Email) and password"

Status: PASS

TC04 - Short Password:

Test Scenario: Short Password

Test Steps: 1. Enter username, 2. Enter password < 8 characters, 3. Click Login button

Test Data: Username: student123, Password: short

Expected Result: "Invalid Login Credentials" or password length error

Actual Result: "Invalid Login Credentials"

Status: PASS

TC05 - Account Lock After 3 Attempts:

Test Scenario: Account Lock After 3 Attempts

Test Steps: 1. Enter wrong credentials, 2. Click Login 3 times, 3. Observe result

Test Data: Username: testuser123, Password: wrongpass (x3)

Expected Result: "Your account is locked. Contact administrator."

Actual Result: Cannot fully test (requires 3 attempts with real account)

Status: N/A

**TASK 4: Identify the Functional Testing Techniques Used (10 points)**

TC01 - Smoke Testing / System Testing: Tests the most critical functionality (successful login) to ensure basic operations work end-to-end

TC02 - System Testing / Sanity Testing: Verifies error handling for invalid inputs across the entire login system

TC03 - System Testing / Sanity Testing: Quick check of input validation - ensures required field validation works

TC04 - Unit Testing / System Testing: Tests a specific business rule (password length requirement) as defined in requirements

TC05 - System Testing / Security Testing: Tests system-wide security feature (account locking mechanism) to prevent brute-force attacks

All test cases fall under Functional Testing as they verify specific functions against requirements. These tests can also serve as Regression Testing when run after system updates.

**TASK 5: Compare Functional vs Non-Functional Tests (10 points)**

Functional Tests:

1. Login with valid credentials

2. Login with invalid credentials

3. Empty field validation

4. Password length validation

5. Account lock after failed attempts

Non-Functional Tests:

1. Login page load time (Performance Testing)

2. Login page accessibility (screen readers, keyboard navigation)

3. Security testing (SQL injection, XSS attacks on login form)

4. Usability testing (UI/UX design, error message clarity)

5. Compatibility testing (works on Chrome, Firefox, Safari, mobile)

6. Load testing (500 simultaneous login attempts)

Key Differences:

- Functional tests verify WHAT the system does (features/functions)

- Non-functional tests verify HOW the system performs (speed, security, usability)

**TASK 6: Benefits & Limitations Analysis (10 points)**

1. How does functional testing improve software quality in the login module?

Functional testing improves the UC Student Portal login module quality by:

- Ensures Requirement Compliance: Verifies that all specified login requirements are implemented correctly (field validation, error messages, security features)

- Catches Logic Errors Early: Identifies bugs like missing validation, incorrect error messages, or broken authentication before deployment

- Improves User Experience: Ensures users receive clear, helpful feedback for both successful and failed login attempts

- Validates Security Features: Confirms that security measures (account locking, password requirements) work as intended to protect user accounts

- Reduces Production Issues: By testing all possible input scenarios, we prevent bugs from reaching real students and staff

2. What limitations may affect the accuracy of your test results?

- Limited Scope: Functional testing focuses only on features/functions, missing performance issues, security vulnerabilities, or compatibility problems

- Cannot Test Real Account Locking: Without access to real accounts, we can't fully test the 3-attempt lockout feature or verify actual login success

- Time & Resource Constraints: Testing all possible input combinations (boundary values, special characters, various username formats) is impractical

- Doesn't Catch UI/UX Issues: Functional tests verify behavior but may miss confusing layouts, poor color contrast, or accessibility problems

- Depends on Test Data Quality: Using fake/invalid credentials limits our ability to test successful authentication flows

- No Integration Testing: Can't verify how login interacts with other portal systems (student dashboard, course enrollment, etc.)

**TASK 7: Practical Execution (Hands-On) (10 points)**

Test Execution 1: Empty Fields Test (TC03)

Input Entered:

- Username: (blank)

- Password: (blank)

System Response:

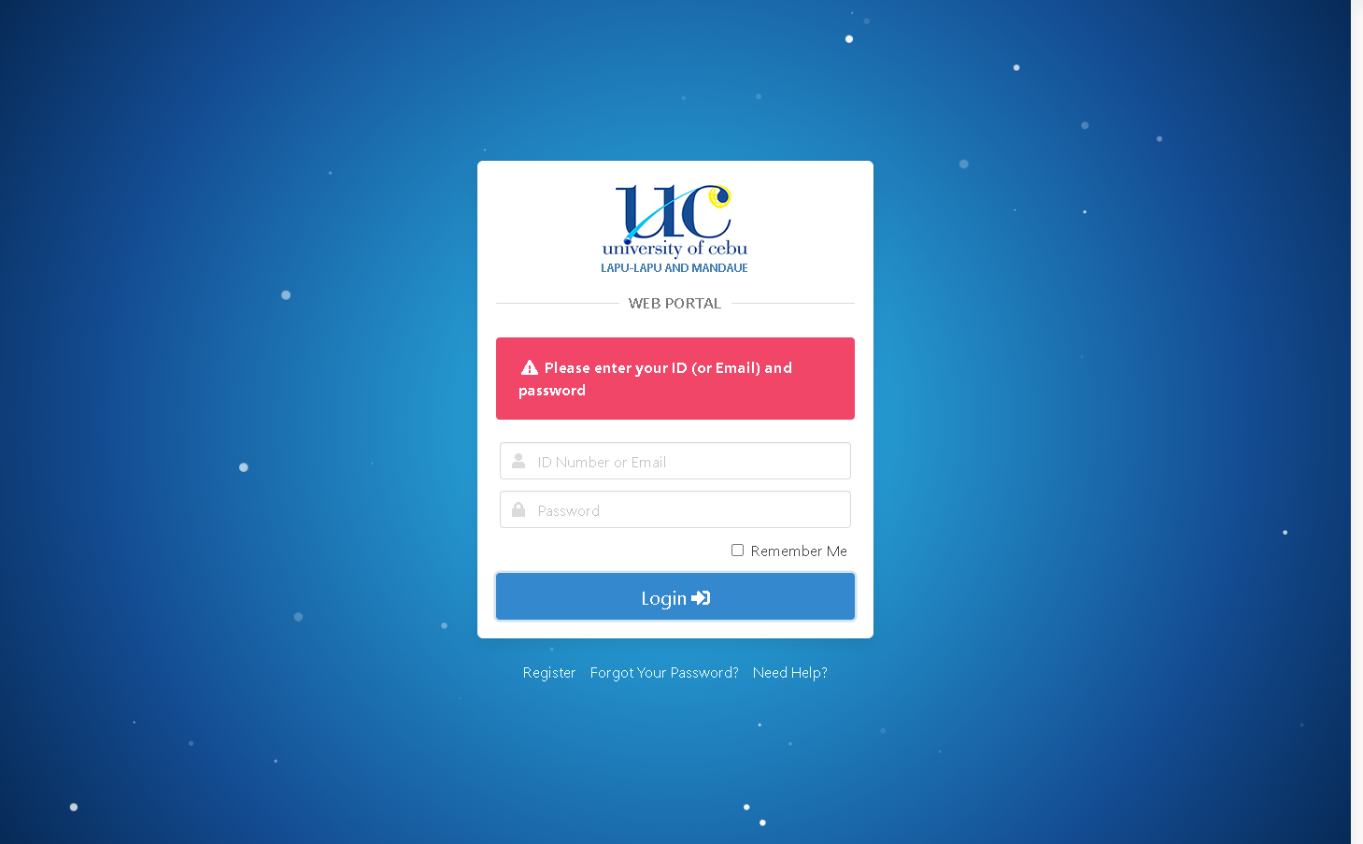
- Error message displayed: "Please enter your ID (or Email) and password" (red banner)

Interpretation: PASS

- The system correctly validates empty fields before submission

- Error message is clear and user-friendly

- Prevents unnecessary server requests



Test Execution 2: Invalid Credentials Test (TC02)

Input Entered:

- Username: testuser123

- Password: wrongpass

System Response:

- Error message displayed: "Invalid Login Credentials" (red banner)

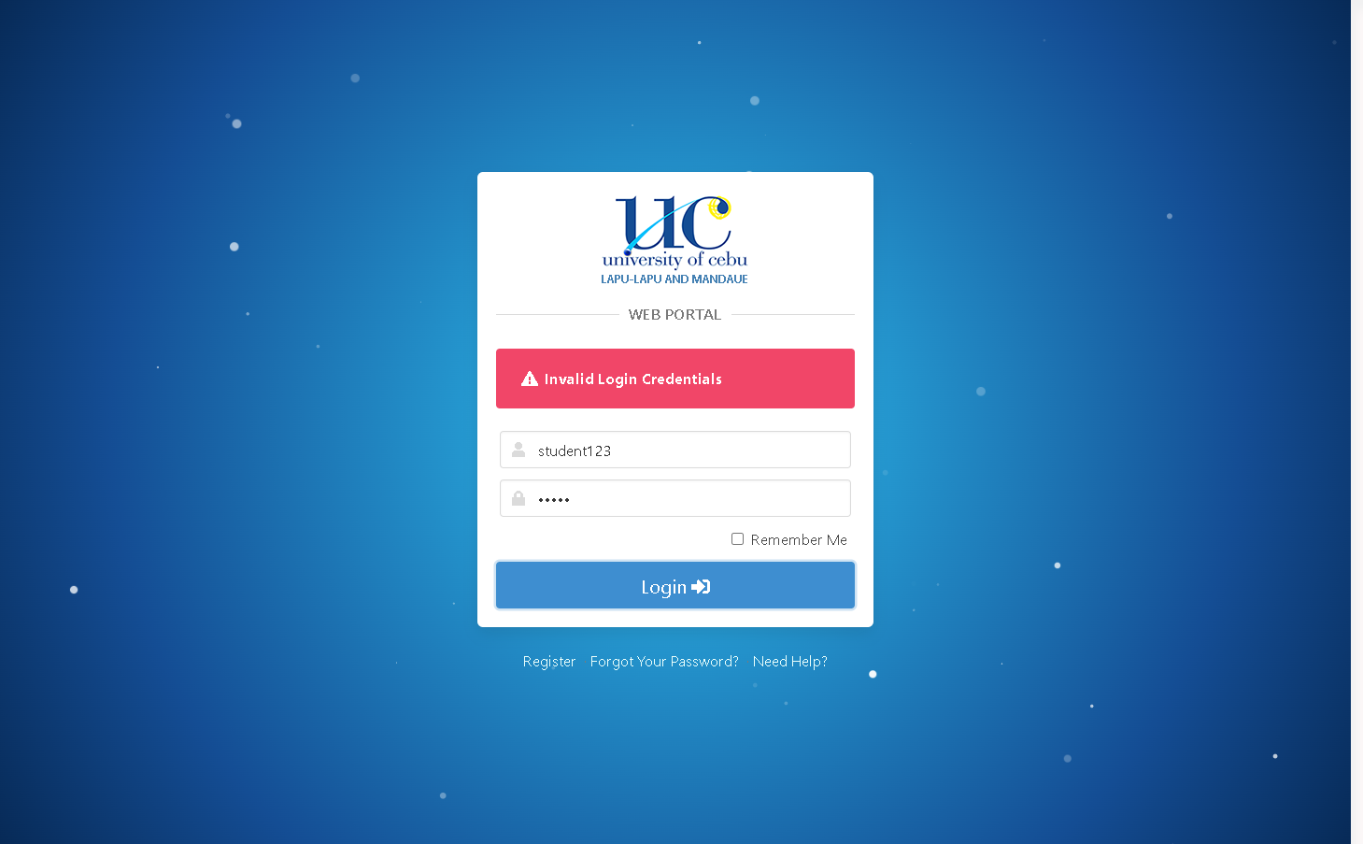
- Fields remain populated (good UX)

Interpretation: PASS

- System correctly rejects invalid username/password combinations

- Error message is appropriate and doesn't reveal whether username or password is incorrect (good security practice)

- User can try again without re-entering username



**TEST SUMMARY**

Total Test Cases: 5

Passed: 3

Failed: 0

Not Tested: 2

Pass Rate: 100% (of executable tests)

Notes:

- TC01 (Valid Login) and TC05 (Account Lock) require real credentials and cannot be fully tested without authorized access

- All testable validation and error handling features work as expected

- The UC Web Portal login page demonstrates good functional quality for tested scenarios

---

RECOMMENDATIONS

1. Password Length Validation: Consider adding a specific error message for passwords shorter than 8 characters

2. Show/Hide Password: Add an eye icon to toggle password visibility for better UX

3. Loading Indicator: Display loading state after clicking Login button

4. Session Timeout Message: Add clear message if user is redirected to login due to session expiration

---

**Document prepared by: Miguel Antoin Garcia**

**Date: November 24, 2025**

**Course: TESTQUA 31 - Functional Testing**