

# Laboratory Practical Assessment

**Topic : Functional Testing – Software Testing**

**Course : TESTQUA 31**

**Type : Laboratory Practical Assessment**

**Student Name: Jancent Corda**

## I. Instructions

1. Read each task carefully.
2. Perform the required activities using any tool (manual, spreadsheet, or simple code).
3. Submit:
  - PDF of your answers
  - Screenshots of executed tests (if applicable)
  - Completed test case table

---

## II. Laboratory Scenario

You are assigned as a **QA Tester** for a new web application: “**UC Student Portal Login Page**”.

The login page has the following requirements:

1. A user must enter a valid **username** and **password**.
2. If the username or password is incorrect, the system must display:  
**“Invalid login credentials.”**
3. If fields are empty, the system must display:  
**“All fields are required.”**
4. For security, passwords must be at least **8 characters** long.
5. After 3 failed attempts, the system must show:  
**“Your account is locked. Contact administrator.”**

Using the Functional Testing concepts from the document , perform the tasks below.

### III. Practical Tasks

---

#### Task 1: Identify What to Test (10 points)

Using the “What to Test in Functional Testing” section (Basic Usability, Main Functions, Accessibility, Error Handling) , list **at least 5 functional components** you will test in the login page.

Example: “Verify login accepts valid credentials.”

---

#### Task 2: Functional Testing Process Application (20 points)

Apply the 4-step functional testing process from the document (Identify input → Expected outcomes → Execute test → Compare output) .

Complete the table below for **3 test cases**:

Step	Test Case 1	Test Case 2	Test Case 3
Identify Inputs	Valid username and password	Invalid username or password	Empty username and password
Expected Output	Login success	Login fail	Please enter your ID (or Email) and password
Actual Output	Login success	Login fail	Please enter your ID (or Email) and passeorr
Comparison (Pass/Fail)	Pass	Pass	Pass

---

#### Task 3: Create Detailed Functional Test Cases (30 points)

Create **at least 5 functional test cases** using the following format:

**Test Case ID   Test Scenario   Test Steps   Test Data   Expected Result   Actual Result   Status**

Your test cases should cover:

- ✓ Valid login
- ✓ Invalid login

- ✓ Empty fields
- ✓ Short password
- ✓ Account lock after 3 attempts

Test Case ID	Test Scenario	Test Steps	Test Data	Expected Results	Actual Results	Status
--------------	---------------	------------	-----------	------------------	----------------	--------

<b>TC-001</b>	Valid login	1. Enter valid username and password. 2. Click Login.	Username: 45324562 Password: 123456	Directly opens dashboard	Directly opens dashboard	Pass
<b>TC-002</b>	Invalid login	1. Enter invalid username and password. 2. Click Login.	Username: 42564536 Password: wrong	Invalid login credentials	Invalid login credentials	Pass
<b>TC-003</b>	Empty fields	1. Leave both fields empty. 2. Click Login	Username: empty Password: empty	Please enter your ID (or Email) and password	Please enter your ID (or Email) and password	Pass
<b>TC-004</b>	Short password	1. Enter valid username and short password. 2. Click Login.	Username: 42564536 Password: 1234	Gives a message displaying “Password is too short”.	Invalid login credentials	Fail
<b>TC-005</b>	Account lock after 3	1. Enter invalid username	Username: 42564536	Account is	Invalid	

	failed attempts	and password. 2. Repeat thrice.	Password: Wrong	locked	login credentials	Fail
--	-----------------	------------------------------------	-----------------	--------	-------------------	------

---

#### Task 4: Identify the Functional Testing Techniques Used (10 points)

For each of your test cases in Task 3, classify them under appropriate techniques such as:

- Unit Testing
- Integration Testing
- System Testing
- Regression Testing
- Smoke Testing
- Sanity Testing
- User Acceptance Testing (UAT)

Use definitions from the document to justify your answers.

Test Case	Technique Used	Justification
Valid login	System Testing	Tests if the system accepts valid credentials
Invalid login	System Testing	Checks if the system gives an error message if encounters invalid credentials.
Empty fields	System Testing	Checks if the system gives an error message if it has empty fields.
Short password	Integration Testing	This test verifies the interaction between the login UI and the backend password validation module.

Account locked after 3 failed attempts	Integration Testing	Used when verifying interaction between multiple components.
--	---------------------	--

---

### Task 5: Compare Functional vs Non-Functional Tests (10 points)

Using the comparison table in the document , list **3 functional** and **3 non-functional** tests applicable to the login page.

Functional	Non-Functional
Valid Login	Empty Fields
Invalid Login	Short Password
Account locked after 3 failed attempts	Login response time

---

### Task 6: Benefits & Limitations Analysis (10 points)

Based on the “Benefits” and “Limitations of Functional Testing” sections of the document , answer the following:

1. How does functional testing improve software quality in the login module?
2. What limitations may affect the accuracy of your test results?

---

### Task 7: Practical Execution (Hands-On) (10 points)

Perform any **2** of your test cases manually (or using a mock HTML login page you create).

Submit **screenshots** showing:

- Input entered
- System response
- Your interpretation (Pass/Fail)

Test Case	Input Entered	System Response	Interpretation
Valid Login	Refer to the “Screenshot”	Refer to the “Screenshot”	Pass

	Folder	Folder	
Invalid Login	Refer to the “Screenshot” Folder	Refer to the “Screenshot” Folder	Pass