Login System Test Plan - Sample

Overview

This is a sample test plan for the Login System. It includes test cases designed to validate whether the system meets the functional requirement:

REQ-001
"The system must allow valid users to log in using their email and password. Invalid login attempts must be rejected with a clear error message."

Test Cases

Test Case ID	Requirement ID	Test Title	Objective	Preconditions	Test Steps	Expected Result	Priority	Owner
TC-001	REQ-001	Valid login	Verify login succeeds for valid user credentials	User has valid account	1. Enter valid email 2. Enter valid password 3. Click login	User is successfully logged in	High	QA Team
TC-002	REQ-001	Invalid password	Verify login fails with incorrect password	User has valid account	1. Enter valid email 2. Enter wrong password 3. Click login	Error: "Invalid username or password"	High	QA Team
TC-003	REQ-001	Invalid email format	Verify login fails for improperly formatted email	None	1. Enter malformed email 2. Enter any password 3. Click login	Error: "Invalid email format"	Medium	QA Team
TC-004	REQ-001	Blank fields	Verify login fails when email and/or password fields are blank	None	1. Leave email and/or password blank 2. Click login	Error: "Email and password cannot be blank"	High	QA Team
TC-005	REQ-001	Case sensitivity	Verify password field is	User has known	1. Enter valid email 2. Enter	Error: "Invalid	Medium	QA Team

Test Case ID	Requirement ID	Test Title	Objective	Preconditions	Test Steps	Expected Result	Priority	Owner
			case-sensitive	password with case	password in wrong case 3. Click login	username or password"		

Traceability Matrix

Requirement ID	Requirement Description	Test Case ID	Test Case Description	Status
REQ-001	Login and error handling logic	TC-001	Valid login	Not started
REQ-001	Login and error handling logic	TC-002	Invalid password	Not started
REQ-001	Login and error handling logic	TC-003	Invalid email format	Not started
REQ-001	Login and error handling logic	TC-004	Blank fields	Not started
REQ-001	Login and error handling logic	TC-005	Case sensitivity	Not started

Additional Test Cases (Analyst Improvements)

Test Case ID: TC-006

Requirement ID: REQ-001

Test title: Reject login with expired password

Objective: Verify the system blocks expired passwords and displays the correct message

Preconditions: User account exists with expired password

Test steps:
1. Enter valid email
2. Enter expired password
3. Click login

Expected result: System shows 'Password expired. Please reset.'

Priority: Medium

Owner: QA

Test Case ID: TC-007

Requirement ID: REQ-001

Test title: Lock account after multiple failed attempts

Objective: Ensure repeated failed logins trigger lockout

Preconditions: Valid account exists, lockout threshold = 5

Test steps:
1. Enter valid email + wrong password 5 times
2. Try with valid password

Expected result: Account locked; error 'Too many failed attempts, please reset or try later.'

Priority: High

Owner: QA

Test Case ID: TC-008

Requirement ID: REQ-001

Test title: Reject login with SQL injection string

Objective: Ensure input sanitization prevents injection attacks

Preconditions: N/A

Test steps:
1. Enter test@example.com' OR '1'='1
2. Any password
3. Click login

Expected result: Login rejected with generic error; no crash/traceback

Priority: High

Owner: QA

Updated Traceability Matrix

Requirement ID	Requirement description	Test case ID	Test case description	Status
REQ-001	Login and error handling logic	TC-001	Valid login	Not started
REQ-001	Login and error handling logic	TC-002	Invalid password	Not started

REQ-001	Login and error handling logic	TC-003	Invalid email format	Not started
REQ-001	Login and error handling logic	TC-004	Blank fields	Not started
REQ-001	Login and error handling logic	TC-005	Case sensitivity	Not started
REQ-001	Login and error handling logic	TC-006	Reject expired password	Not started
REQ-001	Login and error handling logic	TC-007	Lock account after multiple failures	Not started
REQ-001	Login and error handling logic	TC-008	Reject SQL injection attempt	Not started

Analyst Notes

Improvements made: Added edge case test cases (expired password, account lockout, SQL injection). Clarified preconditions where vague or missing. Expanded traceability matrix to ensure complete requirement coverage.

Gaps discovered: Original test plan missed security edge cases, lockout logic, and had inconsistent precondition details.

Recommendations: Standardize error message formats (clear, consistent, no sensitive information). Automate critical login tests (valid, invalid, lockout, injection) to strengthen QA readiness.