

TUTORIAL 2

2. Identify a Bug or Propose a New Feature in an Application with Supporting Test Cases

Objective:

To train students in **real-world software analysis** by identifying a bug or proposing a new feature and designing supporting test cases to verify and validate the behavior.

Sample Application Used:

Login and Registration Web Application (HTML + PHP or Python Flask)

- Use a basic version from GitHub or local copy (<http://localhost/sample-login-app/>)

Tools Required:

- Web Browser (Chrome/Firefox)
- Code Editor (VS Code/Sublime)
- Excel/Google Sheets for documenting test cases
- Optionally: DevTools (F12), Postman (for API), Selenium (for automation)

TASK 1: Identify a Bug

Step 1: Explore the Application

Access Login and Registration pages.

Try these scenarios:

- Register with existing email
- Leave password blank
- Enter a weak password like 123

- Try login with SQL injection string: ' OR '1'='1



Observed Bug:

Bug Title: Registration accepts weak passwords like 123 without any validation.

Bug Report:

| Field | Details |
|--------------------|---|
| Bug ID | BUG-2025-001 |
| Bug Title | Weak passwords allowed in registration |
| Module | Registration Page |
| Environment | Chrome v116, Windows 10 |
| Steps to Reproduce | 1. Open registration page 2. Enter valid email 3. Enter password 123 4. Click Submit |
| Expected Result | Error message: "Password too weak" |
| Actual Result | Registration successful with password 123 |
| Priority | High |
| Status | Open |

Supporting Test Cases:

| TC ID | Module | Test Case Description | Input | Expected Result | Actual Result | Status |
|-------|--------------|-------------------------------|--|---------------------------------|----------------------|--|
| TC01 | Registration | Register with weak password | Email: user@test.com Password: 123 | Show error: "Password too weak" | Registration success |  Fail |
| TC02 | Registration | Register with strong password | Email: user@test.com Password: Abc@1234 | User registered successfully | User registered |  Pass |

Suggested Fix:

Implement password strength validation:

- At least 8 characters
- Include uppercase, lowercase, digits, and special characters
- Use JavaScript + server-side validation

TASK 2: Propose a New Feature

Observed Issue:

No **"Forgot Password"** functionality exists. Users who forget their password can't recover their account.

Proposed Feature: Add "Forgot Password" Link

Feature Description:

| Field | Details |
|---------------|---|
| Feature Title | Forgot Password Function |
| Location | Login Page |
| Description | Add a link under Login form: "Forgot Password?" On click: take user to password reset page |
| Benefit | Improves usability and account recovery |
| Priority | Medium |

Supporting Test Cases:

| TC ID | Module | Test Case Description | Input | Expected Result | Status |
|-------|-----------------|----------------------------------|----------------------|---|--------|
| TC01 | Forgot Password | Click on "Forgot Password?" link | Click | Redirects to Reset Password Page | |
| TC02 | Forgot Password | Submit valid email for reset | Email: user@test.com | Show message: "Reset link sent to your email" | |

| | | | | |
|------|-----------------|-------------------------------------|------------------------|---|
| TC03 | Forgot Password | Submit invalid email | Email: test@fake.com | Show message: "Email not found in our system" |
| TC04 | Forgot Password | Reset password with strong password | New Password: Abc@1234 | Show success message: "Password changed" |
| TC05 | Forgot Password | Reset password with weak password | New Password: 123 | Show error: "Password too weak" |

Summary for Students

| Activity | Result |
|-------------------------------|------------------------------------|
| Bug Identified | Weak password accepted |
| Bug Severity | High |
| New Feature Proposed | Forgot Password Function |
| Supporting Test Cases Created | Yes – for both Bug and New Feature |
| Tools Used | Browser, Excel, Editor |

Student Submission Format

Students should submit:

- 1 PDF containing:
 - Bug Report
 - Proposed Feature Description
 - Test Case Table (for both bug + feature)
 - Screenshots