

SDLC ASSIGNMENT – 5

Skill Description:

Gain practical experience in applying Software Development Life Cycle (SDLC) methodologies, including testing execution, incremental development, and Rapid Application Development (RAD). Develop hands-on skills by designing automated testing suites, incrementally enhancing project management tools, and prototyping social media apps with user-centric iterations.

Problem Statement 5:

Design an automated testing suite for the login and fund transfer functionalities of a banking application. Include test cases for scenarios like multiple concurrent user logins, incorrect PIN entries, and fund transfer validations.

Learning Outcomes:

- Apply automated testing concepts to real-world scenarios.
- Demonstrate the ability to handle various testing scenarios in a banking application.

Automated Testing Suite for Banking Application

1. Test Plan Overview

- Objective: Ensure the reliability, security, and accuracy of login and fund transfer functionalities.
- Scope: Cover functional, performance, and security aspects.
- Tools: Selenium, JUnit, Postman, and TestNG.
- Test Environment: Simulated banking environment with test accounts and secure connections.

2. Test Scenarios and Cases

A. Login Functionality

1. Valid User Login
 - Test Case: Verify successful login with correct credentials.
 - Expected Result: User is redirected to the dashboard.

2. Invalid PIN Entry
 - Test Case: Attempt login with incorrect PIN three times.
 - Expected Result: Account is locked after three failed attempts.
3. Multiple Concurrent Logins
 - Test Case: Attempt multiple logins from different devices simultaneously.
 - Expected Result: Allow only one active session per user or prompt session takeover warning.
4. Session Timeout
 - Test Case: Simulate user inactivity beyond the session timeout limit.
 - Expected Result: User is logged out automatically.

B. Fund Transfer Functionality

1. Successful Fund Transfer
 - Test Case: Transfer funds between valid accounts.
 - Expected Result: Transaction success message displayed, updated account balances.
2. Insufficient Funds
 - Test Case: Attempt transfer exceeding available balance.
 - Expected Result: Transaction is declined, appropriate error message displayed.
3. Invalid Account Number
 - Test Case: Enter an incorrect recipient account number.
 - Expected Result: Transaction fails with an error message.
4. Transfer Limit Exceeded
 - Test Case: Attempt transfer above daily transaction limit.
 - Expected Result: Transfer is blocked with a relevant notification.
5. Security Validation
 - Test Case: Ensure secure data transfer using encryption.
 - Expected Result: No sensitive data is transmitted in plain text.

3. Test Execution and Reporting

- Test Execution: Automate using testing tools and schedule regular test runs.
- Defect Reporting: Use Jira for tracking issues, including severity and resolution status.

4. Conclusion

By implementing this automated testing suite, the banking application's login and fund transfer functionalities will be thoroughly validated, ensuring a secure and seamless user experience while mitigating potential risks.