**Facebook Login and Signup Testing Project- Overview**

**Project Title:** Facebook Login and Signup Functionality Testing  
 **Project ID:** FB-AUTH-TP-001

## **Name:** Project Overview Document **Author:** Dalonda Ikhimokpa **Version:** 1.0 **Date Created:** February 1, 2025

### 1. Objective

The objective of this QA project is to validate the **login** and **signup functionalities** of Facebook’s web application. The testing will ensure that users can successfully register, log in, and receive clear feedback when errors occur. The features will be tested for correct behavior, input validation, and UI messaging, using structured test cases and defect tracking. This effort aims to catch functional issues early, increase confidence in release readiness, and ensure an optimal user experience across desktop and mobile web platforms.

### 2. Scope of Testing

**In Scope:**

* Functional testing of the login and signup workflows
* Validation of input fields (e.g., email, password)
* UI error messages and edge case behavior
* Password reset initiation
* Testing across Google Chrome (Windows) and Firefox (Android)

**Out of Scope:**

* Post-login features (e.g., newsfeed, messaging, user settings)
* Backend, API, or database-level testing
* Load and performance testing
* Native mobile app (iOS/Android) functionality
* Third-party authentication (e.g., Google, Apple login)

### 3. Test Environment

**Browsers & Platforms:**

* Google Chrome (Latest) on Windows 10
* Firefox on Android 12 (Mobile Browser Testing)

**Test Environment Details:**

* Staging or QA build that mirrors production environment
* Use of dummy/test email accounts for verification processes

### 4. Tools & Resources

|  |  |
| --- | --- |
| **Tool** | **Purpose** |
| **Trello** | Task and test management |
| **Microsoft Excel** | Test case and defect documentation |
| **Chrome / Firefox** | Browser compatibility testing |
| **Disposable Email Services** | For signup email validation |

### 5. Stakeholders

|  |  |
| --- | --- |
| **Role** | **Name** |
| QA Engineer | Dalonda Ikhimokpa |
| QA Lead / Manager | Dalonda Ikhimokpa |
| Developer Contact | [To be assigned] |
| Product Owner | [To be assigned] |

### 6. Assumptions

* All login/signup features are complete and available in the QA environment
* The QA environment is stable and reflective of the production setup
* Necessary credentials and test data will be available for execution
* Email delivery systems are functional for testing verification flows

### 7. Risks & Mitigation

|  |  |  |
| --- | --- | --- |
| **Risk** | **Impact** | **Mitigation** |
| Incomplete or unstable features | High | Confirm feature readiness before execution |
| Browser compatibility issues | Medium | Perform cross-browser testing (Chrome/Firefox) |
| Email delivery delays | Medium | Use disposable email tools for faster verification |
| Ambiguity in requirements | Medium | Clarify flows with dev/product teams before test case design |

### 8. Deliverables

* Test Case Document (Excel)
* Bug Report Log (Excel or integrated into Trello/Jira)
* Test Summary Report
* QA Sign-off and Recommendation

**9. Test Strategy**

Our testing approach will incorporate multiple testing techniques to ensure comprehensive coverage:

**9.1 Testing Types**

* Functional Testing: Verify all features work according to requirements
* Usability Testing: Ensure intuitive user interface and experience
* Compatibility Testing: Verify functionality across browsers and devices
* Performance Testing: Validate system behavior under load
* Security Testing: Ensure protection of sensitive data
* Regression Testing: Verify new changes don't break existing functionality
* Integration Testing: Test interactions between different system components

**9.2 Testing Techniques**

* Equivalence Class Partitioning
* Boundary Value Analysis
* Decision Table Testing
* State Transition Testing
* Use Case Testing
* Exploratory Testing
* Error Guessing

**9.3 Testing Approach**

1. Shift-Left Testing: Begin testing early in the development cycle
2. Context-Driven Testing: Adapt testing approach based on application context
3. Risk-Based Testing: Prioritize test cases based on risk assessment
4. End-to-End Flow Testing: Test complete user journeys

### 10. Test Schedule

|  |  |  |
| --- | --- | --- |
| **Phase** | **Start Date** | **End Date** |
| Test Planning | Feb 10, 2025 | Feb 14, 2025 |
| Test Case Design | Feb 15, 2025 | Feb 18, 2025 |
| Test Execution | Feb 19, 2025 | Feb 25, 2025 |
| Bug Reporting & Fix Verification | Feb 26, 2025 | Mar 1, 2025 |
| Test Closure | Mar 2, 2025 | Mar 3, 2025 |

### 11. Approval & Sign-Off

QA efforts will be considered complete when:

* All planned test cases have been executed
* All critical and high-severity defects are resolved or deferred with sign-off
* A formal Test Summary Report has been submitted and reviewed
* QA and Product teams jointly approve the closure of the testing phase

**Approval**

**Test Lead:** *(Dalonda Ikhimokpa)***Approval Date:** *(15/2/25*