**RepMove**

**Test Plan**

**1. Objective**

To verify that the Registration and Login functionalities work as expected, ensuring secure, user-friendly, and error-resistant interactions.

**2. Scope**

* Testing of user registration (sign-up) functionality.
* Testing of user login functionality with various input scenarios.
* Input validations and error handling(empty fields, invalid format)

**3. Feature to be tested**

**A. Registration Functionality**

**Positive Test Cases**

| **TC ID** | **Test Case Description** | **Input Data** | **Expected Result** |
| --- | --- | --- | --- |
| REG-001 | Register with valid data | Valid email, password | User is registered and redirected dashboard |
| REG-002 | Password meets complexity requirements | Strong password (e.g. P@ssw0rd123) | Registration successful |
| REG-003 | Register with email containing special characters | user.name+test@example.com | Registration successful |
| REG-004 | Registration with optional fields left blank | Fill only mandatory fields | Registration successful |

**Negative Test Cases**

| **TC ID** | **Test Case Description** | **Input Data** | **Expected Result** |
| --- | --- | --- | --- |
| REG-005 | Register with already used email | Duplicate email | Error: “Invalid to sign up” |
| REG-006 | Register with invalid email format | user@.com, user@com | Error: “Invalid email address” |
| REG-007 | Weak password | 123, abc | Error: “Min length for password is 5” |
| REG-009 | Blank fields | Empty email/password | Error for each empty mandatory field error: “field is required” |
| REG-010 | SQL Injection in input | ' OR '1'='1 | Input sanitized, error shown or validation failure Invalid to sign up |

**B. Login Functionality**

**Positive Test Cases**

| **TC ID** | **Test Case Description** | **Input Data** | **Expected Result** |
| --- | --- | --- | --- |
| LOG-001 | Login with valid credentials | Valid email & password | User is logged in and redirected to dashboard |

**Negative Test Cases**

| **TC ID** | **Test Case Description** | **Input Data** | **Expected Result** |
| --- | --- | --- | --- |
| LOG-004 | Invalid email | Wrong email | Error: “ Invalid to login ” |
| LOG-005 | Incorrect password | Wrong password | Error: “Invalid to login” |
| LOG-006 | Blank email or password | One or both fields empty | Error: “email/password is required” |
| LOG-007 | SQL Injection attempt | ' OR '1'='1 | Login denied, input sanitized |
| LOG-008 | Login with unregistered email | Not in database | Error Invalid to login |

**4. Feature not to be tested**

* Forgot password
* Sign In with Google, Apple, Microsoft
* Sign Up with Google, Apple, Microsoft

**5.Approach**

An analytical approach it’s a risk-based or requirements-based strategy for testing, focusing testing efforts on critical functionalities by analyzing risks or requirements to identify test conditions. It is a systematic methodology within a test plan document that leverages analytical techniques, data analysis, and predictive insights to optimize testing, ensuring thorough coverage of potential defects and system risks

**6.Item Pass/Fail Criteria**

- 90% of the test cases must pass

- all test cases dealing with critical functionality must pass

- all medium and high severity defects must be fixed

- test coverage must be at least 90%

**7. Deliverables**

* Test Plan
* Test Cases Document
* Bug Reports
* Traceability matrix
* Test Summary Report

**8. Test Environment**

* **Device/Browser:** Chrome
* **Platform:** Web
* **Tools:** Playwright, TypeScript

**9. Responsibilities**

|  |  |
| --- | --- |
| Test analysis, planning | Valerii Silvanovych |
| Test design | Valerii Silvanovych |
| Test implementation | Valerii Silvanovych |
| Test execution | Valerii Silvanovych |
| Evaluating exit criteria | Valerii Silvanovych |
| Reporting | Valerii Silvanovych |

**10. Schedule**

|  |  |
| --- | --- |
| **Test analysis, planning:** | * Creating of test plan * Clarifying requirements for new planned features * Defining approach, test techniques, test types for each functionality |
| **Test design** | * Creation of traceability-matrix, high -level test cases are planned |
| **Test implementation** | * Creation of a check-lists |
| **Test execution** | * Holding Smoke testing * Formal functional, UI testing,using check-lists * Issue reporting * Holding confirmation testing after bug-fixed * Holding Regression testing |
| **Evaluating exit criteria** | * Acceptance testing |
| **Reporting** | * Test Report |

**11. Risks and Contingencies**

Risk 1:

Critical Bug Discovery

Risk: A severe bug is found late in the testing cycle, threatening the scheduled product launch date.

Contingency: Allocate a small portion of the project budget and time for critical bug fixes and immediate patch deployment. The contingency plan would also include a fast-track process for approving and releasing a hotfix.

Risk 2:

Test Environment Unavailability

Risk: The designated test environment becomes unavailable or unstable, halting testing activities.

Contingency: Have an alternative test environment or cloud-based testing environment available as a backup. The plan also includes a communication channel to quickly resolve environment issues with the IT support team.

Risk 3:

Key Tester Unavailability

Risk: A key QA engineer is unable to perform assigned testing tasks due to illness or other unforeseen circumstances.

Contingency: Cross-train other team members on the critical testing areas. Have a list of backup testers available or prioritize critical test cases that can be reassigned

**12. Entry/Exit Criteria**

**Entry:**

* UI and backend are ready.
* Test environment is configured.

**Exit:**

* All test cases executed.
* Critical bugs are fixed and verified.
* Regression tested.

**13. Approvals**

|  |  |
| --- | --- |
| PM: | Valerii Silvanovych |
| Customer: | Valerii Silvanovych |