**Business Requirements Document (BRD) for QA Testing**

**1. Introduction**

* **Project Name:** Web Application QA Automation
* **Prepared By:** QA Team
* **Date:** [23/03/2025]
* **Version:** 1.0
* **Objective:** The purpose of this document is to define the business and functional requirements for QA testing of a web application, ensuring it meets the expected quality standards before release.

**2. Business Requirements** The application should be thoroughly tested to ensure its functionality, performance, security, and user experience meet the required standards. The key requirements include:

* Ensuring compatibility across multiple browsers such as Chrome, Firefox, and Edge, as well as different devices including desktops, tablets, and mobile phones.
* Implementing automated scripts to capture screenshots for every user click and interaction within the application.
* Verifying user authentication and access control mechanisms to prevent unauthorized access.
* Testing form submissions with valid and invalid data to validate proper error handling and data processing.
* Ensuring accurate database storage and retrieval for submitted data.
* Conducting API response validation to ensure expected data exchange between the frontend and backend.
* Performing error handling tests to confirm appropriate messages are displayed for invalid inputs.
* **Adding the Values 1, 2, and 3 to validate proper aggregation functionality.**
* **Adding results with point 5 from the first requirement to ensure correct calculations.**

**3. Functional Requirements**

* **User Login:** Users should be able to log in with valid credentials.
* **Form Submission:** Users should be able to submit forms with valid data.
* **Data Validation:** Submitted data should be stored correctly in the database.
* **Screenshot Capture:** Each user interaction should be logged with screenshots.
* **Error Handling:** System should provide appropriate error messages for invalid inputs.

**Requirements:**

* **Adding the sum of Values 1, 2, and 3 to validate proper aggregation functionality.**
* **Adding results with Value 5 from the first requirement to ensure correct calculations.**

**Functionality:**

1. Add 1,2 and 3, repeat the functionality with all possibilities
2. Add 5 with point 1 results aggregation

**Case Creation:**

Create no of cases at possibilities with help of Functionality points

Table format:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Test Case ID** | **Test Scenario** | **Test Steps** | **Expected Result** | **Status** |
| TC\_001 | Add 1 with 2 | A=1, B=2 | 3 | Pass |

**5. Assumptions & Dependencies**

* The test environment is stable and mirrors the production environment.
* Test data is available and accessible.
* Required automation tools (Selenium, Puppeteer) are installed.

**6. Conclusion**

* This document provides a structured approach to QA validation.
* Test cases ensure all functionalities are verified before release.
* Any issues found should be documented and reported for resolution.