Software Requirement Specification Document

# 1. Introduction

This document outlines the requirements for the Login, Registration, and Dashboard functionalities of the system. The document includes the expected behaviors, test cases, and acceptance criteria for each of the mentioned features. It is intended to ensure that the system meets the user's needs and is thoroughly tested for functionality.

# 2. Login Requirements

## 2.1 Functional Requirements

The login page should allow users to authenticate themselves using their email and password. It should provide a mechanism for users to reset their password and access the system after successful authentication.

## 2.2 Test Cases for Login

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Expected Outcome | Status |
| TC\_Login\_01 | Verify that the login form displays correctly. | The login form should contain fields for email, password, and a login button. |  |
| TC\_Login\_02 | Test login with valid credentials. | User should be redirected to the dashboard page. |  |
| TC\_Login\_03 | Test login with invalid credentials. | An error message should be displayed. |  |
| TC\_Login\_04 | Test login with empty email field. | An error message should appear indicating that the email is required. |  |
| TC\_Login\_05 | Test login with empty password field. | An error message should appear indicating that the password is required. |  |
| TC\_Login\_06 | Test login with incorrect email format. | The email field should display an error message for incorrect format. |  |
| TC\_Login\_07 | Test forgot password functionality. | When a valid email is entered, a password reset email should be sent. |  |
| TC\_Login\_08 | Test forgot password with empty email field. | An alert should prompt the user to enter an email for password reset. |  |

# 3. Registration Requirements

## 3.1 Functional Requirements

The registration page should allow new users to sign up by providing necessary details, including name, email, password, and password confirmation. It should validate the data, and provide clear error messages for incorrect inputs, such as mismatched passwords or invalid email format.

## 3.2 Test Cases for Registration

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Expected Outcome | Status |
| TC\_Register\_01 | Verify that the registration form displays correctly. | The form should display fields for name, email, password, and confirm password. |  |
| TC\_Register\_02 | Test registration with valid credentials. | User should be successfully registered and redirected to the login page. |  |
| TC\_Register\_03 | Test registration with existing email. | An error message should appear indicating that the email is already in use. |  |
| TC\_Register\_04 | Test registration with mismatched password and confirm password. | An error message should be displayed indicating that passwords do not match. |  |
| TC\_Register\_05 | Test registration with empty email field. | An error message should appear indicating that the email is required. |  |
| TC\_Register\_06 | Test registration with empty password field. | An error message should appear indicating that the password is required. |  |
| TC\_Register\_07 | Test registration with invalid email format. | The email field should display an error message for incorrect format. |  |
| TC\_Register\_08 | Test successful registration with valid data and check for login redirection. | The user should be redirected to the login page after successful registration. |  |

# 4. Dashboard Requirements

## 4.1 Functional Requirements

The dashboard should display different content depending on the user’s role (Student or Professor). Students should see their enrolled courses, while professors should see the courses they are teaching, with an option to manage students in each course. The system should handle loading and error states appropriately, ensuring a smooth user experience.

## 4.2 Test Cases for Dashboard

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case ID | Description | Expected Outcome | Status |
| TC\_Dashboard\_01 | Verify that the correct role (Student/Professor) is displayed. | Displays 'Welcome, Student!' or 'Welcome, Professor!' based on the user's role. |  |
| TC\_Dashboard\_02 | Verify that the user's courses are displayed correctly. | Lists the courses associated with the user, displaying their names and descriptions. |  |
| TC\_Dashboard\_03 | Test the 'Manage Students' button for professors. | Professors should see a 'Manage Students' button and clicking it should show an alert. |  |
| TC\_Dashboard\_04 | Verify behavior when no courses are available for the user. | Displays a message 'No courses found.' if the user has no courses. |  |
| TC\_Dashboard\_05 | Test the loading state while data is being fetched. | A loading message 'Loading dashboard...' should be displayed while the data is loading. |  |
| TC\_Dashboard\_06 | Test error handling during data fetching. | Any errors during data fetching should be handled gracefully (e.g., error message displayed in console or UI). |  |
| TC\_Dashboard\_07 | Test the redirect functionality for users not logged in. | If the user is not logged in, they should be redirected to the login page. |  |

# 5. Non-Functional Requirements

## 5.1 Usability

The login, registration, and dashboard pages should be user-friendly and responsive, adapting to different screen sizes and devices.

## 5.2 Performance

The system should respond within 3 seconds for login, registration, and dashboard loading actions.

## 5.3 Security

All user passwords should be securely hashed and never stored in plaintext. Authentication and authorization should be enforced to ensure users can only access resources they are permitted to view.

## 5.4 Error Handling

Proper error handling mechanisms should be in place to ensure a smooth user experience even in case of unexpected errors.

# 6. Conclusion

This document specifies the functional and non-functional requirements for the login, registration, and dashboard functionalities of the application. The test cases outlined in this document should be executed to ensure the system meets the expected behavior and performance criteria.