



# **Business Requirement Document (BRD)**

**Project Title: User Management System Development**

**Document Version: 1.0**

**Prepared By: Abirami Dhevi**

**Date: 22-Apr-2024**

## 1. Executive Summary

The User Management Application aims to provide users with a simple interface to manage a list of users. Users can add, edit, and delete entries efficiently. This application will enhance user management processes for small teams and personal projects.

## 2. Business Objectives

- Provide an easy-to-use interface for managing user information.
- Allow users to perform CRUD operations seamlessly.
- Ensure data is handled efficiently without backend storage (initially).
- Create a responsive design suitable for various devices.

## 3. Scope

### In-Scope:

- Development of a web-based User List application.
- Implementation of CRUD functionalities:
  - Create: Add a new user to the list.
  - Read: Display a list of users.
  - Update: Edit user details.
  - Delete: Remove a user from the list.
- Basic styling for user interface using CSS.
- Implementation of tests using Cypress.

### Out-of-Scope:

- User authentication and authorization.
- Backend database integration (initial version).
- Advanced styling or frameworks (focus on basic functionality).

## 4. Stakeholders

- **Project Manager:** Oversee project execution and delivery.
- **Development Team:** Responsible for building the application.
- **Quality Assurance Team:** Ensure application functionality and performance.
- **End Users:** Individuals who will use the application for user management.

## 5. Requirements

### 5.1 Functional Requirements

#### 1. User Interface

- The application must have an input field for user names.
- A button must be present to submit new users.
- A list displaying all users with options to edit or delete each entry.

#### 2. CRUD Operations

- **Create:** Users can enter a name and add it to the list.
- **Read:** The application displays the current list of users.
- **Update:** Users can click an "Edit" button to change a user's name.
- **Delete:** Users can click a "Delete" button to remove a user from the list.

#### 3. Validation

- Input field must not accept empty names.
- A confirmation prompt should appear before deleting a user.

### 5.2 Non-Functional Requirements

- The application should load within 2 seconds.
- The interface should be user-friendly and intuitive.
- The application should be compatible with major browsers (Chrome, Firefox, Edge).

## 6. User Stories

1. **As a user,** I want to add a new user to the list, so that I can keep track of my contacts.
2. **As a user,** I want to view all users in a list, so that I can manage them easily.
3. **As a user,** I want to edit user details, so that I can update their information.
4. **As a user,** I want to delete a user, so that I can remove outdated entries.

## 7. Acceptance Criteria

1. **User can add a user:**
  - User enters a name and clicks "Add User."
  - The name appears in the user list.
2. **User can edit a user:**
  - User clicks "Edit" next to a name.
  - A prompt appears, allowing the user to enter a new name.
  - The user list updates with the new name.
3. **User can delete a user:**
  - User clicks "Delete" next to a name.
  - A confirmation prompt appears.
  - If confirmed, the user is removed from the list.
4. **Input validation:**
  - The application does not allow empty submissions.
  - An error message is displayed if a user attempts to submit an empty name.

## 8. Timeline

Task	Duration	Start Date	End Date
Requirements Gathering	1 week	TBD	TBD
Development	3 weeks	TBD	TBD
Testing	1 week	TBD	TBD
Deployment	1 week	TBD	TBD

## 9. Risks

- **Limited User Adoption:** Users may find the application too basic.
- **Technical Limitations:** Challenges in implementing more complex features later without a backend.