 **COMSATS University Islamabad, Vehari Campus**

**Department of Computer Science**

**Lab Assignment 2**

**Spring-2025**

**CSC337 Advanced Web Technologies**

**Class: BSSE Instructor: Yasmeen Jana Assignment given: 01/06/2025**

**Marks: 100 Submission Date: 13/06/2025**

**Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Reg. No: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

|  |  |  |
| --- | --- | --- |
| **CLO NO** | **Course Learning Outcome** | **Blooms Taxonomy Learning Level** |
| **CLO6** | Develop dynamic clients using modern development technologies. | **Creating** |

**Note:** You must submit a ZIP file containing the complete source code of your project. Additionally, include the following:

* A link to your GitHub repository
* A PDF document with your source code and explanation
* A demonstration video showcasing the working application

**Assignment Scenario**

You are employed as a front-end developer at a startup company. The company is building a web-based Product Management System. The system should allow users to manage products including creating, reading, updating, and deleting products in a seamless and responsive single-page application (SPA).

Your application must be developed using the latest features of React, including hooks, component-based architecture, routing, form handling, state management using Redux, API integration using Axios/Fetch, and styling with modern CSS frameworks. You will also write unit tests for key components and deploy the application.

**Marking Rubric (Total: 100 Marks)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr. No.** | **Assessment Criteria** | **Description** | **Marks** |
| 1 | **Project Setup** | Project initialized with create-react-app, well-structured folder system, dependencies installed | 5 |
| 2 | **Product List Page** | Display products using Axios/Fetch, list view with Edit, Delete, and View Details buttons | 10 |
| 3 | **Add Product Functionality** | Controlled form, input validation, and integration with API/state | 10 |
| 4 | **Edit Product Functionality** | Pre-filled form with update feature | 10 |
| 5 | **Product Detail View** | Dynamic routing to display full product info | 5 |
| 6 | **Delete Product** | Deletion with confirmation, updates state/API correctly | 5 |
| 7 | **React Router Implementation** | react-router-dom used for SPA navigation including dynamic routes | 5 |
| 8 | **Hooks Usage** | Effective use of useState, useEffect, and optionally useRef | 10 |
| 9 | **Reusable Components** | Reusable ProductForm used in Add/Edit | 5 |
| 10 | **Error and Loading Handling** | Spinners or messages during async operations; proper error display | 5 |
| 11 | **Redux Toolkit for State Management** | Global state managed via Redux Toolkit | 5 |
| 12 | **Async Actions with Redux Thunk** | API calls handled using redux-thunk middleware | 5 |
| 13 | **Form Libraries (Formik or React Hook Form)** | Add/Edit forms implemented using modern form library with validation | 5 |
| 14 | **Unit Testing** | Minimum 2 components tested with Jest and React Testing Library | 5 |
| 15 | **Responsive UI/UX Design** | User-friendly, mobile-responsive interface using Tailwind or Bootstrap | 5 |
| 16 | **Report and Documentation** | Detailed report with project overview, structure, features, screenshots, and reflections | 5 |