

Experiment No. – 1				
Date of Performance:	11/07/2025			
Date of Submission:	18/07/2025			
Execution (06)	Viva (02)	Professional Ethics (02)	Experiment Total (10)	Sign with Date

Experiment No. 1

1.1 Aim: Apply functional and class components to include reusable UI blocks like navigation, events, lists and forms for developing a dynamic Responsive Web Application in React JS.

1.2 Course Outcome:

Apply foundational React concepts to build functional front-end applications.

1.3 Learning Objectives:

To Apply functional and class components to include reusable UI blocks.

1.4 Requirement:

Software: Node.js, React.js, Code Editor (VS Code).

Hardware: PC/Laptop with minimum 4 GB RAM.

Packages: React Router (for navigation), CSS/Bootstrap (for styling).

1.5 Related Theory:

- 2 • **Functional Components:** The primary building blocks of modern React. They are simple JavaScript functions that accept properties (**props**) and return JSX (UI elements). They use **Hooks** (like `useState` and `useEffect`) for local logic.
- 3 • **Redux Toolkit:** The recommended library for managing **global state** (data needed across many components, like `isAuthenticated`). It uses a central `store` and `slices` to ensure predictable state updates.
- 4 • **React Router:** A standard library used to define application routes, allowing the user to navigate between different views (pages) without forcing a full browser refresh.
- 5 • **Conditional Rendering:** The technique used in the `Header.jsx` to dynamically show either the **Login/Signup** links or the **Profile/Upload/Logout** links based on the user's Redux state.

5.1 Procedure:

1. Install Node.js and set up a React project using `npm create-react-app`.

2. Create a **functional component** for navigation bar.
3. Create a **class component** for displaying a list of items.
4. Add an **event handler** (e.g., button click event).
5. Create a **form component** to collect user input.
6. Style components using CSS/Bootstrap for responsiveness.
7. Run the application with `npm start` and verify the output.

5.2 Program and Output:

Header.jsx:

```

client > src > components > Headerjsx > Navbar
1  import React from "react";
2  import { Link, useNavigate } from "react-router-dom";
3  import { FaSearch } from "react-icons/fa";
4  import { MdOutlineFileUpload } from "react-icons/md";
5  import { GiHamburgerMenu } from "react-icons/gi";
6  import { useDispatch, useSelector } from "react-redux";
7  import { removeUserData, setUserData } from "../Redux/slices/user-slice";
8
9  const Navbar = () => {
10
11     const dispatch = useDispatch();
12     const navigate = useNavigate();
13
14     const isAuthenticated = useSelector((state) => state.user.isAuthenticated);
15     const user = useSelector((state) => state.user.userData);
16
17     const handleLogout = () => {
18         dispatch(removeUserData());
19         navigate("/");
20     }
21
22     return (
23         <header className="flex h-[80px] items-center justify-center shadow-md">
24             <div className="mx-5 flex w-full max-w-[1550px] items-center justify-between">
25                 {/* image section */}
26                 <div className="flex h-[60px] w-[120px] items-center justify-center overflow-hidden">
27                     
28                 </div>
29                 {/* nav links */}
30                 <GiHamburgerMenu className="text-xl md:hidden" />
31                 <div className="hidden md:flex md:items-center md:justify-center md:gap-4">
32                     <Link to="/">
33                         Home
34                     </Link>
35                     <Link to="/about">
36                         About
37                     </Link>
38                 </div>
39                 {/* Conditional Rendering */}
40                 {isAuthenticated ? (
41

```

```

39                 {/* Conditional Rendering */}
40                 {isAuthenticated ? (
41                     <>
42                         <Link to="/search">
43                             <FaSearch className="text-xl" />
44                         </Link>
45                         <Link to="/upload">
46                             <MdOutlineFileUpload className="text-[24px]" />
47                         </Link>
48                         <Link to="/profile">
49                             <button className="rounded-xl bg-blue-500 px-5 py-2 font-semibold hover:bg-blue-600">
50                                 Profile
51                             </button>
52                         </Link>
53                         <button className="rounded-xl bg-blue-500 px-5 py-2 font-semibold hover:bg-blue-600" onClick={handleLogout}>
54                             Logout
55                         </button>
56                     </>
57                 ) : (
58                     <>
59                         <Link to="/login">
60                             <button className="rounded-xl bg-blue-500 px-5 py-2 font-semibold hover:bg-blue-600">
61                                 Login
62                             </button>
63                         </Link>
64                         <Link to="/signup">
65                             <button className="rounded-xl bg-blue-500 px-5 py-2 font-semibold hover:bg-blue-600">
66                                 Signup
67                             </button>
68                         </Link>
69                     </>
70                 )
71             }
72         </div>
73     </div>
74 </header>
75 );
76
77 export default Navbar;

```

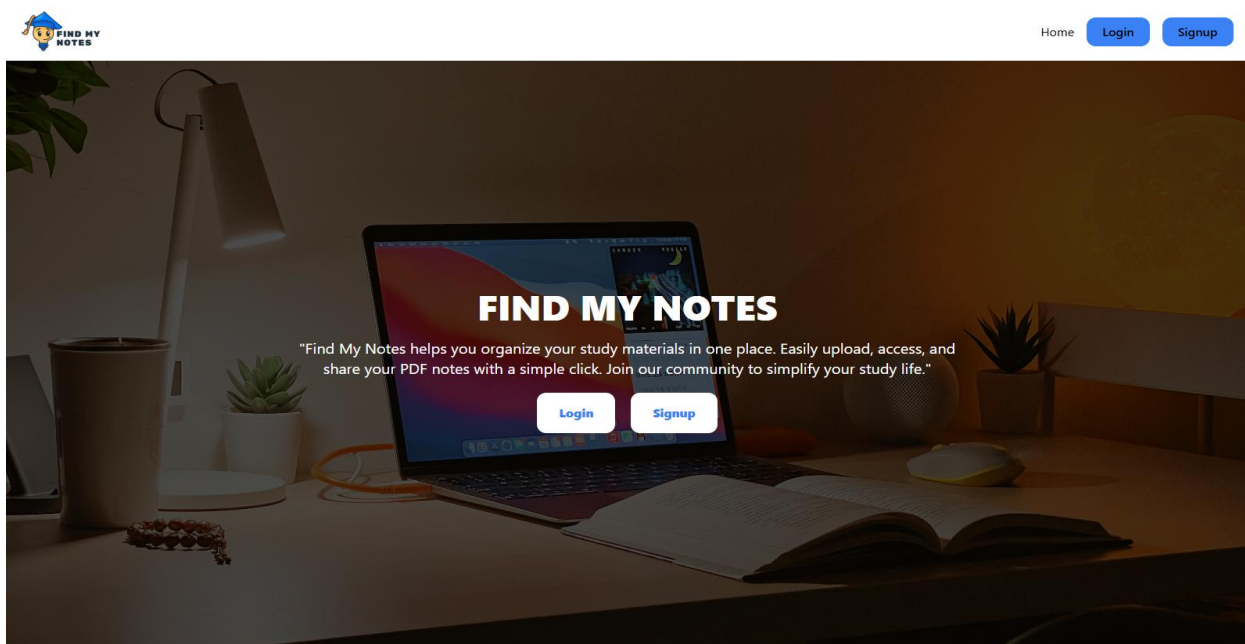
Footer.jsx

```
client > src > components > Footer.jsx > ...
1 import React from "react";
2 import { Link } from "react-router-dom";
3
4 const Footer = () => {
5   return (
6     <footer className="flex items-center justify-center p-16">
7       <div className="flex h-full w-full flex-col gap-10 px-20 lg:flex-row lg:justify-between">
8         <div className="lg:w-[450px]">
9           <h2 className="relative mb-3 text-2xl font-black before:absolute before:top-[30px] before:h-[3px] before:w-[50px] before:bg-blue-500">
10             About Us
11           </h2>
12           <p className="text-gray-600">
13             Because your planning is not always perfect, you need to be able to
14             study whenever, wherever. Just read your notes one last time on your
15             tablet or phone while you're on the go.
16           </p>
17         </div>
18         /* Quick Links section removed */
19         <div className="">
20           <h2 className="relative mb-3 text-2xl font-black before:absolute before:top-[30px] before:h-[3px] before:w-[50px] before:bg-blue-500">
21             Contact Info
22           </h2>
23           <ul className="text-gray-600">
24             <li className="mb-1">
25               <p>+91 12345 67890</p>
26             </li>
27             <li className="mb-1">
28               <p>+91 12345 67890</p>
29             </li>
30             <li className="mb-1">
31               <p>findmynotes2022@gmail.com</p>
32             </li>
33           </ul>
34         </div>
35       </div>
36     </footer>
37   );
38 };
39
40 export default Footer;
41
```

Home.jsx

```
client > src > pages > Home.jsx > [0] default
1 import React from "react";
2 import Footer from "../components/footer";
3 import { useSelector } from "react-redux";
4 import { Link } from "react-router-dom";
5
6 const Home = () => {
7   const isAuthenticated = useSelector((state) => state.user.isAuthenticated);
8
9   return (
10     <div className="h-heightWithoutNavbar">
11       { /* Hero component content moved here */ }
12       <div className="bg-unsplashBgImage relative flex h-full items-center justify-center bg-cover bg-center">
13         <div className="absolute inset-0 bg-black bg-opacity-70" />
14         <div className="relative z-10 w-full max-w-[860px] text-center text-white">
15           <h1 className="text-4xl font-black md:text-5xl">FIND MY NOTES</h1>
16           <p className="mt-5 text-sm font-light md:text-xl md:font-normal">
17             "Find My Notes helps you organize your study materials in one place. Easily upload, access, and share your PDF notes with a simple click. Join our community to simplify your study life."
18           </p>
19           <div className="mt-5">
20             <div className="flex items-center justify-center gap-5">
21               {isAuthenticated ? (
22                 <Link to="/search" className="mr-10 rounded-xl bg-white px-6 py-3 text-lg font-bold text-blue-500 hover:bg-gray-100">Get Started</Link>
23               ) : (
24                 <div>
25                   <Link to="/login">
26                     <button className="rounded-xl bg-white px-7 py-4 font-black text-blue-500">
27                       Login
28                     </button>
29                   </Link>
30                   <Link to="/signup">
31                     <button className="rounded-xl bg-white px-7 py-4 font-black text-blue-500">
32                       Signup
33                     </button>
34                   </Link>
35                 </div>
36               )
37             </div>
38           </div>
39         </div>
40       </div>
41       { /* End of Hero component content */ }
42       <Footer />
43     </div>
44   );
45 };
46
47 export default Home;
```

OUTPUT:



5.3 Conclusion:

We successfully implemented functional and class components in React to build reusable UI blocks including navigation, lists, events, and forms. The experiment demonstrates the importance of reusable components in building responsive and dynamic React applications.

5.4 Questions:

1. What are the key differences between functional components and class components in React?

ANS: Functional Components: Simple functions, use hooks for state and lifecycle, easier to write. Class Components: Use ES6 classes, manage state with `this.state`, lifecycle methods.

2. Define reusable components and explain their importance in building dynamic applications.

ANS: Reusable components are independent UI blocks that can be used multiple times with different props. They promote code reusability, reduce duplication, and make apps more scalable and maintainable.

3. What is React Router and how would you integrate it with a navigation component?

ANS: React Router is a library for routing in React apps. Integration: Wrap the app in `<BrowserRouter>`, use `<Link>` instead of `<a>` for navigation, and define routes using `<Route>`