Govt. College,Solan

BCA VIth Semesters

List of Project Titles

Session 2024-25

Sr. No	Roll No.	Name	Project Title	Software Used
1	2201	Pankaj	Uber Website (Food Delivery)	HTML,CSS,JavaScript
2	2202	Aditya	Real Estate Property sale Rent Application	HTML,CSS,JavaScript
3	2203	Mansi	Portfolio website	HTML,CSS
4	2204	Saksham	Uber Website (Food Delivery)	HTML,CSS,JavaScript
5	2205	Preeti	Weather Website	HTML,CSS,JavaScript
6	2207	Simran	E commerce website (Fashion World)	HTML,CSS
7	2208	Divya	E commerce website (Fashion World)	HTML,CSS
8	2209	Ruchika	Online Voting System	HTML,PHP,CSS
9	2210	Varun	Uber Website (Food Delivery)	HTML,CSS,JavaScript
10	2211	Isha	Travel Website	HTML,CSS
11	2212	Muskan	Student Attendance System	HTML,CSS,JavaScript
12	✓2214	Kunal	E commerce website	HTML,JAVA script,css
13	2215	Arun	Stock Vision	HTML,CSS,Chart.JS
14	2216	Manasvi	Crime Reporting and monitoring system	HTML,CSS
15	2218	Preeti	Crime Reporting and monitoring system	HTML,CSS
16	2220	Sahil	Uber Website (Food Delivery)	HTML,CSS,JavaScript
17	✓2221	Mayur	E commerce website	HTML,JAVA script,css
18	2222	Abhinav	Airline Reservation Website	HTML,CSS,JavaScript
19	2223	Deepak	Stock Vision	HTML,CSS,Chart.JS
20	2225	Ram Kumar	Online Voting System	HTML,PHP,CSS
21	2226	Aakriti	Producitvity Hub	HTML,CSS,JavaScript
22	2227	Priyanka	Producitvity Hub	HTML,CSS,JavaScript
23	2228	Nidhi	Producitvity Hub	HTML,CSS,JavaScript
24	2229	Komal	Portfolio website	HTML,CSS
25	2230	Malika	Weather Website	HTML,CSS,JavaScript
26	2231	Sahil Kumar	Ludo Game	JavaScript,HTML
27	2232	Tushar	Airline Reservation Website	HTML,CSS,JavaScript
28	2233	Deepika	Crime Reporting and monitoring system	HTML,CSS
29	2234	Gurpreet	Airline Reservation Website	HTML,CSS,JavaScript

30	2235	Lavya	Ludo Game	JavaScript,HTML
31	2236	Kartik Sharma	Flappy Bird Clone	HTML,CSS,JavaScript
32	2237	Vansh	AUDIONIX	HTML,CSS
33	2238	Kunal Bhaskar	Ludo Game	JavaScript,HTML
34	2239	Akash	Airline Reservation Website	HTML,CSS,JavaScript
35	2240	Shruti	Online Shopping Platform	HTML,CSS
36	2241	Suman	Skincare website	Xampp,Apache,Mysql
37	2243	Shiva	Online Shopping System	HTML,CSS
38	2244	Varsha	Online Shopping System	HTML,CSS
39	2245	Diya	Skincare website	Xampp,Apache,Mysql
40	2246	Suhani	Online Shopping Platform	HTML,CSS
41	2247	Muskan	Online Shopping System	HTML,CSS
42	2248	Sarthak	Real Estate Property sale Rent Application	HTML,CSS,JavaScript
43	2249	Sneha	Travel Website	HTML,CSS
44	2250	Uma	Portfolio website	HTML,CSS
45	2251	Pratik	AUDIONIX	HTML,CSS
46	2252	Shreya Thakur	Travel Website	HTML,CSS
47	2253	Sachin	Flappy Bird Clone	HTML,CSS,JavaScript
48	2254	Kunal Bharti	Real Estate Property sale Rent Application	HTML,CSS,JavaScript
49	2255	Gargi	Crime Reporting and monitoring system	HTML,CSS
50	2256	Kartik	Flappy Bird Clone	HTML,CSS,JavaScript
51	2257	Vaibhav	Flappy Bird Clone	HTML,CSS,JavaScript
52	2259	Sahitya	Online Voting System	HTML,PHP,CSS
53	2260	Manish	Real Estate Property sale Rent Application	HTML,CSS,JavaScript
54	2261	Bhavika	Portfolio website	HTML,CSS
55	2262	Jatin	E-commerce (Airbuds)	HTML,JAVA script,css
56	2263	Priya	Skincare website	Xampp,Apache,Mysql

Govt. Degree College, Solan

PROJECT REPORT ON

E-COMMERCE WEBSITE

Submitted to:

Mrs. Simmi Sahni

Submitted By:

Kunal and Mayur
2214, 2221

CERTIFICATE
OF Completion

Govt. Degree College, Solan

This is to certify that the project report titled “Ecommerce Website” submitted by [Kunal and Mayur] [Roll No: 2214 and 2221] of BCA department during the academic year [2024-25] is an authentic work carried out under my supervision and guidance.

The project report is submitted in partial fulfillment of the requirements .

This work presented in this project report has not submitted elsewhere for the award of any other degree or diploma

Simmi

Mrs. Simmi Sahni
Project Guide
Department of
BCA
Date:

Principal
Govt College Solan
Date:

Signature of Co-ordinator

Acknowledgment

We would like to express our sincere gratitude to all those who contributed to the successful completion of this project.

Firstly we would like to thank our college for providing us the opportunity to undertake this project. A special thanks to our guide (Mrs. Simmi mam) for their valuable support, guidance and advice throughout the project.

This project has been a wonderful learning experience, and I sincerely thanks everyone who helped us along the way.

Name: Mayur and Kunal
Roll no: 2221 and 2214
Student of BCA department

Declaration

We [Mayur] and [Kunal] students of BCA department at Govt. Degree College, Solan hereby declare that the project titled "ecommerce website" submitted in partial fulfillment is our original work.

We confirm that this work has not been submitted elsewhere for any other degree or diploma. Any references, quotations, or borrowed ideas have been duty acknowledged in the report.

This project was carried out under the guidance of [Mrs. Simmi Sahni]

Date:

Signature
[Mayur, Roll No: 2221]: Mayur

Signature
[Kunal, Roll No: 2214]: Kunal

Project Report: Ecommerce Website(content)

Introduction:

The E-commerce Website project is designed to provide users with an efficient online shopping experience.
It offers features like product browsing, secure payments, and order management.
This report outlines the key aspects of the project, including its objectives, design, and implementation.

Objectives:

To create a user-friendly platform for online shopping.
To ensure secure and seamless payment processing.
To provide an efficient product management system for administrators.

System Testing:

The existing manual shopping systems are time-consuming and inconvenient.The proposed e-commerce website addresses these issues with automated product management, secure payment integration, and order tracking.

Development Environment:

Frontend: React.js, HTML, CSS, JavaScript
Backend: Node.js, Express.js
Database: MongoDB

Implementation:

User Module: Users can register, log in, browse products, and manage orders.
Admin Module: Admins can manage products, view sales reports, and oversee orders.
Product Management: Products are organized into categories with detailed descriptions and images.
Shopping Cart and Checkout: Users can add products to the cart, review orders, and make secure payments.

Results and Discussion:

The website provides an intuitive user interface, secure payment gateway integration, and efficient product management. Performance analysis shows minimal latency, ensuring a smooth user experience.

References:

MongoDB Documentation
Node.js and Express.js Guides
React.js Official Documentation

Working Description of the Project

This project is an **E-commerce Website for Clothing**, designed t provide users with a seamless online shopping experience. The platform allows customers to browse a wide range of apparel, view product details, add items to their cart, and complete purchases securely.

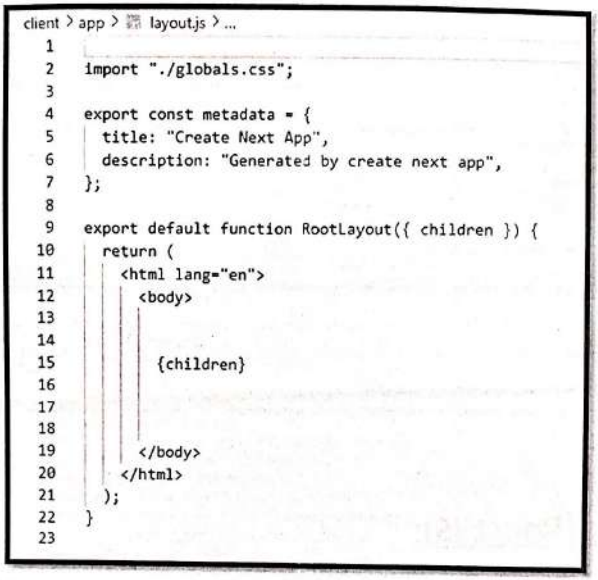
The website features **user authentication**, enabling customers to create accounts, manage their orders, and track deliveries. It also includes an **admin panel** for store owners to manage inventory, update product details, and monitor sales.

Developed using **HTML, CSS, JavaScript, and backend technologies**, the website ensures a **responsive design** for mobile and desktop users. Additionally, secure payment gateways and order management systems are integrated to enhance user convenience and security.

This project aims to bridge the gap between retailers and consumers by offering a **user-friendly, visually appealing, and functional online clothing store**.

Coding

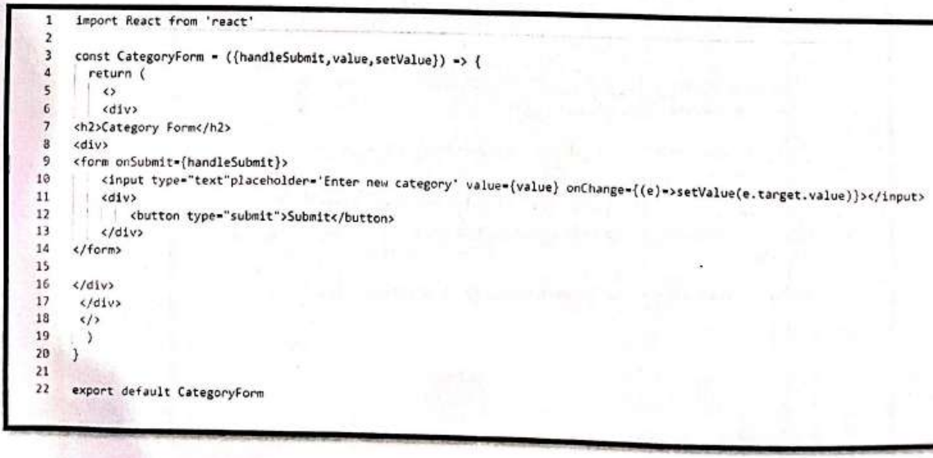
Root Layout (React JS):



client > app > layout.js > ...

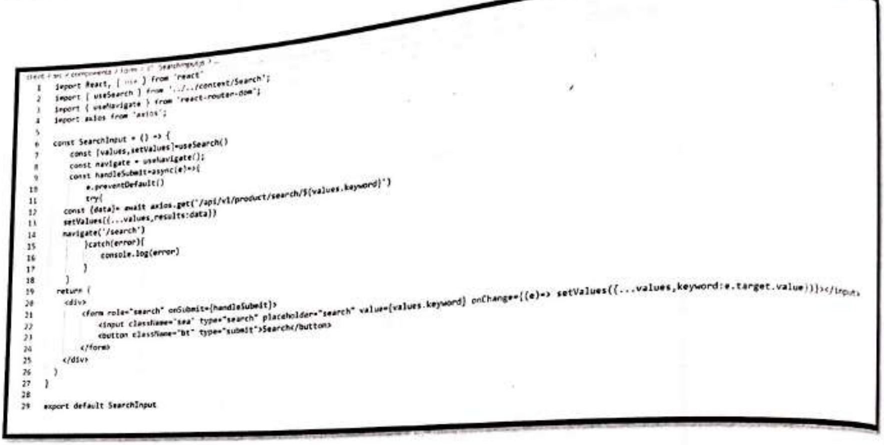
```
import "./globals.css";

export const metadata = {
  title: "Create Next App",
  description: "Generated by create next app",
};

export default function RootLayout({ children }) {
  return (
    <html lang="en">
      <body>


        {children}



      </body>
    </html>
  );
}
```

Category Form (React JS):



```
import React from 'react'

const CategoryForm = ({handleSubmit,value,setValue}) => {
  return (
    <>
    <div>
<h2>Category Form</h2>
<div>
<form onSubmit={handleSubmit}>
    <input type="text"placeholder='Enter new category' value={value} onChange={(e)=>setValue(e.target.value)}></input>
    <div>
        <button type="submit">Submit</button>
    </div>
</form>

</div>
 </div>
 </>
  )
}

export default CategoryForm
```

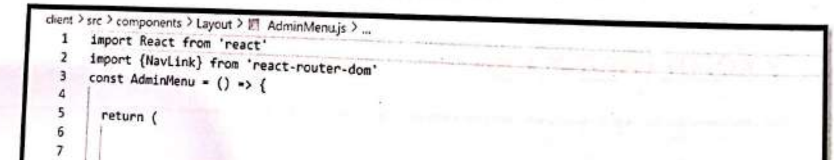
Search Input (React JS):



client > src > components > Form > SearchInput.js > ...

```
import React, { use } from 'react'
import { useSearch } from '../../context/Search';
import { useNavigate } from 'react-router-dom';
import axios from 'axios';

const SearchInput = () => {
    const [values,setValues]=useSearch()
    const navigate = useNavigate();
    const handleSubmit=async(e)=>{
        e.preventDefault()
        try{
  const {data}= await axios.get(`/api/v1/product/search/${values.keyword}`)
  setValues({...values,results:data})
  navigate('/search')
        }catch(error){
            console.log(error)
        }
    }
  return (
    <div>
        <form role="search" onSubmit={handleSubmit}>
            <input className='sea' type="search" placeholder="search" value={values.keyword} onChange={(e)=> setValues({...values,keyword:e.target.value})}></input>
            <button className="bt" type="submit">Search</button>
        </form>
    </div>
  )
}

export default SearchInput
```

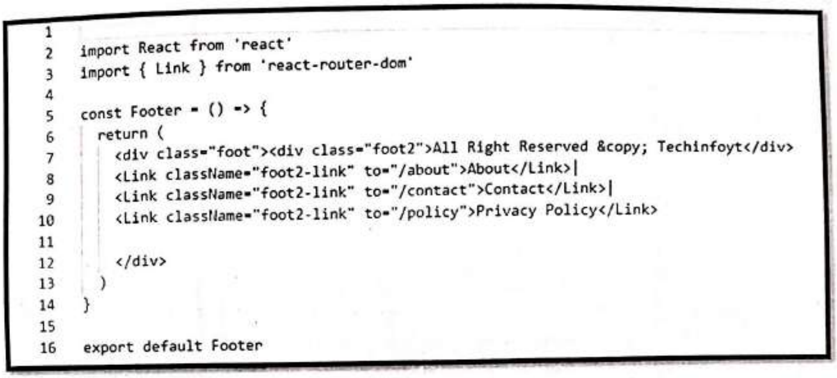
Admin Menu (React JS):



client > src > components > Layout > AdminMenu.js > ...

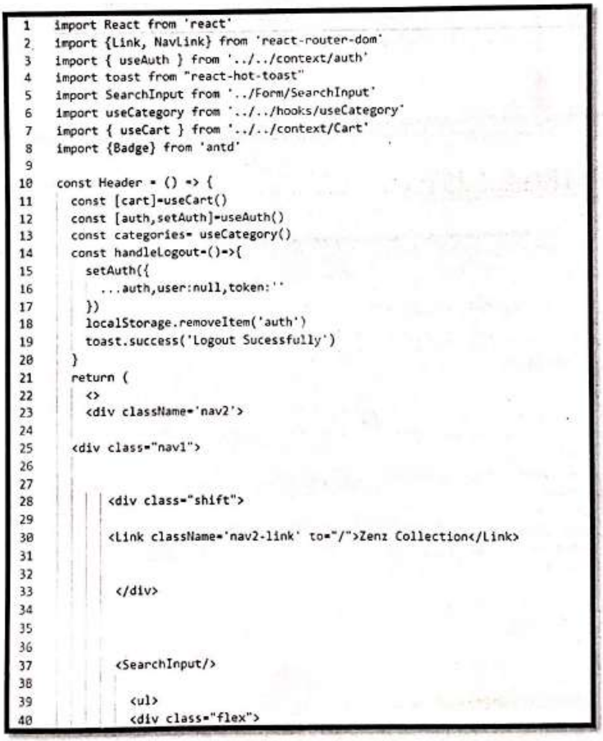
```
import React from 'react'
import {NavLink} from 'react-router-dom'
const AdminMenu = () => {

  return (


    <div className='list-group'>
        <h1>Admin Panel</h1>

<NavLink className="a" to="/dashboard/admin/create-category">Create Category</NavLink>
<br></br>
<NavLink className="b" to="/dashboard/admin/create-product">Create Product</NavLink>
<br></br>
<NavLink className="c" to="/dashboard/admin/products">products</NavLink>
<br></br>

<NavLink className="d" to="/dashboard/admin/users">Users</NavLink>


    </div>
  )
}

export default AdminMenu
```

Footer (React JS):



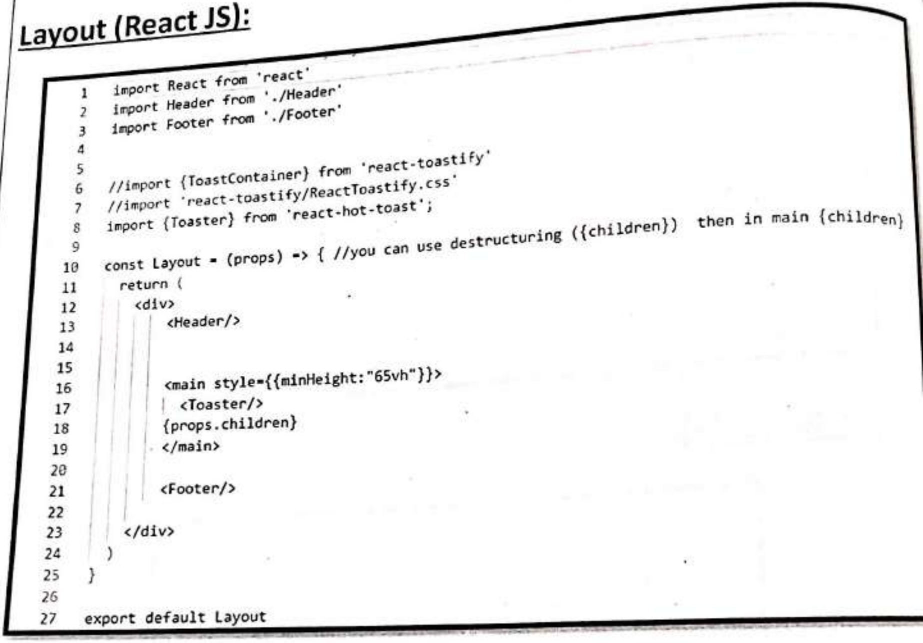
```
import React from 'react'
import { Link } from 'react-router-dom'

const Footer = () => {
  return (
    <div class="foot"><div class="foot2">All Right Reserved &copy; Techinfoyt</div>
    <Link className="foot2-link" to="/about">About</Link>|
    <Link className="foot2-link" to="/contact">Contact</Link>|
    <Link className="foot2-link" to="/policy">Privacy Policy</Link>

    </div>
  )
}

export default Footer
```

Header (React JS):



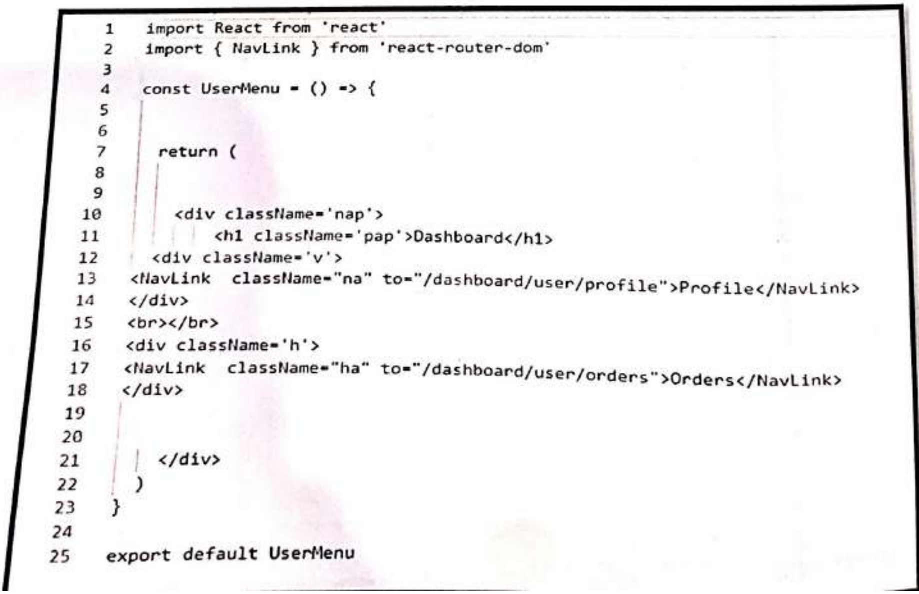
```
import React from 'react'
import {Link, NavLink} from 'react-router-dom'
import { useAuth } from '../../context/auth'
import toast from "react-hot-toast"
import SearchInput from '../Form/SearchInput'
import useCategory from '../../hooks/useCategory'
import { useCart } from '../../context/Cart'
import {Badge} from 'antd'

const Header = () => {
  const [cart]=useCart()
  const [auth,setAuth]=useAuth()
  const categories= useCategory()
  const handleLogout=()=>{
    setAuth({
      ...auth,user:null,token:''
    })
    localStorage.removeItem('auth')
    toast.success('Logout Sucessfully')
  }
  return (
    <>
    <div className='nav2'>

  <div class="nav1">


      <div class="shift">

      <Link className='nav2-link' to="/">Zenz Collection</Link>


       </div>



       <SearchInput/>

         <ul>
         <div class="flex">
```

Layout (React JS):



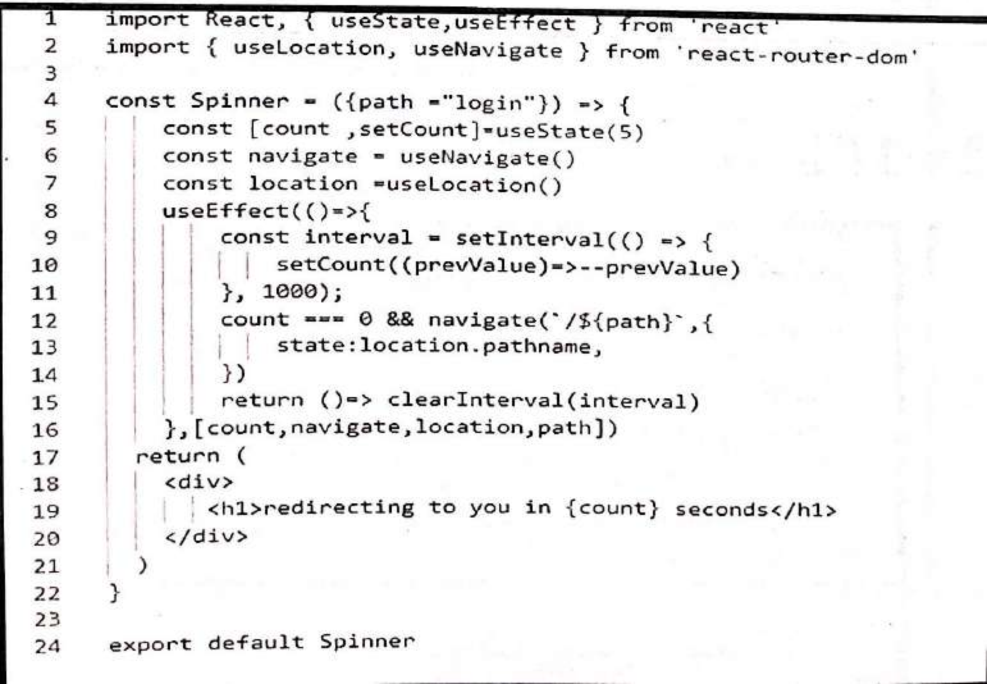
```
import React from 'react'
import Header from './Header'
import Footer from './Footer'


//import {ToastContainer} from 'react-toastify'
//import 'react-toastify/ReactToastify.css'
import {Toaster} from 'react-hot-toast';

const Layout = (props) => { //you can use destructuring ({children})  then in main {children}
  return (
    <div>
        <Header/>


        <main style={{minHeight:"65vh"}}>
          <Toaster/>
        {props.children}
        </main>

        <Footer/>

    </div>
  )
}

export default Layout
```

User Menu (React JS):



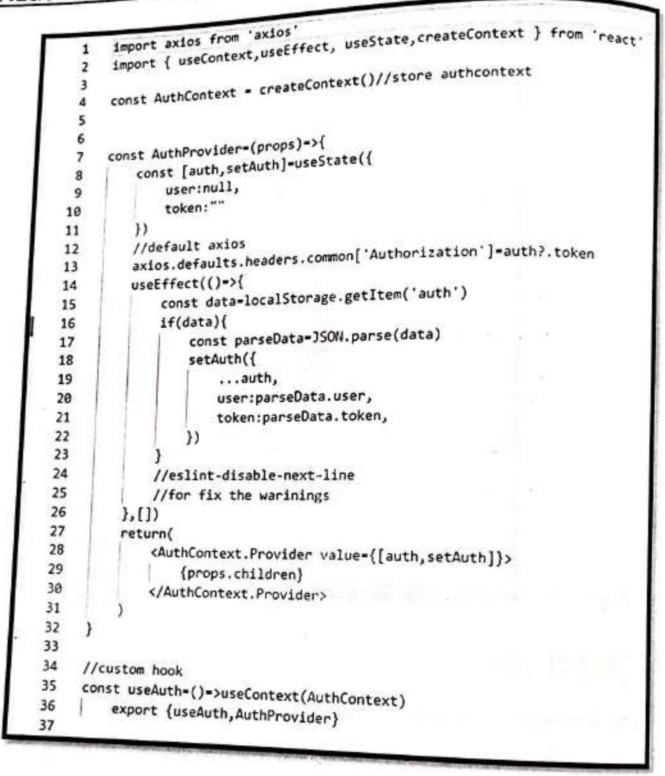
```
import React from 'react'
import { NavLink } from 'react-router-dom'

const UserMenu = () => {


  return (


    <div className='nap'>
        <h1 className='pap'>Dashboard</h1>
   <div className='v'>
 <NavLink  className="na" to="/dashboard/user/profile">Profile</NavLink>
 </div>
 <br></br>
 <div className='h'>
 <NavLink  className="ha" to="/dashboard/user/orders">Orders</NavLink>
 </div>


    </div>
  )
}

export default UserMenu
```

(React JS):



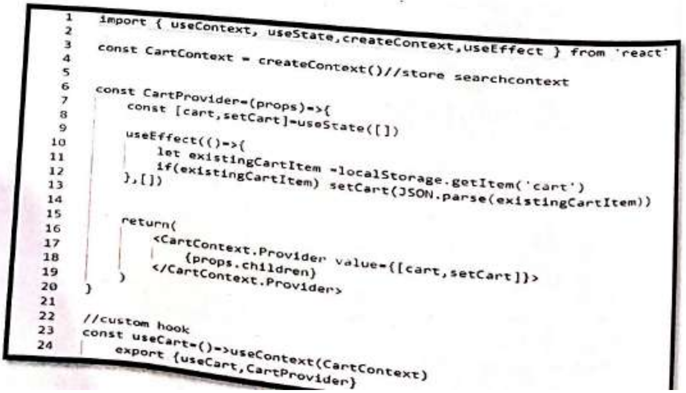
```
import React, { useState,useEffect } from 'react'
import { useLocation, useNavigate } from 'react-router-dom'

const Spinner = ({path ="login"}) => {
    const [count ,setCount]=useState(5)
    const navigate = useNavigate()
    const location =useLocation()
    useEffect(()=>{
        const interval = setInterval(() => {
            setCount((prevValue)=>--prevValue)
        }, 1000);
        count === 0 && navigate(`/${path}`,{
            state:location.pathname,
        })
        return ()=> clearInterval(interval)
    },[count,navigate,location,path])
  return (
    <div>
      <h1>redirecting to you in {count} seconds</h1>
    </div>
  )
}

export default Spinner
```

Authorization (React JS):



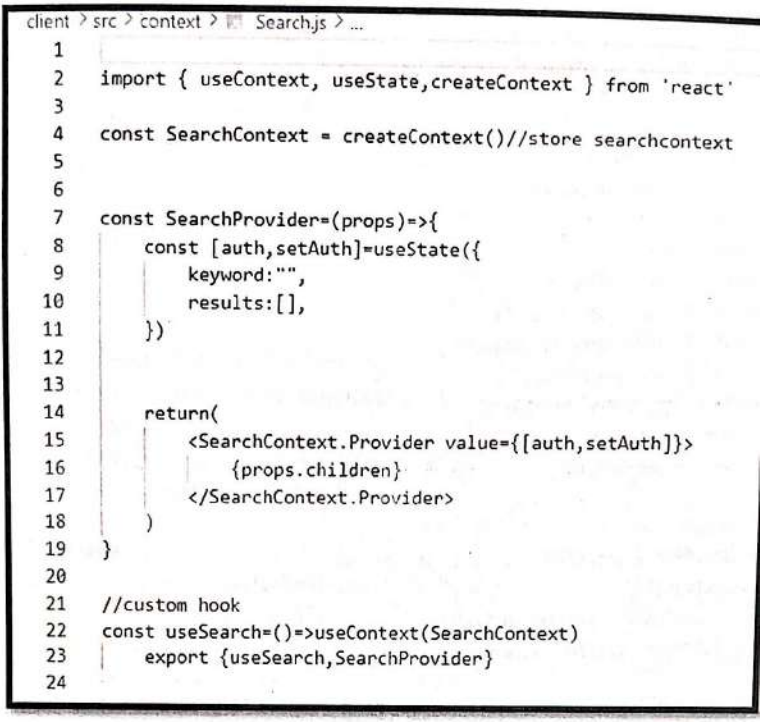
```
import axios from 'axios'
import { useContext,useEffect, useState,createContext } from 'react'

const AuthContext = createContext()//store authcontext


const AuthProvider=(props)=>{
    const [auth,setAuth]=useState({
        user:null,
        token:""
    })
    //default axios
    axios.defaults.headers.common['Authorization']=auth?.token
    useEffect(()=>{
        const data=localStorage.getItem('auth')
        if(data){
            const parseData=JSON.parse(data)
            setAuth({
                ...auth,
                user:parseData.user,
                token:parseData.token,
            })
        }
        //eslint-disable-next-line
        //for fix the warinings
    },[])
    return(
        <AuthContext.Provider value={[auth,setAuth]}>
            {props.children}
        </AuthContext.Provider>
    )
}

//custom hook
const useAuth=()=>useContext(AuthContext)
    export {useAuth,AuthProvider}
```

Cart (React JS):



```
import { useContext, useState,createContext,useEffect } from 'react'

const CartContext = createContext()//store searchcontext

const CartProvider=(props)=>{
    const [cart,setCart]=useState([])

    useEffect(()=>{
        let existingCartItem =localStorage.getItem('cart')
        if(existingCartItem) setCart(JSON.parse(existingCartItem))
    },[])

    return(
        <CartContext.Provider value={[cart,setCart]}>
            {props.children}
        </CartContext.Provider>
    )
}

//custom hook
const useCart=()=>useContext(CartContext)
    export {useCart,CartProvider}
```

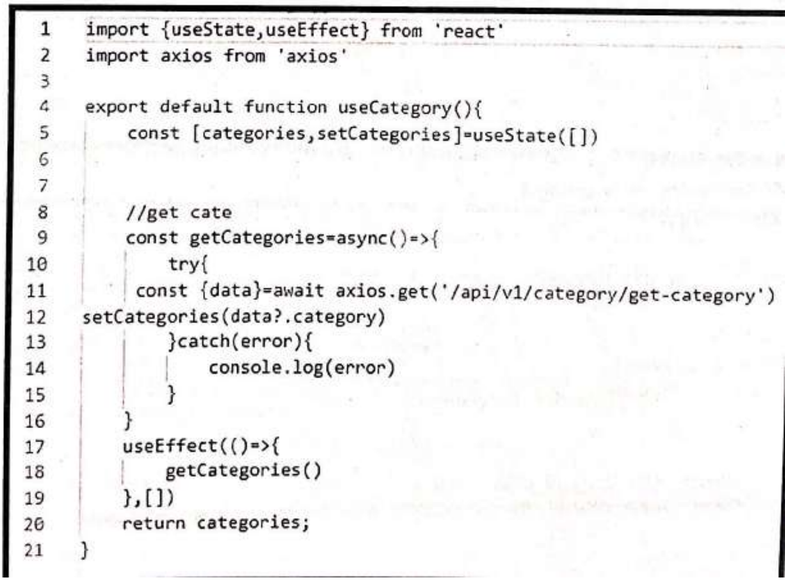
Search (React JS):



client > src > context > Search.js > ...

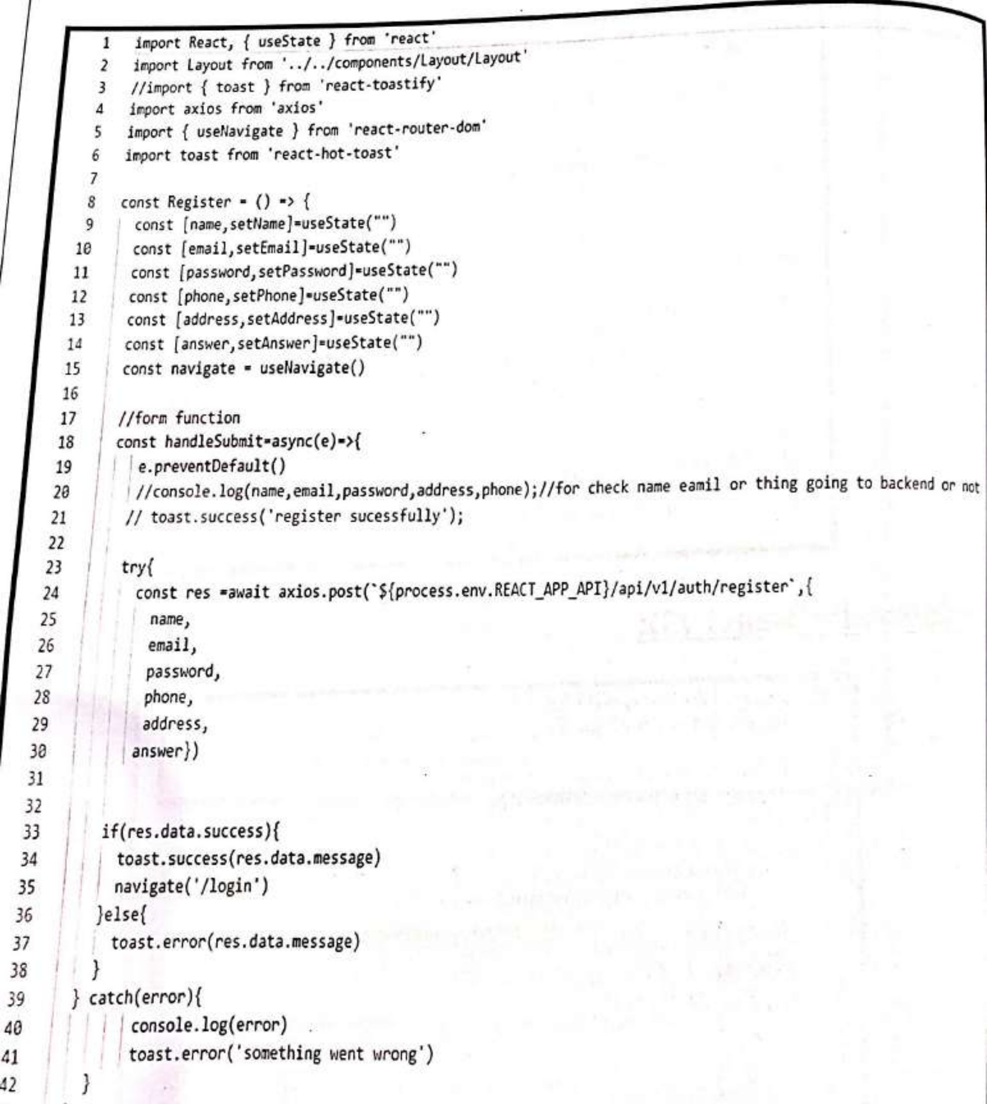
```

import { useContext, useState,createContext } from 'react'

const SearchContext = createContext()//store searchcontext


const SearchProvider=(props)=>{
    const [auth,setAuth]=useState({
        keyword:"",
        results:[],
    })


    return(
        <SearchContext.Provider value={[auth,setAuth]}>
            {props.children}
        </SearchContext.Provider>
    )
}

//custom hook
const useSearch=()=>useContext(SearchContext)
    export {useSearch,SearchProvider}

```

Use Category (React JS):



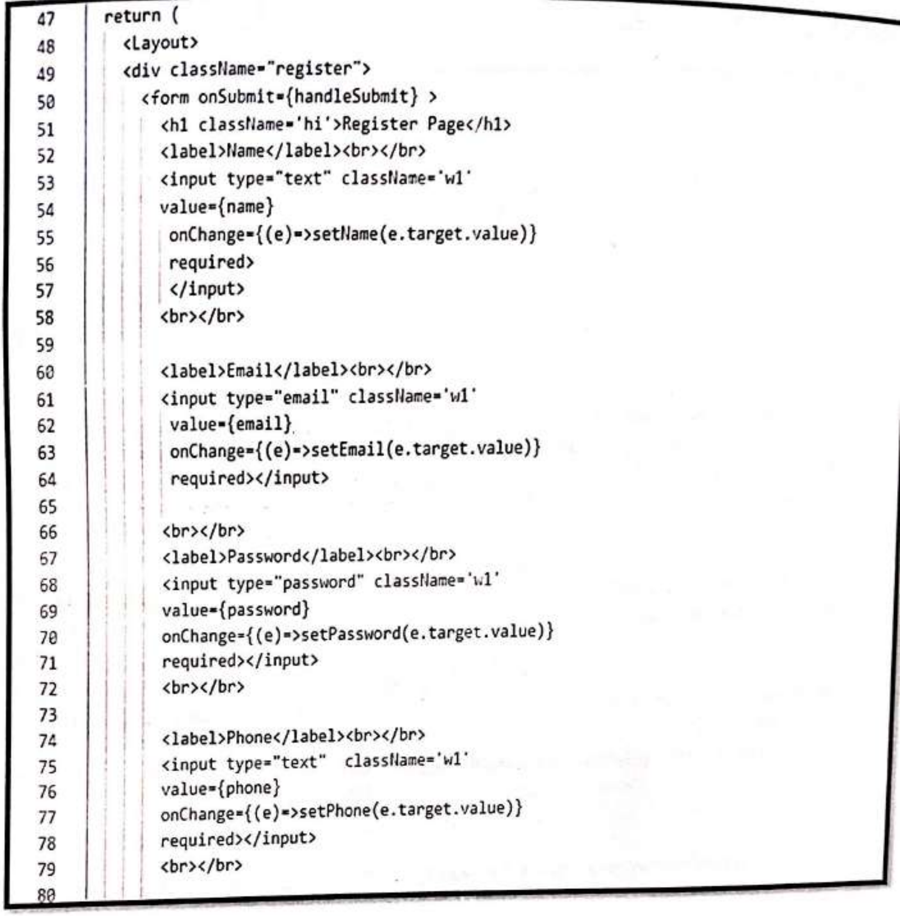
```
import {useState,useEffect} from 'react'
import axios from 'axios'

export default function useCategory(){
    const [categories,setCategories]=useState([])


    //get cate
    const getCategories=async()=>{
        try{
     const {data}=await axios.get('/api/v1/category/get-category')
setCategories(data?.category)
        }catch(error){
            console.log(error)
        }
    }
    useEffect(()=>{
        getCategories()
    },[])
    return categories;
}
```

Register Page (React JS):

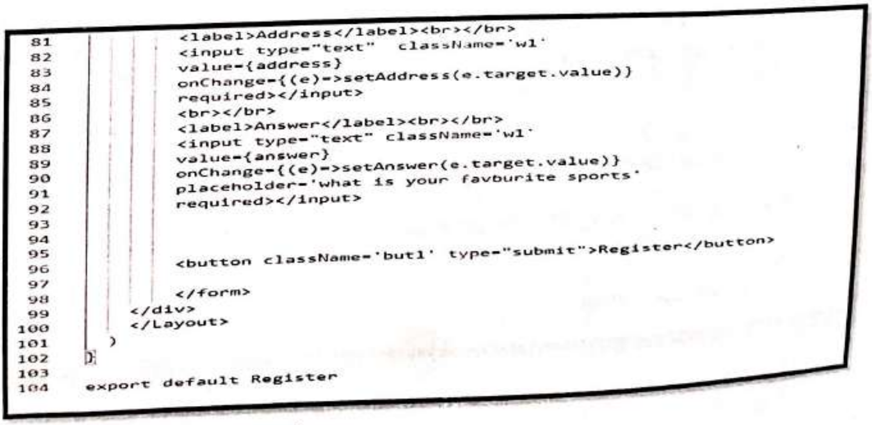


```
import React, { useState } from 'react'
import Layout from '../../components/Layout/Layout'
//import { toast } from 'react-toastify'
import axios from 'axios'
import { useNavigate } from 'react-router-dom'
import toast from 'react-hot-toast'

const Register = () => {
  const [name,setName]=useState("")
  const [email,setEmail]=useState("")
  const [password,setPassword]=useState("")
  const [phone,setPhone]=useState("")
  const [address,setAddress]=useState("")
  const [answer,setAnswer]=useState("")
  const navigate = useNavigate()

  //form function
  const handleSubmit=async(e)=>{
    e.preventDefault()
    //console.log(name,email,password,address,phone);//for check name eamil or thing going to backend or not
    // toast.success('register sucessfully');

    try{
      const res =await axios.post(`${process.env.REACT_APP_API}/api/v1/auth/register`,{
        name,
        email,
        password,
        phone,
        address,
        answer})


      if(res.data.success){
        toast.success(res.data.message)
        navigate('/login')
      }else{
        toast.error(res.data.message)
      }
    } catch(error){
        console.log(error)
        toast.error('something went wrong')
    }
  };
  console.log(process.env.REACT_APP_API)
```

Scanned with CamScanner

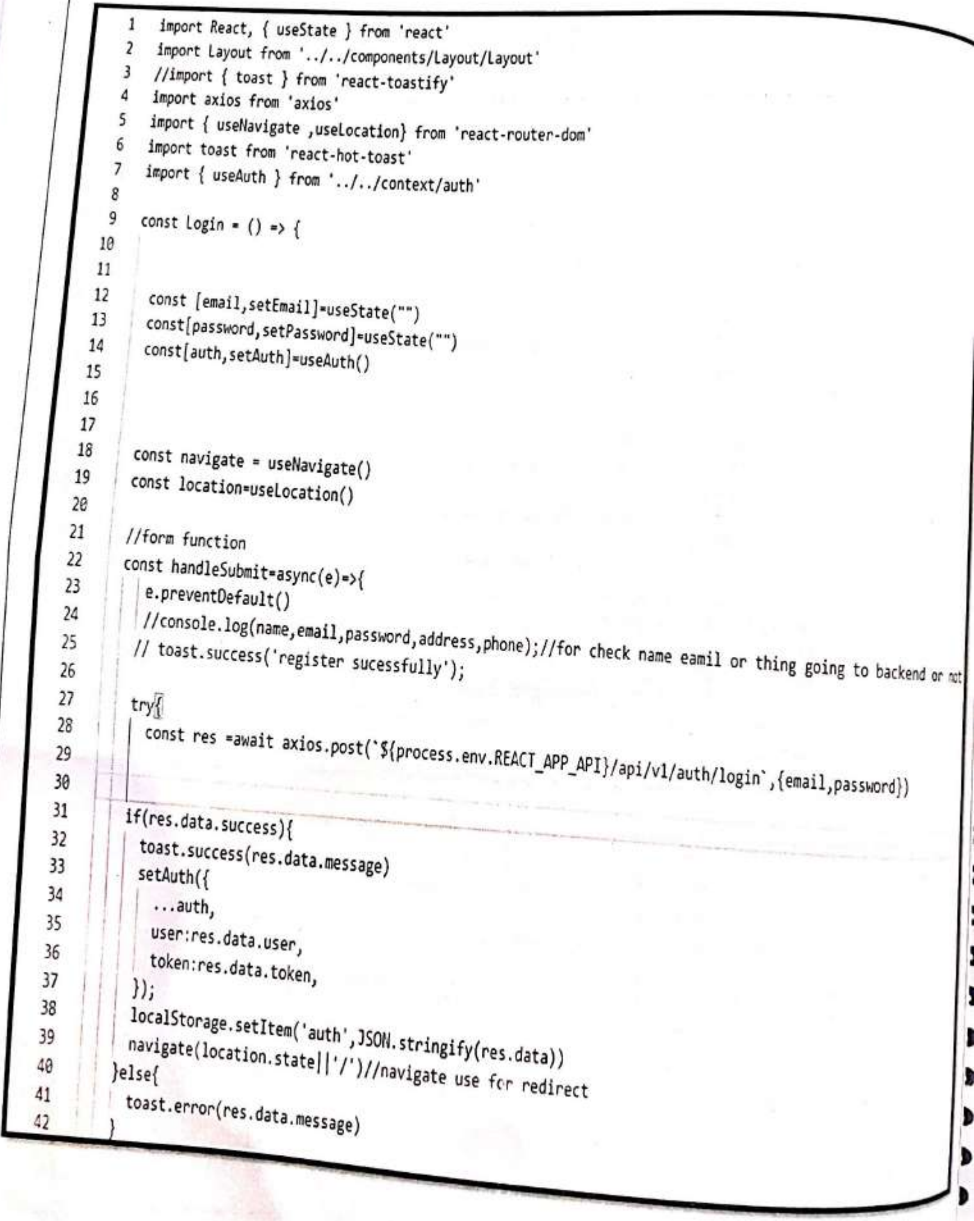


```
  return (
    <Layout>
    <div className="register">
      <form onSubmit={handleSubmit} >
        <h1 className='hi'>Register Page</h1>
        <label>Name</label><br></br>
        <input type="text" className='w1'
        value={name}
         onChange={(e)=>setName(e.target.value)}
         required>
         </input>
        <br></br>

        <label>Email</label><br></br>
        <input type="email" className='w1'
         value={email}
         onChange={(e)=>setEmail(e.target.value)}
         required></input>

        <br></br>
        <label>Password</label><br></br>
        <input type="password" className='w1'
        value={password}
        onChange={(e)=>setPassword(e.target.value)}
        required></input>
        <br></br>

        <label>Phone</label><br></br>
        <input type="text"  className='w1'
        value={phone}
        onChange={(e)=>setPhone(e.target.value)}
        required></input>
        <br></br>

```



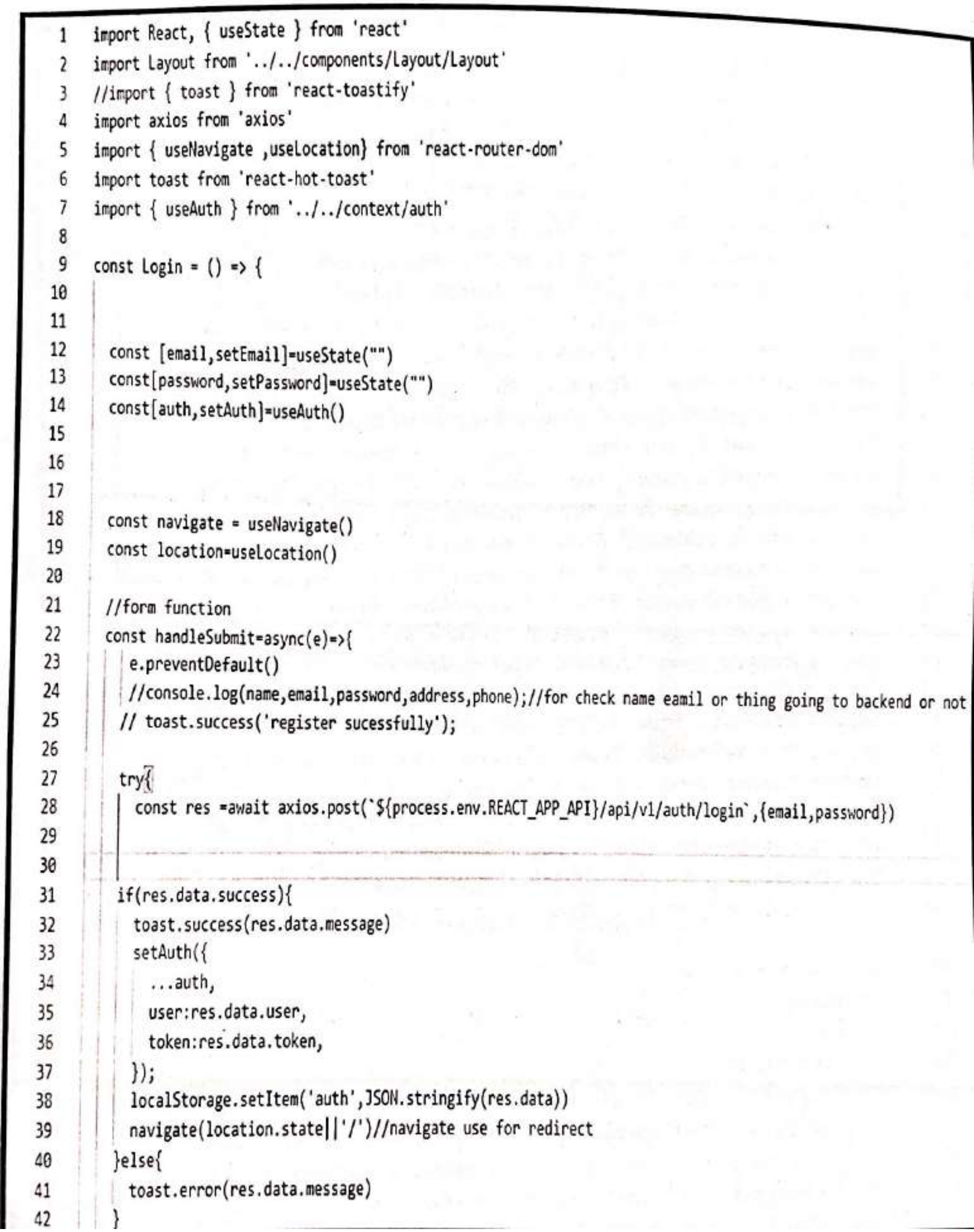
```
        <label>Address</label><br></br>
        <input type="text"  className='w1'
        value={address}
        onChange={(e)=>setAddress(e.target.value)}
        required></input>
        <br></br>
        <label>Answer</label><br></br>
        <input type="text" className='w1'
        value={answer}
        onChange={(e)=>setAnswer(e.target.value)}
        placeholder='what is your favburite sports'
        required></input>



        <button className='but1' type="submit">Register</button>

        </form>
    </div>
    </Layout>
  )
}

export default Register
```

Login Page (React JS):



```
import React, { useState } from 'react'
import Layout from '../../components/Layout/Layout'
//import { toast } from 'react-toastify'
import axios from 'axios'
import { useNavigate ,useLocation} from 'react-router-dom'
import toast from 'react-hot-toast'
import { useAuth } from '../../context/auth'

const Login = () => {


    const [email,setEmail]=useState("")
    const[password,setPassword]=useState("")
    const[auth,setAuth]=useAuth()



  const navigate = useNavigate()
  const location=useLocation()

 //form function
 const handleSubmit=async(e)=>{
   e.preventDefault()
   //console.log(name,email,password,address,phone);//for check name eamil or thing going to backend or not
  // toast.success('register sucessfully');

  try{
    const res =await axios.post(`${process.env.REACT_APP_API}/api/v1/auth/login`,{email,password})

 if(res.data.success){
   toast.success(res.data.message)
   setAuth({
     ...auth,
     user:res.data.user,
     token:res.data.token,
   });
   localStorage.setItem('auth',JSON.stringify(res.data))
   navigate(location.state||'/')//navigate use for redirect
 }else{
   toast.error(res.data.message)
 }
```

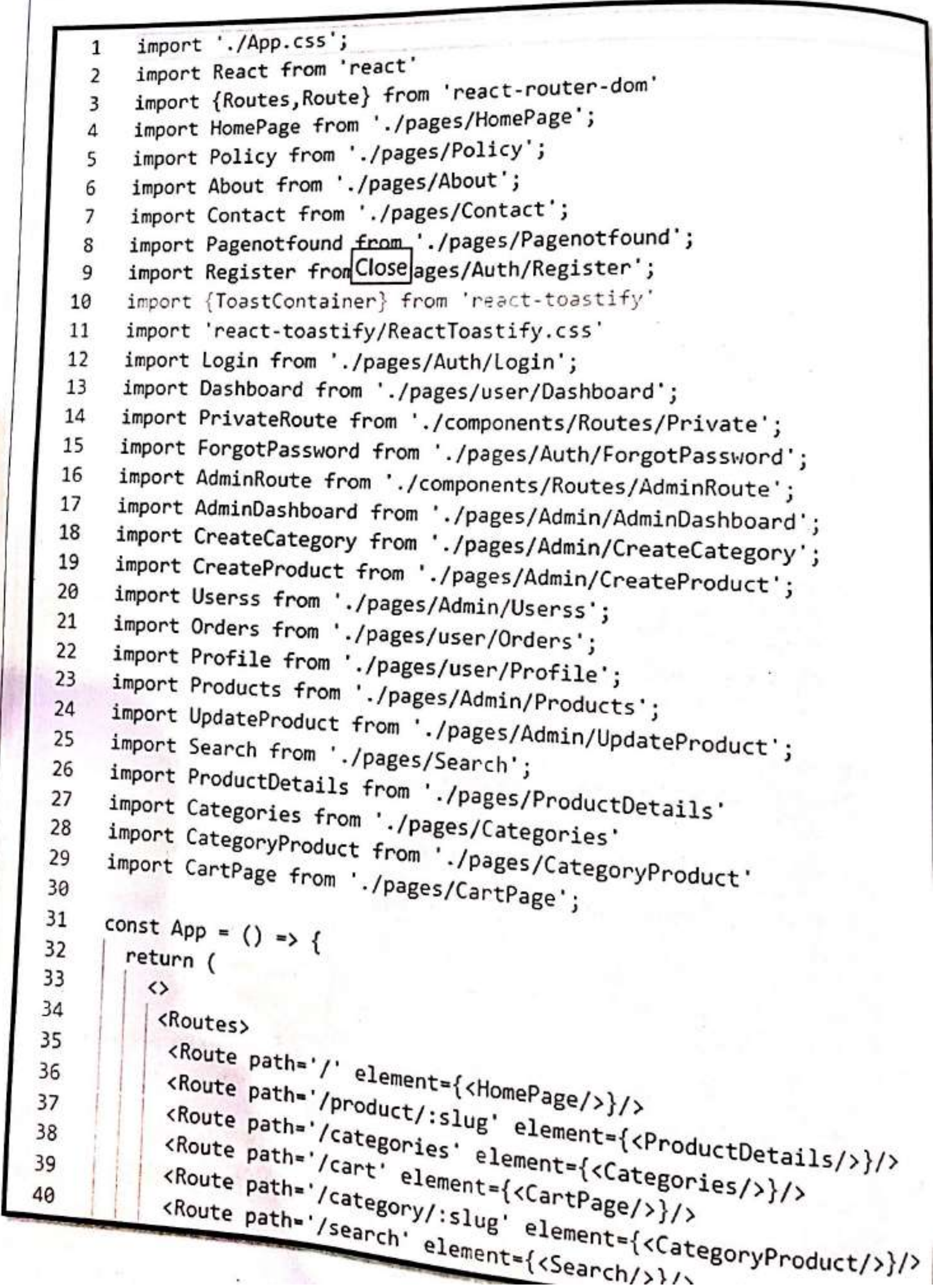
Login Page (React JS):



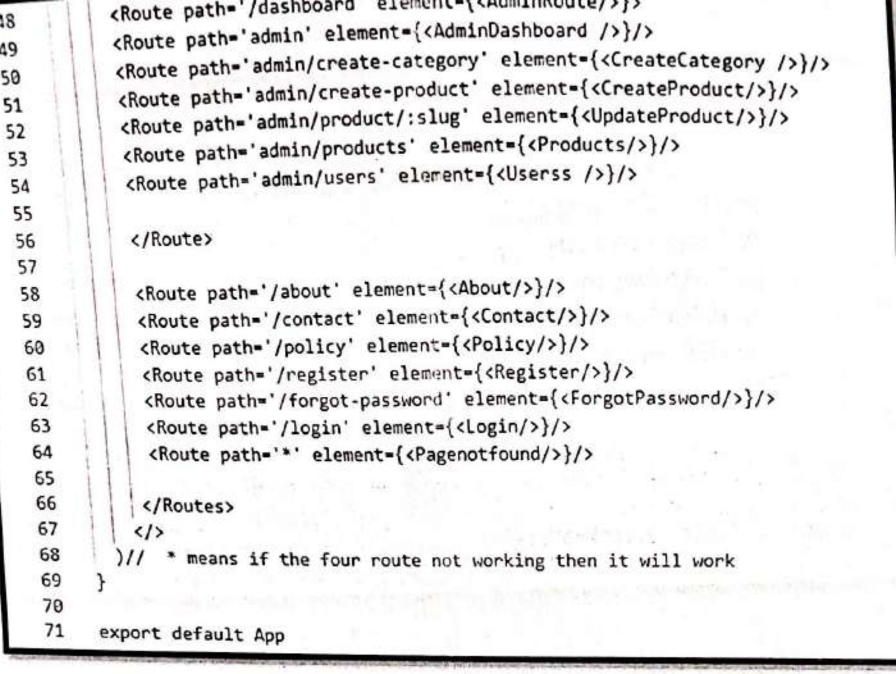
```
import React, { useState } from 'react'
import Layout from '../../components/Layout/Layout'
//import { toast } from 'react-toastify'
import axios from 'axios'
import { useNavigate ,useLocation} from 'react-router-dom'
import toast from 'react-hot-toast'
import { useAuth } from '../../context/auth'

const Login = () => {


  const [email,setEmail]=useState("")
  const[password,setPassword]=useState("")
  const[auth,setAuth]=useAuth()



  const navigate = useNavigate()
  const location=useLocation()

  //form function
  const handleSubmit=async(e)=>{
    e.preventDefault()
    //console.log(name,email,password,address,phone);//for check name eamil or thing going to backend or not
   // toast.success('register sucessfully');

   try{
    const res =await axios.post(`${process.env.REACT_APP_API}/api/v1/auth/login`,{email,password})


   if(res.data.success){
     toast.success(res.data.message)
     setAuth({
       ...auth,
       user:res.data.user,
       token:res.data.token,
     });
     localStorage.setItem('auth',JSON.stringify(res.data))
     navigate(location.state||'/')//navigate use for redirect
   }else{
     toast.error(res.data.message)
   }
```

Scanned with CamScanner

App (all routes) React JS:

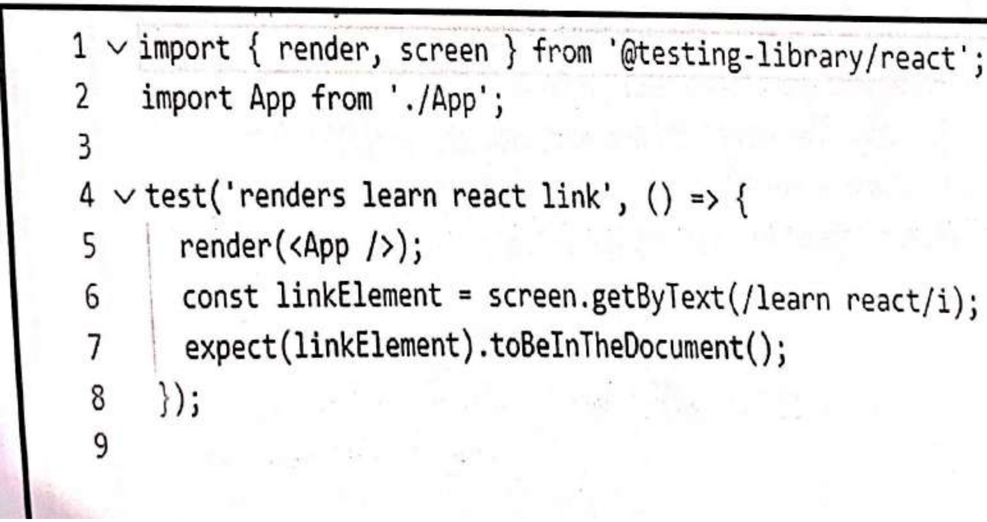


```
import './App.css';
import React from 'react'
import {Routes,Route} from 'react-router-dom'
import HomePage from './pages/HomePage';
import Policy from './pages/Policy';
import About from './pages/About';
import Contact from './pages/Contact';
import Pagenotfound from './pages/Pagenotfound';
import Register fron Close ages/Auth/Register';
import {ToastContainer} from 'react-toastify'
import 'react-toastify/ReactToastify.css'
import Login from './pages/Auth/Login';
import Dashboard from './pages/user/Dashboard';
import PrivateRoute from './components/Routes/Private';
import ForgotPassword from './pages/Auth/ForgotPassword';
import AdminRoute from './components/Routes/AdminRoute';
import AdminDashboard from './pages/Admin/AdminDashboard';
import CreateCategory from './pages/Admin/CreateCategory';
import CreateProduct from './pages/Admin/CreateProduct';
import Userss from './pages/Admin/Userss';
import Orders from './pages/user/Orders';
import Profile from './pages/user/Profile';
import Products from './pages/Admin/Products';
import UpdateProduct from './pages/Admin/UpdateProduct';
import Search from './pages/Search';
import ProductDetails from './pages/ProductDetails'
import Categories from './pages/Categories'
import CategoryProduct from './pages/CategoryProduct'
import CartPage from './pages/CartPage';

const App = () => {
  return (
    <>
     <Routes>
      <Route path='/' element={<HomePage/>}/>
      <Route path='/product/:slug' element={<ProductDetails/>}/>
      <Route path='/categories' element={<Categories/>}/>
      <Route path='/cart' element={<CartPage/>}/>
      <Route path='/category/:slug' element={<CategoryProduct/>}/>
      <Route path='/search' element={<Search/>}/>
```



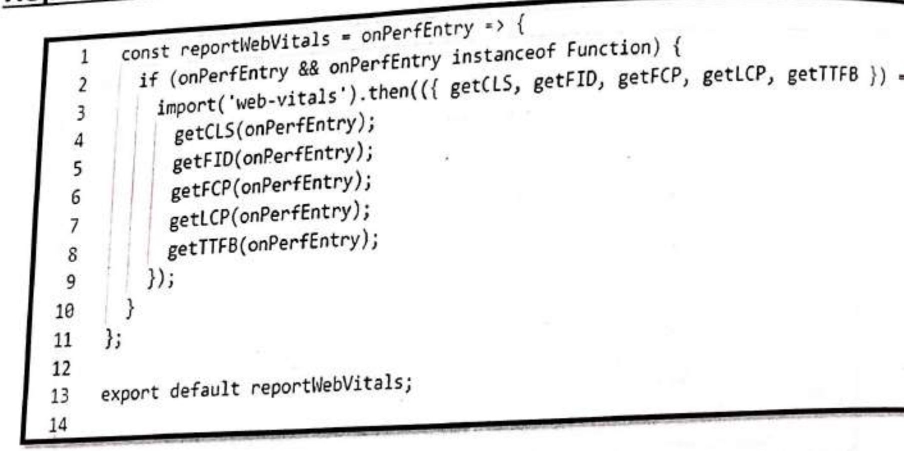
```
    <Route path='/dashboard' element={<AdminRoute/>}>
      <Route path='admin' element={<AdminDashboard />}/>
      <Route path='admin/create-category' element={<CreateCategory />}/>
      <Route path='admin/create-product' element={<CreateProduct/>}/>
      <Route path='admin/product/:slug' element={<UpdateProduct/>}/>
      <Route path='admin/products' element={<Products/>}/>
      <Route path='admin/users' element={<Userss />}/>

    </Route>

    <Route path='/about' element={<About/>}/>
    <Route path='/contact' element={<Contact/>}/>
    <Route path='/policy' element={<Policy/>}/>
    <Route path='/register' element={<Register/>}/>
    <Route path='/forgot-password' element={<ForgotPassword/>}/>
    <Route path='/login' element={<Login/>}/>
    <Route path='*' element={<Pagenotfound/>}/>

    </Routes>
    </>
  )//  * means if the four route not working then it will work
}

export default App
```

pp Test (React JS):



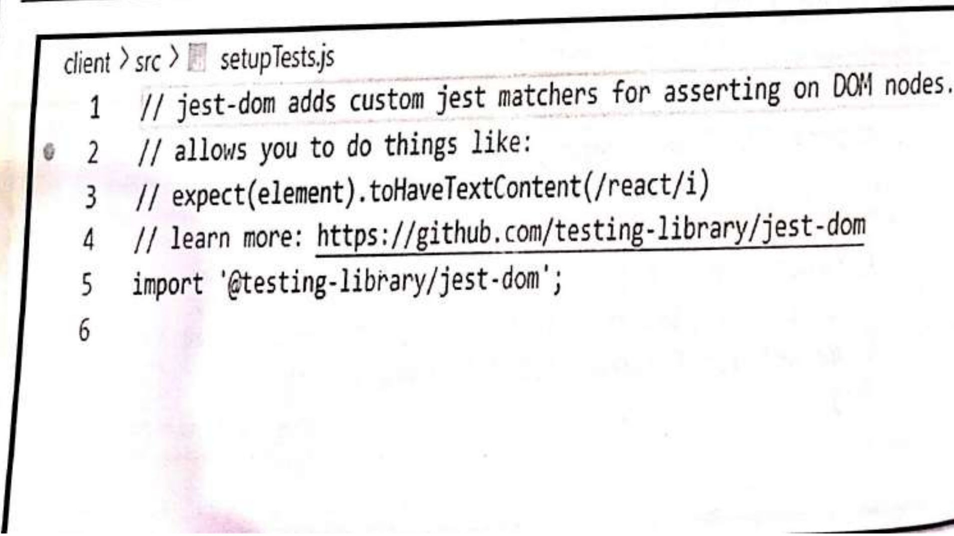
```
import { render, screen } from '@testing-library/react';
import App from './App';

test('renders learn react link', () => {
  render(<App />);
  const linkElement = screen.getByText(/learn react/i);
  expect(linkElement).toBeInTheDocument();
});

```

Report Web Titles (React JS):



```
const reportWebVitals = onPerfEntry => {
  if (onPerfEntry && onPerfEntry instanceof Function) {
    import('web-vitals').then(({ getCLS, getFID, getFCP, getLCP, getTTFB }) =
      getCLS(onPerfEntry);
      getFID(onPerfEntry);
      getFCP(onPerfEntry);
      getLCP(onPerfEntry);
      getTTFB(onPerfEntry);
    });
  }
};

export default reportWebVitals;
```

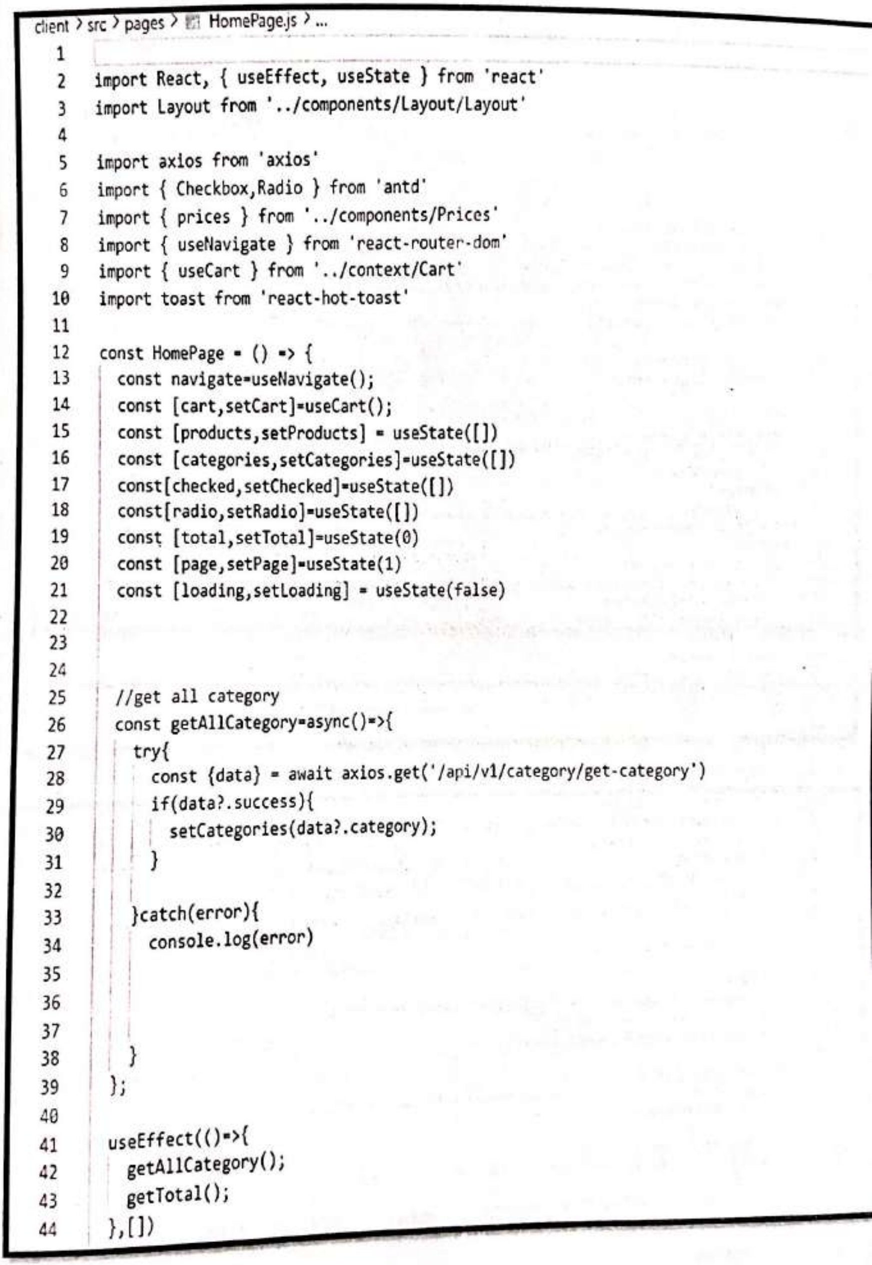
Setup Test (React JS):



client > src > setupTests.js

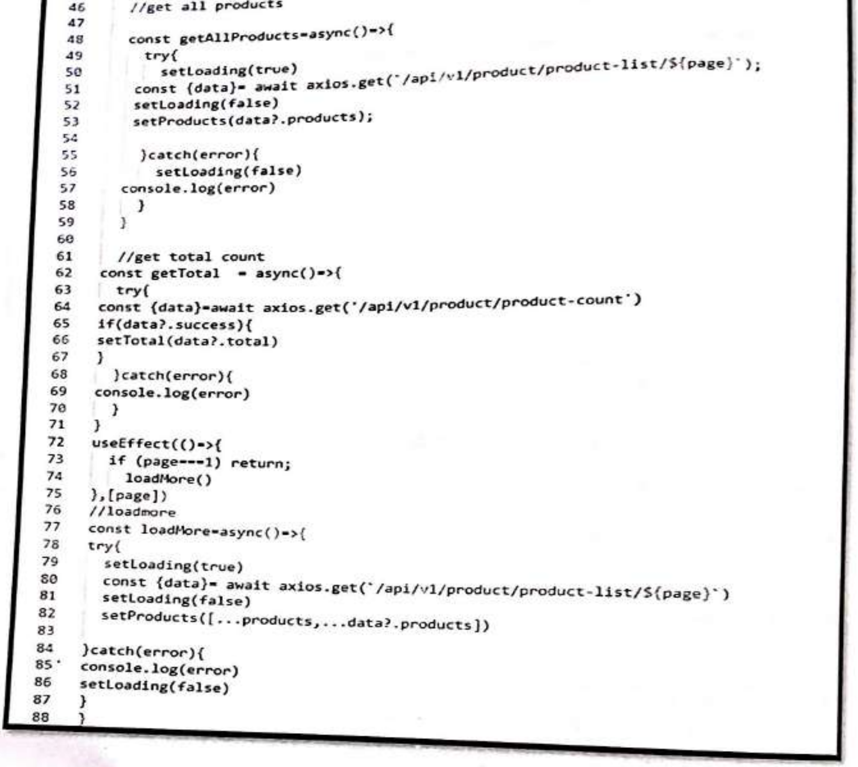
```
// jest-dom adds custom jest matchers for asserting on DOM nodes.
// allows you to do things like:
// expect(element).toHaveTextContent(/react/i)
// learn more: https://github.com/testing-library/jest-dom
import '@testing-library/jest-dom';
```

Home Page Code (React JS):

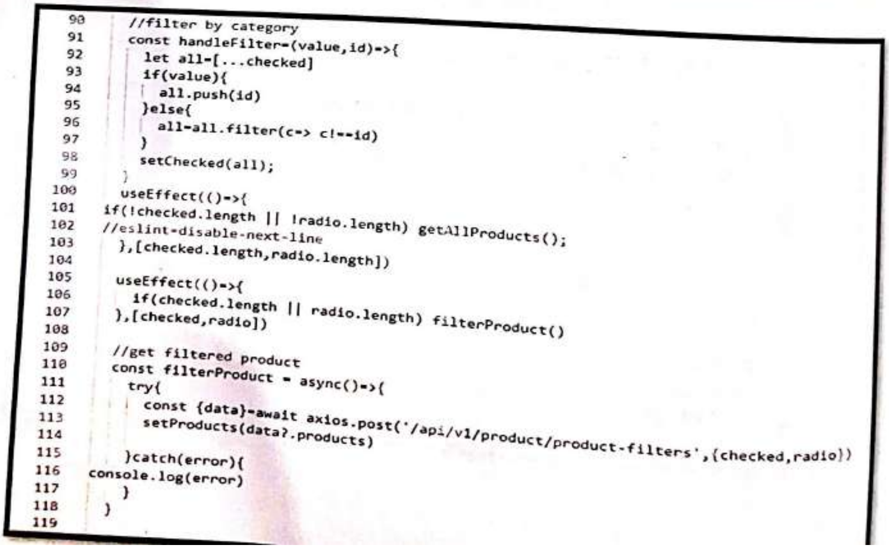


client > src > pages > HomePage.js > ...

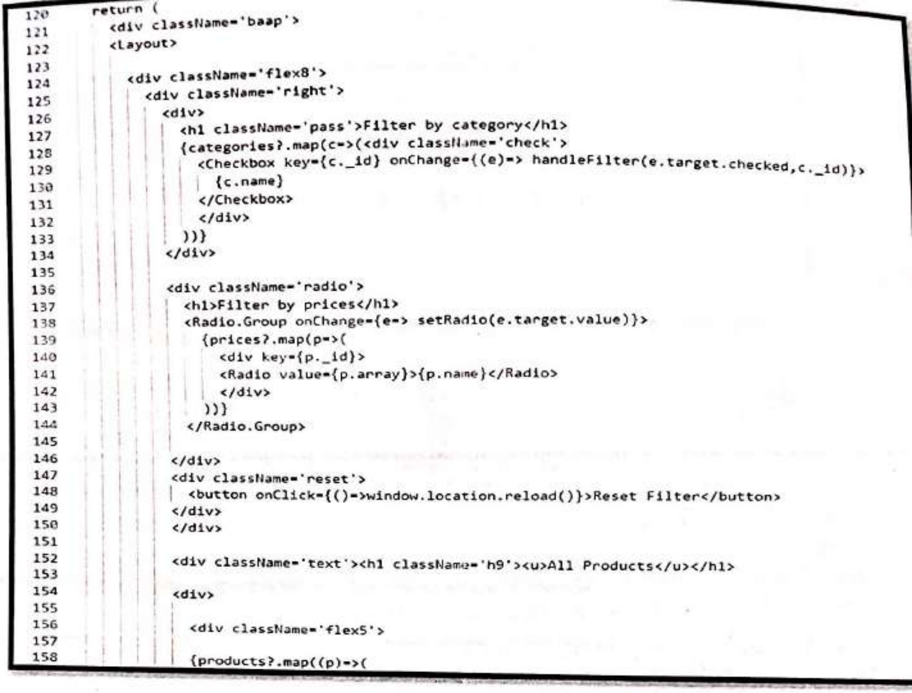
```
import React, { useEffect, useState } from 'react'
import Layout from '../components/Layout/Layout'

import axios from 'axios'
import { Checkbox,Radio } from 'antd'
import { prices } from '../components/Prices'
import { useNavigate } from 'react-router-dom'
import { useCart } from '../context/Cart'
import toast from 'react-hot-toast'

const HomePage = () => {
  const navigate=useNavigate();
  const [cart,setCart]=useCart();
  const [products,setProducts] = useState([])
  const [categories,setCategories]=useState([])
  const[checked,setChecked]=useState([])
  const[radio,setRadio]=useState([])
  const [total,setTotal]=useState(0)
  const [page,setPage]=useState(1)
  const [loading,setLoading] = useState(false)



  //get all category
  const getAllCategory=async()=>{
    try{
      const {data} = await axios.get('/api/v1/category/get-category')
      if(data?.success){
        setCategories(data?.category);
      }

    }catch(error){
      console.log(error)



    }
  };

  useEffect(()=>{
    getAllCategory();
    getTotal();
  },[])
```



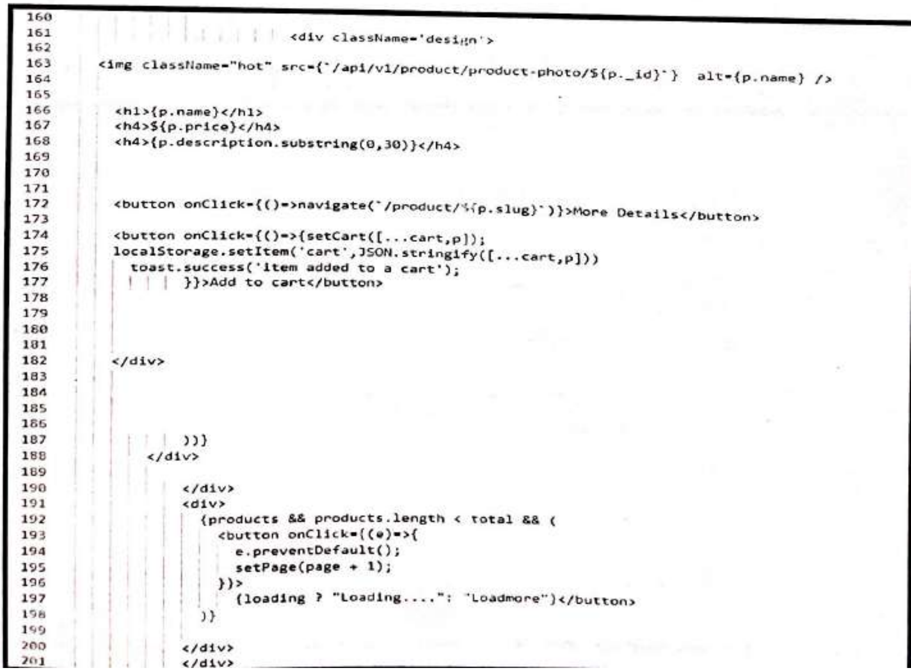
```
    //get all products

    const getAllProducts=async()=>{
      try{
        setLoading(true)
     const {data}= await axios.get(`/api/v1/product/product-list/${page}`);
     setLoading(false)
     setProducts(data?.products);

      }catch(error){
        setLoading(false)
    console.log(error)
      }
    }

   //get total count
  const getTotal  = async()=>{
   try{
  const {data}=await axios.get('/api/v1/product/product-count')
  if(data?.success){
  setTotal(data?.total)
  }
   }catch(error){
  console.log(error)
   }
  }
 useEffect(()=>{
   if (page===1) return;
     loadMore()
 },[page])
 //loadmore
 const loadMore=async()=>{
 try{
   setLoading(true)
   const {data}= await axios.get(`/api/v1/product/product-list/${page}`)
   setLoading(false)
   setProducts([...products,...data?.products])

}catch(error){
console.log(error)
setLoading(false)
}
}
```



```
    //filter by category
    const handleFilter=(value,id)=>{
      let all=[...checked]
      if(value){
        all.push(id)
      }else{
        all=all.filter(c=> c!==id)
      }
      setChecked(all);
    }
   useEffect(()=>{
 if(!checked.length || !radio.length) getAllProducts();
 //eslint-disable-next-line
   },[checked.length,radio.length])

  useEffect(()=>{
    if(checked.length || radio.length) filterProduct()
  },[checked,radio])

  //get filtered product
  const filterProduct = async()=>{
    try{
      const {data}=await axios.post('/api/v1/product/product-filters',{checked,radio})
      setProducts(data?.products)

    }catch(error){
 console.log(error)
    }
  }
```



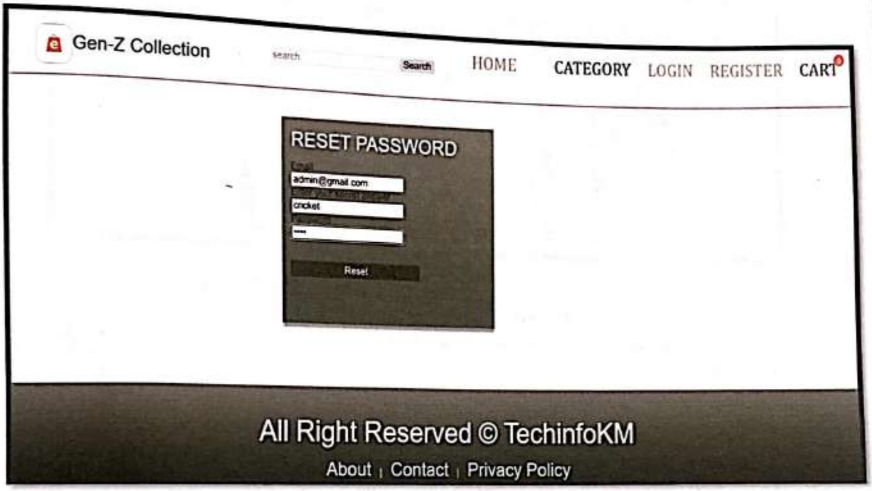
```
return (
  <div className='baap'>
  <Layout>

    <div className='flex8'>
      <div className='right'>
        <div>
          <h1 className='pass'>Filter by category</h1>
          {categories?.map(c=>(<div className='check'>
            <Checkbox key={c._id} onChange={(e)=> handleFilter(e.target.checked,c._id)}>
              {c.name}
            </Checkbox>
            </div>
          ))}
        </div>

        <div className='radio'>
          <h1>Filter by prices</h1>
          <Radio.Group onChange={e=> setRadio(e.target.value)}>
            {prices?.map(p=>(
              <div key={p._id}>
              <Radio value={p.array}>{p.name}</Radio>
              </div>
            ))}
          </Radio.Group>

        </div>
        <div className='reset'>
          <button onClick={()=>window.location.reload()}>Reset Filter</button>
        </div>
        </div>

        <div className='text'><h1 className='h9'><u>All Products</u></h1>

        <div>

          <div className='flex5'>

          {products?.map((p)=>(
```



```

                    <div className='design'>

  <img className="hot" src={`/api/v1/product/product-photo/${p._id}`}  alt={p.name} />

    <h1>{p.name}</h1>
    <h4>${p.price}</h4>
    <h4>{p.description.substring(0,30)}</h4>



    <button onClick={()=>navigate(`/product/${p.slug}`)}>More Details</button>

    <button onClick={()=>{setCart([...cart,p]);
    localStorage.setItem('cart',JSON.stringify([...cart,p]))
      toast.success('item added to a cart');
              }}>Add to cart</button>




    </div>




              ))}
          </div>

              </div>
              <div>
                {products && products.length < total && (
                  <button onClick={(e)=>{
                    e.preventDefault();
                    setPage(page + 1);
                  }}>
                    {loading ? "Loading....": "Loadmore"}</button>
                )}

              </div>
              </div>
```

OUTPUT

Forget and Reset Page:



Gen-Z Collection

search Search

HOME CATEGORY LOGIN REGISTER CART 0

RESET PASSWORD

Email

admin@gmail.com

cricket

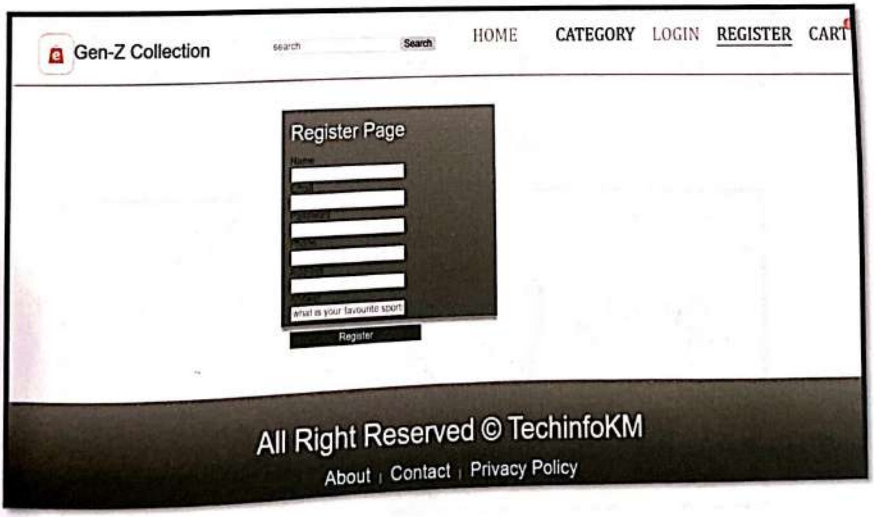
****

Reset

All Right Reserved © TechinfoKM

About | Contact | Privacy Policy

Register Page:



Gen-Z Collection

search Search

HOME CATEGORY LOGIN REGISTER CART

Register Page

Name

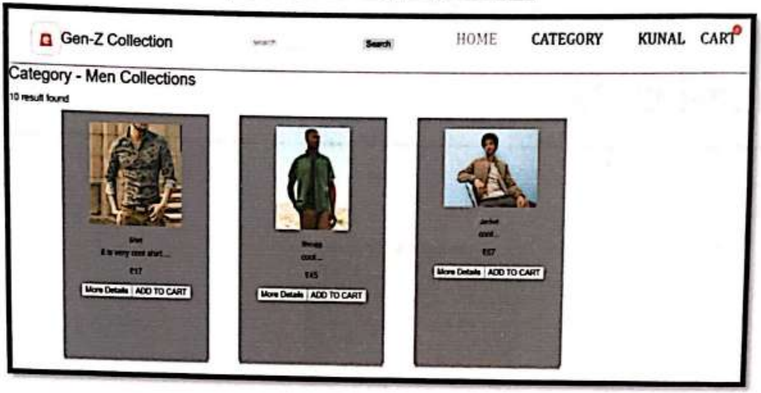
what is your favourite sport

Register

All Right Reserved © TechinfoKM

About | Contact | Privacy Policy

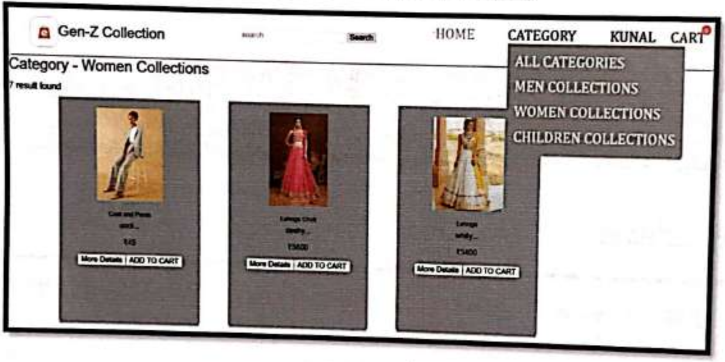
Men's Category Page:



Category - Men Collections

10 result found

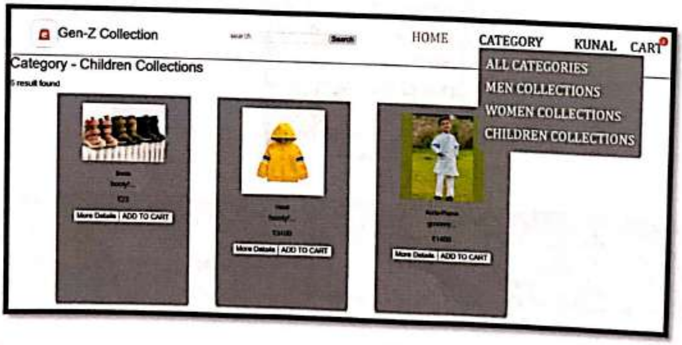
Women's Category Page:



Category - Women Collections

7 result found

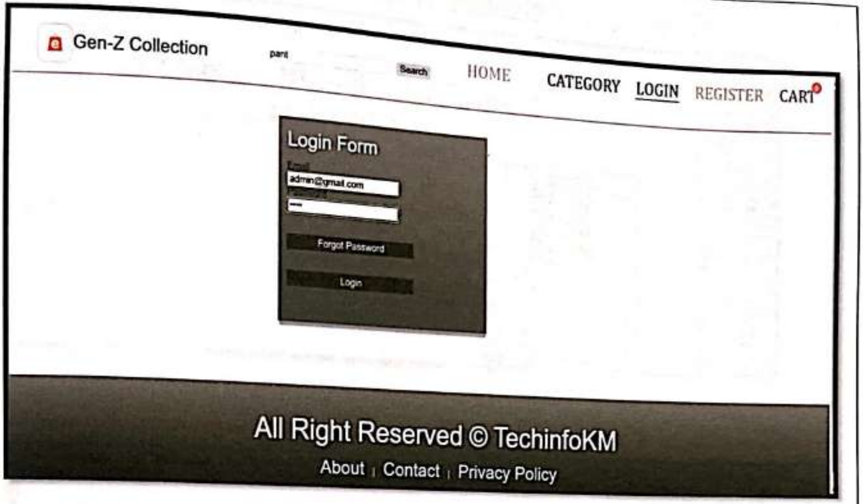
Kid's Category Page:



Category - Children Collections

5 result found

Login Page:



Login Form

Email

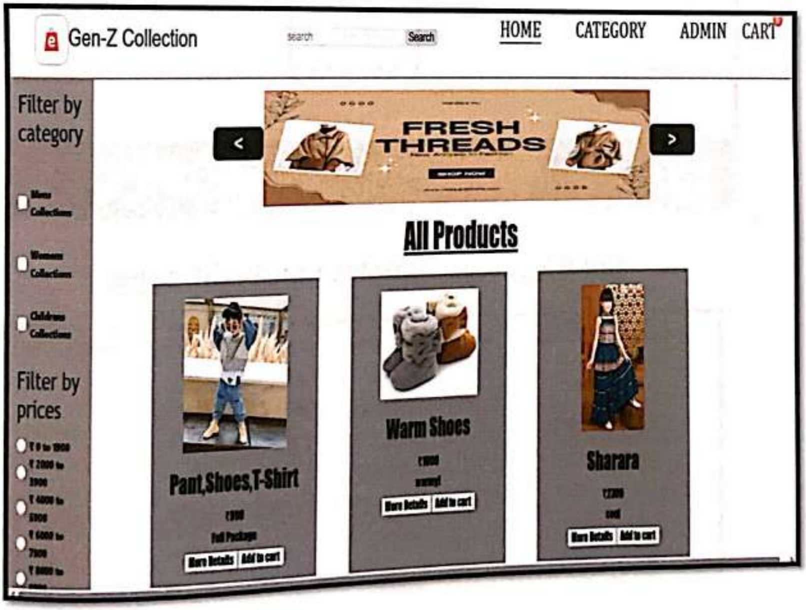
admin@gmail.com

Password

****

All Right Reserved © TechinfoKM

Home Page:



Pant,Shoes,T-Shirt

₹900

Full Package

Warm Shoes

₹1000

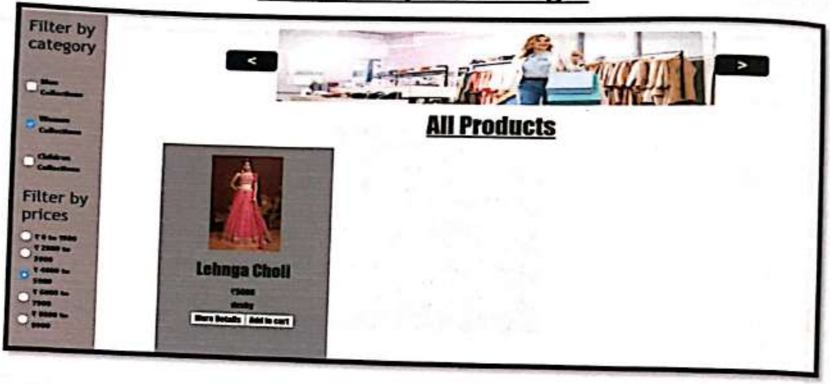
warm!

Sharara

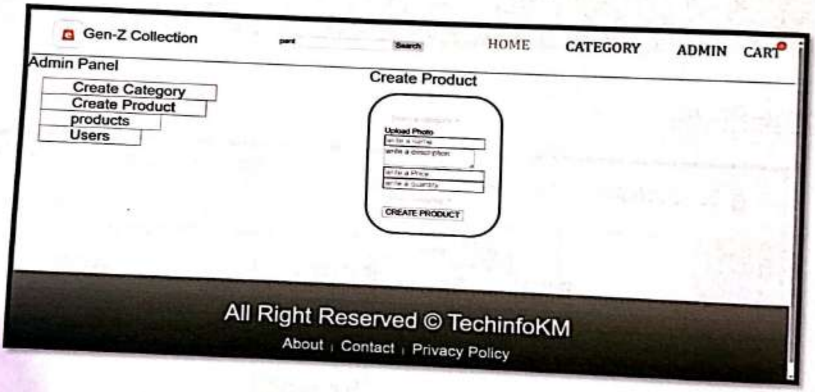
₹2300

cool

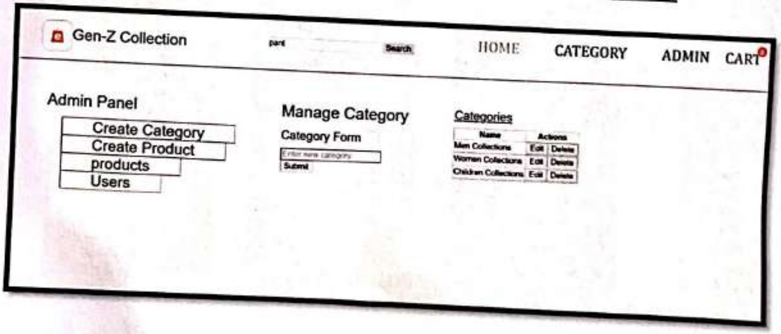
Selected by filters Page:



Admin Dashboard:

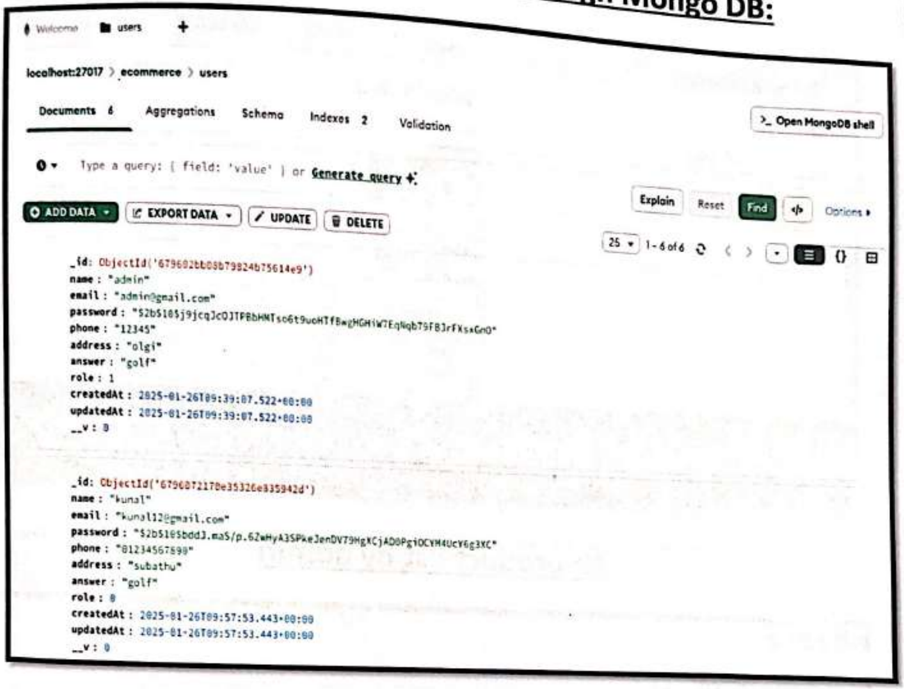


Create Category, Delete and edit page:



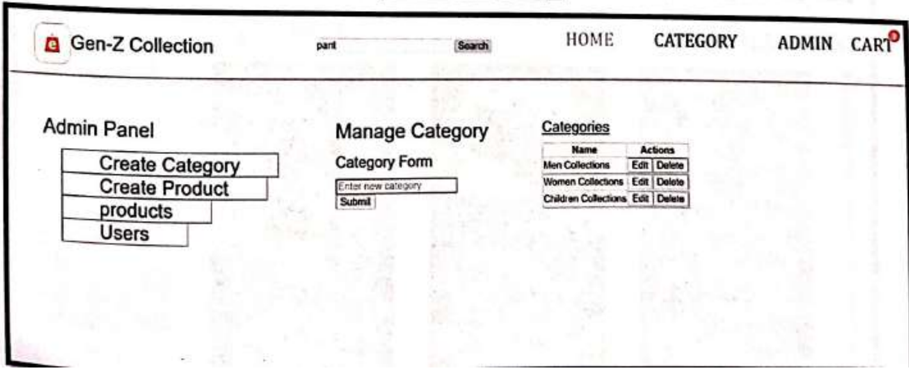
Name	Actions	
Men Collections	Edit	Delete
Women Collections	Edit	Delete
Children Collections	Edit	Delete

Data collected by admin through Mongo DB:



```
_id: ObjectId('679692bb08b79824b75614e9')
name : "admin"
email : "admin@gmail.com"
password : "$2b$10$j9jcqJcOJTPBbHNTso6t9uoHTfBwgHGHiW7EqNqb79FBJrFXsxGnO"
phone : "12345"
address : "olgi"
answer : "golf"
role : 1
createdAt : 2025-01-26T09:39:07.522+00:00
updatedAt : 2025-01-26T09:39:07.522+00:00
__v : 0

_id: ObjectId('6796072170e35326e835942d')
name : "kunal"
email : "kunal12@gmail.com"
password : "$2b$10$bddJ.maS/p.6ZwHyA3SPkeJenDV79HgXCjAD0PgiOCYM4UcY6g3XC"
phone : "01234567890"
address : "subathu"
answer : "golf"
role : 0
createdAt : 2025-01-26T09:57:53.443+00:00
updatedAt : 2025-01-26T09:57:53.443+00:00
__v : 0
```

All Category:



Admin Panel

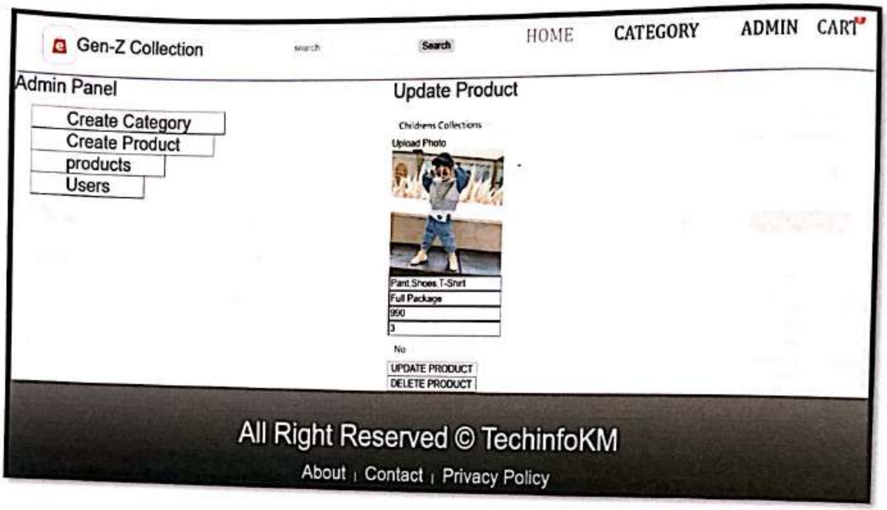
Manage Category

Category Form

Categories

Name	Actions	
Men Collections	Edit	Delete
Women Collections	Edit	Delete
Children Collections	Edit	Delete

Update and delete product by admin:



Gen-Z Collection

search Search

HOME CATEGORY ADMIN CART

Admin Panel

- Create Category
- Create Product
- products
- Users

Update Product

Childrens Collections

Upload Photo

Pant,Shoes,T-Shirt

Full Package

990

3

No

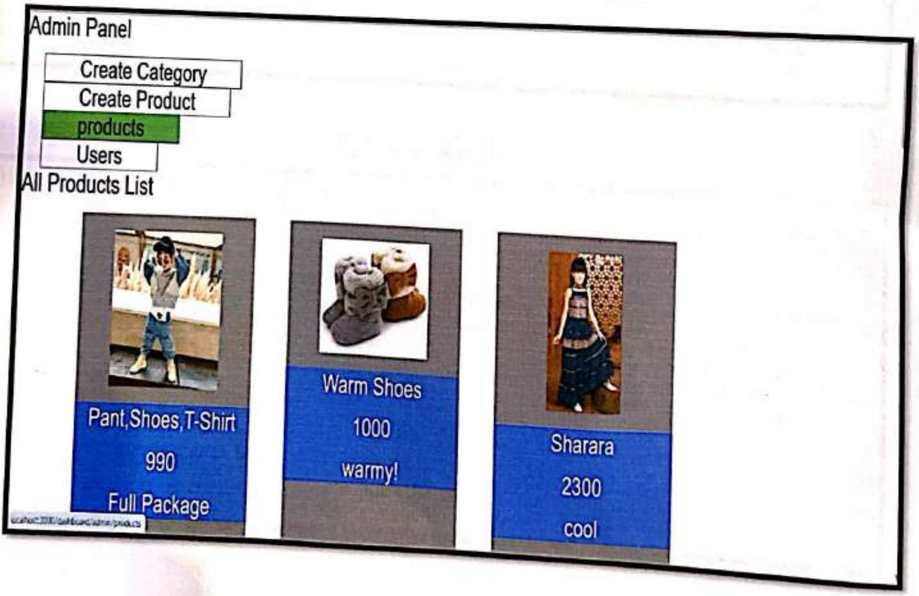
UPDATE PRODUCT

DELETE PRODUCT

All Right Reserved © TechinfoKM

About | Contact | Privacy Policy

All product list by admin



Admin Panel

- Create Category
- Create Product
- products
- Users

All Products List

Pant,Shoes,T-Shirt

990

Full Package

Warm Shoes

1000

warmy!

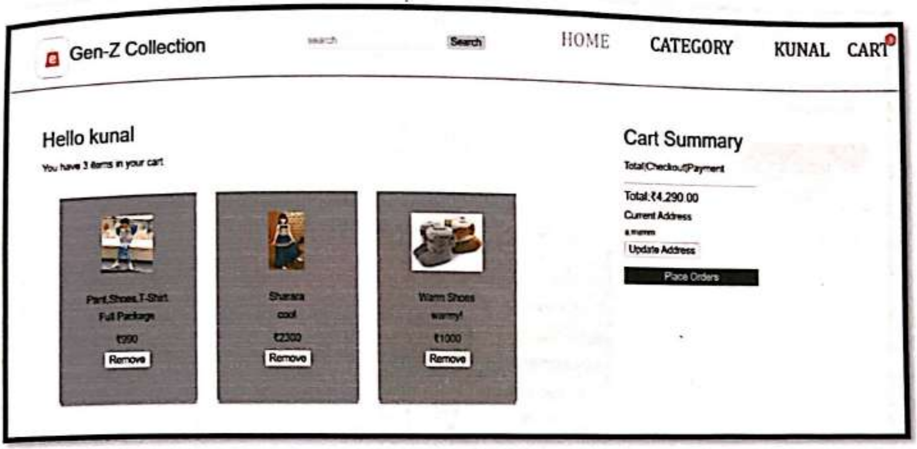
Sharara

2300

cool

localhost:3000/dashboard/admin/products

Add to Cart



Gen-Z Collection

search Search HOME CATEGORY KUNAL CART

Hello kunal

You have 3 items in your cart

Pant,Shoes,T-Shirt
Full Package
₹990
Remove

Sharara
cool
₹2300
Remove

Warm Shoes
warmy!
₹1000
Remove

Cart Summary

Total|Checkout|Payment

Total:₹4,290.00

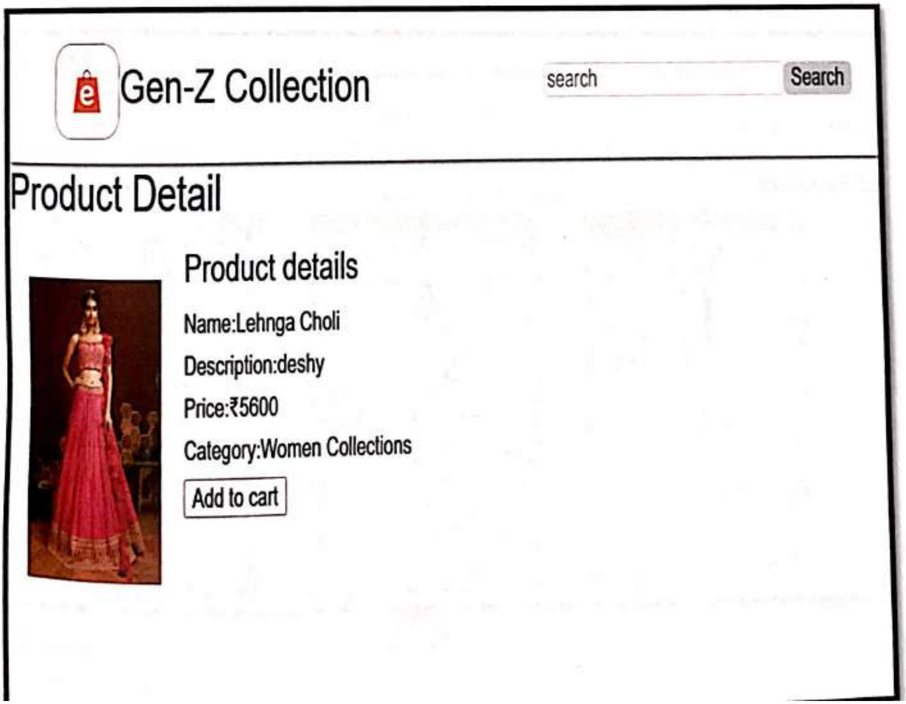
Current Address

[illegible]

Update Address

Place Orders

Product Detail



Gen-Z Collection

search Search

Product Detail

Product details

Name:Lehnga Choli

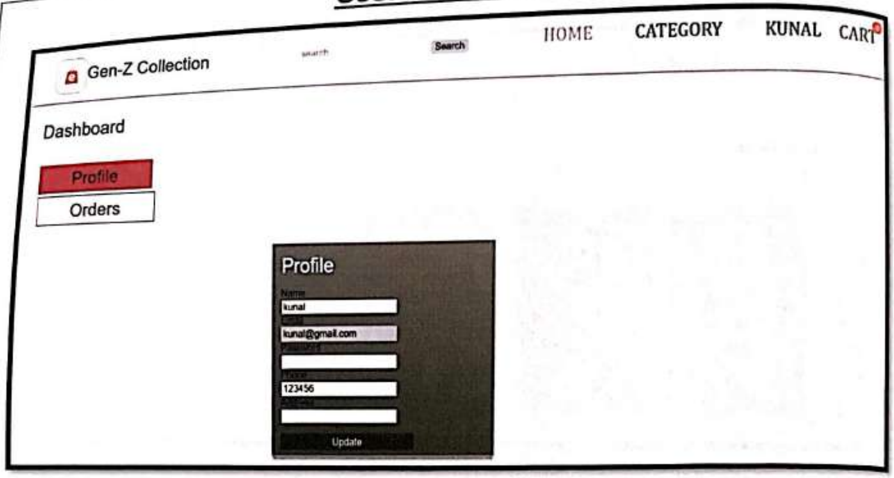
Description:deshy

Price:₹5600

Category:Women Collections

Add to cart

User Dashboard



Dashboard

Profile

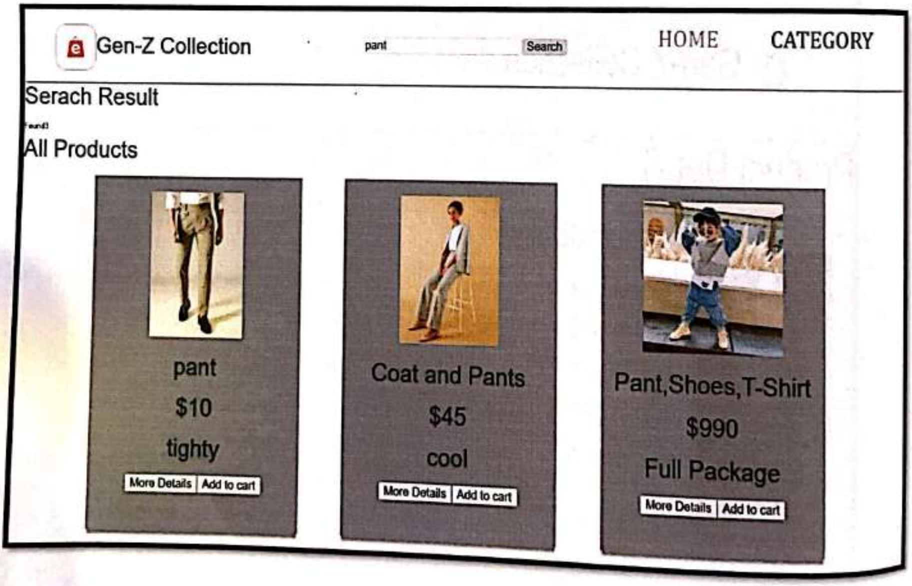
Name

kunal

kunal@gmail.com

123456

Search Result



pant

Serach Result

Found3

All Products

pant

$10

tighty

Coat and Pants

$45

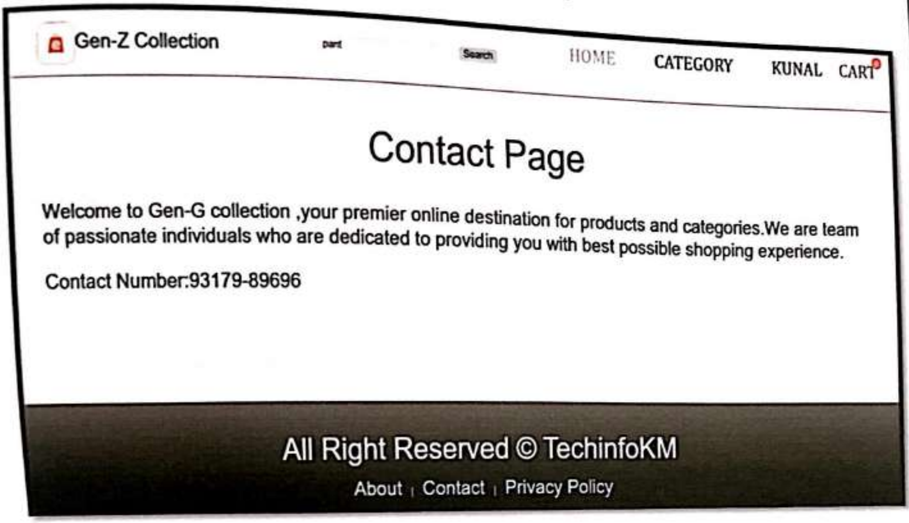
cool

Pant,Shoes,T-Shirt

$990

Full Package

Contact Page



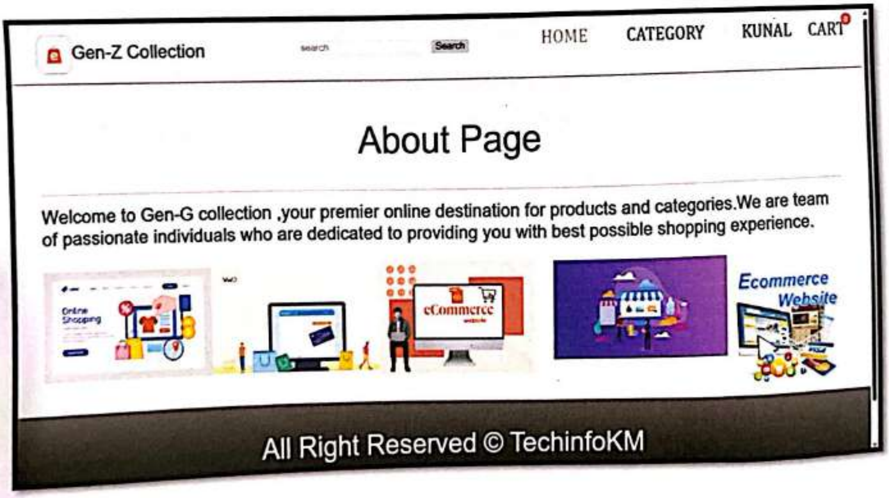
Contact Page

Welcome to Gen-G collection ,your premier online destination for products and categories.We are team of passionate individuals who are dedicated to providing you with best possible shopping experience.

Contact Number:93179-89696

All Right Reserved © TechinfoKM

About Page



About Page

Welcome to Gen-G collection ,your premier online destination for products and categories.We are team of passionate individuals who are dedicated to providing you with best possible shopping experience.

All Right Reserved © TechinfoKM

Bibliography

YouTube Channel:

- Code with Harry
- Sheryian's coding school

Websites:

- W3schools.org
- Geeks for greeks
- Hackerranks

Software Used:

- Visual Studio Code (developed by **Microsoft,** in **2015**)
- MongoDB Compass (developed by **MongoDB Incorporation,** and released in **2016**).