PROJECT REPORT ON

E-commerce

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSITUTE OF ENGINNERING & TECHNOLOGY

GLA UNIVERSITY, MATHURA

SUBMITTED TO:	SUBMITTED BY:
Ms. Ruchi Talwar	Abhishek Verma (201500027)
	Dhruv Modi (201500222
	Naman Sharma (201500428)
	Nishchal Tiwari (201500446)
	Varun Jadon (201500776)

GLA
UNIVERSITY
MATHURA

Accredited with A Grade by NAAC

DECLARATION

I would like to express my special thanks of gratitude to my project guide Ms. Ruchi Talwar mam who gave us the golden opportunity to do this wonderful project on the topic Poster Bazar(E-commerce website), which also helped us in doing a lot of research and we came to know about so many new things we are really thankful to them.

Secondly, we would also like to thank our parents and friends who helped us a lot in finalizing this project within the limited time frame.

Candidate’s Names:

AbhishekVerma (201500027)

Dhruv Modi (201500222)

Naman Sharma (201500428)

Nishchal Tiwari (201500446)

Varun Jadon (201500776)

CERTIFICATE

This is to certify that the above statements made by the candidates are correct to the best of my/our knowledge and belief.

Project Supervisor

Ms. Ruchi Talwar

Assistant Professor

Date: 24-Nov-2022

Table of Content

1. Introduction

1.1 Overview …………………………….

1.2 Objective.………………..…………

2. Technology Used

2.1 React …………………….………….

2.2 Node………………………………………

2.3 BOOTSTRAP……………………………

2.4 MongoDB…………………………………

3. System Requirements

3.1 Software Required…………………………

3.2 Hardware Required.……….……………………...

4. Implementation

4.1 Explanation of Source Code……………………......

4.2 Final Code ……………………...............................

4.3 Output…………………………………………....

5. Conclusion

INTRODUCTION

Poster Used for Motivation in hostel's rooms for students. The trend of Posters is increasing rapidly and is have several advantages also like Customers can order from their homes, workplaces as per their comfort, it is easy to cancel the transactions in online Poster ordering , online poster ordering made ordering easy as there are no lines to wait so that we can have Poster in some days and during lockdown, people were obligated to stay indoors. Considering all these things the idea of this project came into picture. The idea is to create a website that act as a platform for sellers (complex stores owner) and customer in GLA campus to get Posters for their rooms online.

Web:

Web development refers to the process of creating websites on the Internet.

The term "web development" is relatively broad in its application. You could create a single website page from a wix template, or you could painstakingly develop a massive website with thousands of original pages — and technically, both of those would count as web development.

Now a day's web development is the thing which is in boost.

TECHNOLOGY USED

HTML:

HTML is an acronym which stands for Hyper Text Markup Language which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

Hyper Text: Hyper Text simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. Hyper Text is a way to link two or more web pages (HTML documents) with each other.

Markup language: A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page: A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. With the help of HTML only, we can create static web pages.

CSS:

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

BOOTSTRAP:

- Bootstrap is the most popular HTML, CSS and JavaScript framework for developing a responsive and mobile friendly website.
- It is absolutely free to download and use.
- It is a front-end framework used for easier and faster web development. o It includes HTML and CSS based design templates for typography, forms, buttons, tables, navigation, modals, image carousels and many others. o It can also use JavaScript plug-ins. o It facilitates you to create responsive designs

REACT :

- Instead of manipulating the browser's DOM directly, React creates a virtual DOM in memory, where it does all the necessary manipulating, before making the changes in the browser DOM.
- A ReactJS application is made up of multiple components, each component responsible for outputting a small, reusable piece of HTML code. The components are the heart of all React applications. These Components can be nested with other

components to allow complex applications to be built of simple building blocks. ReactJS uses virtual DOM based mechanism to fill data in HTML DOM.

- Today, many JavaScript frameworks are available in the market(like angular, node), but still, React came into the market and gained popularity amongst them. The previous frameworks follow the traditional data flow structure, which uses the DOM (Document Object Model). DOM is an object which is created by the browser each time a web page is loaded. It dynamically adds or removes the data at the back end and when any modifications were done, then each time a new DOM is created for the same page. This repeated creation of DOM makes unnecessary memory wastage and reduces the performance of the application.

SYSTEM REQUIREMENTS

Software Requirement-

To build application –

- 64-bit Windows 8/10/11
- Libraries
- Visual Studio code (latest version).

To Run Website –

- Web Browsers (chrome, Mozilla)

Hardware Requirement –

- x86_64 CPU architecture; 2nd generation Intel Core or newer
- 8 GB RAM or more
- 8 GB of available disk space minimum

IMPLEMENTATION

Final Code:

□ SIGN UP Page:

```
import Axios from 'axios';
import { Link, useLocation, useNavigate } from 'react-router-dom';
import Container from 'react-bootstrap/Container';
import Form from 'react-bootstrap/Form';
import Button from 'react-bootstrap/Button';
import { Helmet } from 'react-helmet-async';
import { useContext, useEffect, useState } from 'react';
import { Store } from '../Store';
import { toast } from 'react-toastify';
import { getError } from '../utils';
```

```
export default function SignupScreen() {
  const navigate = useNavigate();
  const { search } = useLocation();
  const redirectInUrl = new URLSearchParams(search).get('redirect');
  const redirect = redirectInUrl ? redirectInUrl : '/';

  const [name, setName] = useState('');
  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');
  const [confirmPassword, setConfirmPassword] = useState('');

  const { state, dispatch: ctxDispatch } = useContext(Store);
  const { userInfo } = state;
  const submitHandler = async (e) => {
    e.preventDefault();
    if (password !== confirmPassword) {
      toast.error('Passwords do not match');
      return;
    }
    try {
      const { data } = await Axios.post('/users/signup', {
        name,
        email,
        password,
      });
      ctxDispatch({ type: 'USER_SIGNIN', payload: data });
      localStorage.setItem('userInfo', JSON.stringify(data));
      navigate(redirect || '/');
    } catch (err) {
      toast.error(getError(err));
    }
  };

  useEffect(() => {
    if (userInfo) {
      navigate(redirect);
    }
  }, [navigate, redirect, userInfo]);

  return (
    <Container className="small-container">
      <Helmet>
        <title>Sign Up</title>
      </Helmet>
      <h1 className="my-3">Sign Up</h1>
      <Form onSubmit={submitHandler}>
        <Form.Group className="mb-3" controlId="name">
          <Form.Label>Name</Form.Label>
          <Form.Control onChange={(e) => setName(e.target.value)} required />
        </Form.Group>

        <Form.Group className="mb-3" controlId="email">
          <Form.Label>Email</Form.Label>
          <Form.Control
            type="email"
            required
            onChange={(e) => setEmail(e.target.value)}
          />
        </Form.Group>
        <Form.Group className="mb-3" controlId="password">
          <Form.Label>Password</Form.Label>
```

```
          <Form.Control
            type="password"
            required
            onChange={(e) => setPassword(e.target.value)}
          />
          <Form.Group className="mb-3" controlId="confirmPassword">
            <Form.Label>Confirm Password</Form.Label>
            <Form.Control
              type="password"
              onChange={(e) => setConfirmPassword(e.target.value)}
              required
            />
          </Form.Group>
        </Form.Group>
        <div className="mb-3">
          <Button type="submit">Sign Up</Button>
        </div>
        <div className="mb-3">
          Already have an account?{' '}
          <Link to={/signin?redirect=${redirect}}>Sign-In</Link>
        </div>
      </Form>
    </Container>
  );
}
```

SIGN IN Page :

```
import { Link, useLocation, useNavigate } from 'react-router-dom';
import Container from 'react-bootstrap/Container';
import Form from 'react-bootstrap/Form';
import Button from 'react-bootstrap/Button';
import { Helmet } from 'react-helmet-async';
import { useContext, useEffect, useState } from 'react';
import { Store } from '../Store';
import { toast } from 'react-toastify';
import { getError } from '../utils';
import Axios from 'axios';

export default function SigninScreen() {
  const navigate = useNavigate();
  const { search } = useLocation();
  const redirectInUrl = new URLSearchParams(search).get('redirect');
  const redirect = redirectInUrl ? redirectInUrl : '/';

  const [email, setEmail] = useState('');
  const [password, setPassword] = useState('');

  const { state, dispatch: ctxDispatch } = useContext(Store);
  const { userInfo } = state;
  const submitHandler = async (e) => {
    e.preventDefault();
    try {
      const { data } = await Axios.post('http://localhost:4000/users/signin', {
        email,
        password,
      });
      ctxDispatch({ type: 'USER_SIGNIN', payload: data });
      localStorage.setItem('userInfo', JSON.stringify(data));
      navigate(redirect || '/');
    } catch (err) {
      toast.error(getError(err));
    }
  };

  useEffect(() => {
    if (userInfo) {
      navigate(redirect);
    }
  }, [navigate, redirect, userInfo]);

  return (
    <Container className="small-container">
      <Helmet>
        <title>Sign In</title>
      </Helmet>
      <h1 className="my-3">Sign In</h1>
      <Form onSubmit={submitHandler}>
        <Form.Group className="mb-3" controlId="email">
          <Form.Label>Email</Form.Label>
          <Form.Control
            type="email"
            required
            onChange={(e) => setEmail(e.target.value)}
          />
        </Form.Group>
        <Form.Group className="mb-3" controlId="password">
          <Form.Label>Password</Form.Label>
          <Form.Control
            type="password"
            required
            onChange={(e) => setPassword(e.target.value)}
          />
        </Form.Group>
        <div className="mb-3">
          <Button type="submit">Sign In</Button>
        </div>
        <div className="mb-3">
          New customer?{' '}
        <Link to={/signup?redirect=${redirect}}>Create your account</Link>
      </div>
    </Form>
   </Container>
  );
}
```

HOME Page:

```
import { useEffect } from "react";
import axios from "axios";
import { useReducer,useState } from 'react';
import logger from 'use-reducer-logger';
import Product from '../components/Product';
import Row from 'react-bootstrap/Row';
import Col from 'react-bootstrap/Col';
import LoadingBox from "../components/LoadingBox";
import MessageBox from "../components/MessageBox";
import {getError} from "../utils.js";
import Carousel from "../components/carousel";
import Dropdown from '../components/dropdown';
import { useParams } from "react-router-dom";
const reducer = (state, action) => {
  switch (action.type) {
    case 'FETCH_REQUEST':
      return { ...state, loading: true };
    case 'FETCH_SUCCESS':
      return { ...state, loading: false, products: action.payload };
    case 'FETCH_FAIL':
      return { ...state, loading: false, error: action.payload };
    default:
      return state;
  }
```

```
};


function HomeScreen() {


  console.log('filter')
  const [{ loading, error, products }, dispatch] = useReducer(logger(reducer), {
    products: [],
    loading: true,
    error: '',
  });

  useEffect(() => {
    const fetchData = async () => {

      dispatch({ type: 'FETCH_REQUEST' });
      try {
        var result;

          result = await axios.get('/api/products')
        dispatch({ type: 'FETCH_SUCCESS', payload: result.data })
      }
      catch (err) {
        dispatch({ type: 'FETCH_FAIL', payload: getError(error) } )
      }

    };
    fetchData();
  }, []);
```

```
return (
  <>
  <Carousel  />
    <div>
      <h1 className="fproduct">India's Largest Wall Art Store</h1>

      <Dropdown   />
      <div className="products">
        {
          loading ? (<div> <LoadingBox/></div>) : error ? (<MessageBox
variant="danger">{error}</MessageBox>) :
            <Row>
              {
                products.map(p => (

                  <Col sm={6} md={4} lg={3} className="mb-3" >
                    <Product product={p}  />
                  </Col>
                ))
              }
            </Row>
        }
      </div>
    </div>
  </>
);
}
export default HomeScreen;
```

PRODUCT Page:

```
import React from 'react'

import axios from 'axios'

import logger from 'use-reducer-logger'

import Row from 'react-bootstrap/Row';

import Col from 'react-bootstrap/Col'

import Product from '../components/Product'

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

import { useParams } from 'react-router-dom';

import ListGroup from 'react-bootstrap/ListGroup'

import Rating from '../components/Rating'

import Card from 'react-bootstrap/Card'

import Badge from 'react-bootstrap/Badge'

import Button from 'react-bootstrap/Button'

import {getError} from '../utils'

import MessageBox from '../components/MessageBox';

import LoadingBox from '../components/LoadingBox';

import { Store  } from '../Store';

const reducer = (state, action) => {

  switch (action.type) {

    case 'FETCH_REQUEST':

      return { ...state, loading: true };

    case 'FETCH_SUCCESS':

      return { ...state, loading: false, products: action.payload };

    case 'FETCH_FAIL':

      return { ...state, loading: false, error: action.payload };
```

```
        default:
            return state;
    }
};



function ProductScreen() {

    const params=useParams()
    const {slug}=params;


    const [{ loading, error, products }, dispatch] = useReducer(logger(reducer), {
        products: [],
        loading: true,
        error: '',
    });

    useEffect(() => {
        const fetchData = async () => {

            dispatch({ type: 'FETCH_REQUEST' });
            try {
                const result = await axios.get(/api/products/slug/${slug})
            dispatch({ type: 'FETCH_SUCCESS', payload: result.data })
          }
          catch (err) {
            dispatch({ type: 'FETCH_FAIL', payload:getError(err) })
          }

        };
        fetchData();
    }, [slug]);






    const { state, dispatch: ctxDispatch } = useContext(Store);
const {cart}=state;

const addtocart = async () => {
    const existItem=cart.cartItems.find(x=>x._id==products._id)
    const quantity = existItem?existItem.quantity+1:1
    const {data} = await axios.get(/api/products/${products._id})
    if(data.countInStock<quantity) {
        window.alert("Sorry , Product purchase limit exceeded")
    }
    ctxDispatch({type:'CART_ADD_ITEM', payload:{...products,quantity}})


  };
return loading?(
    <div><LoadingBox/></div>
  ):error?(<div> <MessageBox variant="danger"> {error}</MessageBox></div>):(
    <div>
    <Row>
        <Col  md={6}>
            <img  className='img-large'  src={products.image} alt={products.name} />
        </Col>
        <Col md={3}>
            <ListGroup variant="flush">
                <ListGroup.Item>
                    <h1>{products.name}</h1>
                </ListGroup.Item>


                <ListGroup.Item>
                    <Rating   rating={products.rating} numReviews={products.numReviews}  />
                </ListGroup.Item>
                <ListGroup.Item>
                    <h5>Price : $ {products.price}</h5>
                </ListGroup.Item>
                <ListGroup.Item>
                    <p><b>Description : </b> {products.description} </p>
        </ListGroup.Item>
      </ListGroup>
    </Col>
    <Col md={3}>
    <Card>
    <Card.Body>
    <ListGroup  variant='flush'>
      <ListGroup.Item>
        <Row>
          <Col>Price:</Col>
          <Col>${products.price}</Col>
        </Row>
      </ListGroup.Item>
      <ListGroup.Item>
        <Row>
          <Col>Status:</Col>
          <Col>{products.countInStock>0?<Badge bg="success">In
Stock</Badge>:<Badge bg="danger">Unavailiable</Badge>     }</Col>
        </Row>
      </ListGroup.Item>
      {
        products.countInStock > 0 && (
          <ListGroup.Item>
            <div className="d-grid">
              <Button  variant="primary"  onClick={addtocart} >
```

```
                          Add to Cart
                        </Button>
                      </div>
                    </ListGroup.Item>
                  )
                }
          </ListGroup>
          </Card.Body>
          </Card>
          </Col>

        </Row>

        </div>
      )
    }

export default ProductScreen;
```

CART Page:

```
import { useContext } from 'react';
import { Store } from '../Store';
import { Helmet } from 'react-helmet-async';
import Row from 'react-bootstrap/Row';
import Col from 'react-bootstrap/Col';
import MessageBox from '../components/MessageBox';

import ListGroup from 'react-bootstrap/ListGroup';

import Button from 'react-bootstrap/Button';

import Card from 'react-bootstrap/Card';

import { Link, useNavigate } from 'react-router-dom';

import axios from 'axios';


export default function CartScreen() {

  const navigate = useNavigate();

  const { state, dispatch: ctxDispatch } = useContext(Store);

  const {

    cart: { cartItems },

  } = state;


  const updateCartHandler = async (item, quantity) => {

    const { data } = await axios.get(/api/products/${item._id});

    if (data.countInStock < quantity) {

      window.alert('Sorry. Product is out of stock');

      return;

    }

    ctxDispatch({

      type: 'CART_ADD_ITEM',

      payload: { ...item, quantity },

    });

  };
```

```
const removeItemHandler = (item) => {
  ctxDispatch({ type: 'CART_REMOVE_ITEM', payload: item });
};

const checkoutHandler = () => {
  navigate('/signin?redirect=/shipping');
};

return (
  <div>
    <Helmet>
      <title>Shopping Cart</title>
    </Helmet>
    <h1>Shopping Cart</h1>
    <Row>
      <Col md={8}>
        {cartItems.length === 0 ? (
          <MessageBox>
            Cart is empty. <Link to="/">Go Shopping</Link>
          </MessageBox>
        ) : (
          <ListGroup>
            {cartItems.map((item) => (
              <ListGroup.Item key={item._id}>
                <Row className="align-items-center">
          <Col md={4}>
            <img
              src={item.image}
              alt={item.name}
              className="img-fluid rounded img-thumbnail"
            ></img>{' '}
            <Link className="item-name"
to={/product/${item.slug}}>{item.name}</Link>
          </Col>
          <Col md={3}>
            <Button
              onClick={() =>
                updateCartHandler(item, item.quantity - 1)
              }
              variant="light"
              disabled={item.quantity === 1}
            >
              <i className="fas fa-minus-circle"></i>
            </Button>{' '}
            <span>{item.quantity}</span>{' '}
            <Button
              variant="light"
              onClick={() =>
                updateCartHandler(item, item.quantity + 1)
              }
```

```
        disabled={item.quantity === item.countInStock}
       >
        <i className="fas fa-plus-circle"></i>
       </Button>
      </Col>
      <Col md={3}>${item.price}</Col>
      <Col md={2}>
       <Button
        onClick={() => removeItemHandler(item)}
        variant="light"
       >
        <i className="fas fa-trash"></i>
       </Button>
      </Col>
     </Row>
    </ListGroup.Item>
   ))}
  </ListGroup>
 )}
</Col>
<Col md={4}>
 <Card>
  <Card.Body>
   <ListGroup variant="flush">
    <ListGroup.Item>
```

```
         <h3>
          Subtotal ({cartItems.reduce((a, c) => a + c.quantity, 0)}{' '}
          items) : $
          {cartItems.reduce((a, c) => a + c.price * c.quantity, 0)}
         </h3>
        </ListGroup.Item>
        <ListGroup.Item>
         <div className="d-grid">
          <Button
           type="button"
           variant="primary"
           onClick={checkoutHandler}
disabled={cartItems.length === 0}
          >
           Proceed to Checkout
          </Button>
         </div>
        </ListGroup.Item>
       </ListGroup>
      </Card.Body>
     </Card>
    </Col>
   </Row>
  </div>
 );
```

```
}
```

Shipping Address Page:

```
import React, { useContext, useEffect, useState } from 'react';

import { Helmet } from 'react-helmet-async';

import Form from 'react-bootstrap/Form';

import Button from 'react-bootstrap/Button';

import { useNavigate } from 'react-router-dom';

import { Store } from '../Store';

import CheckoutSteps from '../components/CheckoutSteps';


export default function ShippingAddressScreen() {

  const navigate = useNavigate();

  const { state, dispatch: ctxDispatch } = useContext(Store);

  const {

    fullBox,

    userInfo,

    cart: { shippingAddress },

  } = state;

  const [fullName, setFullName] = useState(shippingAddress.fullName  '');

  const [address, setAddress] = useState(shippingAddress.address  '');

  const [city, setCity] = useState(shippingAddress.city  '');

  const [postalCode, setPostalCode] = useState(

    shippingAddress.postalCode  ''

  );

  useEffect(() => {
```

```
  if (!userInfo) {
    navigate('/signin?redirect=/shipping');
  }
}, [userInfo, navigate]);
const [country, setCountry] = useState(shippingAddress.country || '');
const submitHandler = (e) => {
  e.preventDefault();
  ctxDispatch({
    type: 'SAVE_SHIPPING_ADDRESS',
    payload: {
      fullName,
      address,
      city,
      postalCode,
      country,
      location: shippingAddress.location,
    },
  });
  localStorage.setItem(
    'shippingAddress',
    JSON.stringify({
      fullName,
      address,
      city,
      postalCode,
   country,
   location: shippingAddress.location,
  })
 );
 navigate('/payment');
};

useEffect(() => {
 ctxDispatch({ type: 'SET_FULLBOX_OFF' });
}, [ctxDispatch, fullBox]);

return (
 <div>
  <Helmet>
   <title>Shipping Address</title>
  </Helmet>

  <CheckoutSteps step1 step2></CheckoutSteps>
  <div className="container small-container">
   <h1 className="my-3">Shipping Address</h1>
   <Form onSubmit={submitHandler}>
    <Form.Group className="mb-3" controlId="fullName">
     <Form.Label>Full Name</Form.Label>
     <Form.Control
      value={fullName}
    onChange={(e) => setFullName(e.target.value)}
    required
  />
</Form.Group>
<Form.Group className="mb-3" controlId="address">
  <Form.Label>Address</Form.Label>
  <Form.Control
    value={address}
    onChange={(e) => setAddress(e.target.value)}
    required
  />
</Form.Group>
<Form.Group className="mb-3" controlId="city">
  <Form.Label>City</Form.Label>
  <Form.Control
    value={city}
    onChange={(e) => setCity(e.target.value)}
    required
  />
</Form.Group>
<Form.Group className="mb-3" controlId="postalCode">
  <Form.Label>Postal Code</Form.Label>
  <Form.Control
    value={postalCode}
    onChange={(e) => setPostalCode(e.target.value)}
```

```
              required
            />
          </Form.Group>
          <Form.Group className="mb-3" controlId="country">
            <Form.Label>Country</Form.Label>
            <Form.Control
              value={country}
              onChange={(e) => setCountry(e.target.value)}
              required
            />
          </Form.Group>


          <div className="mb-3">
            <Button variant="primary" type="submit">
              Continue
            </Button>
          </div>
        </Form>
      </div>
    </div>
  );
}
```

User Profile Page:

```
import React from 'react'
```

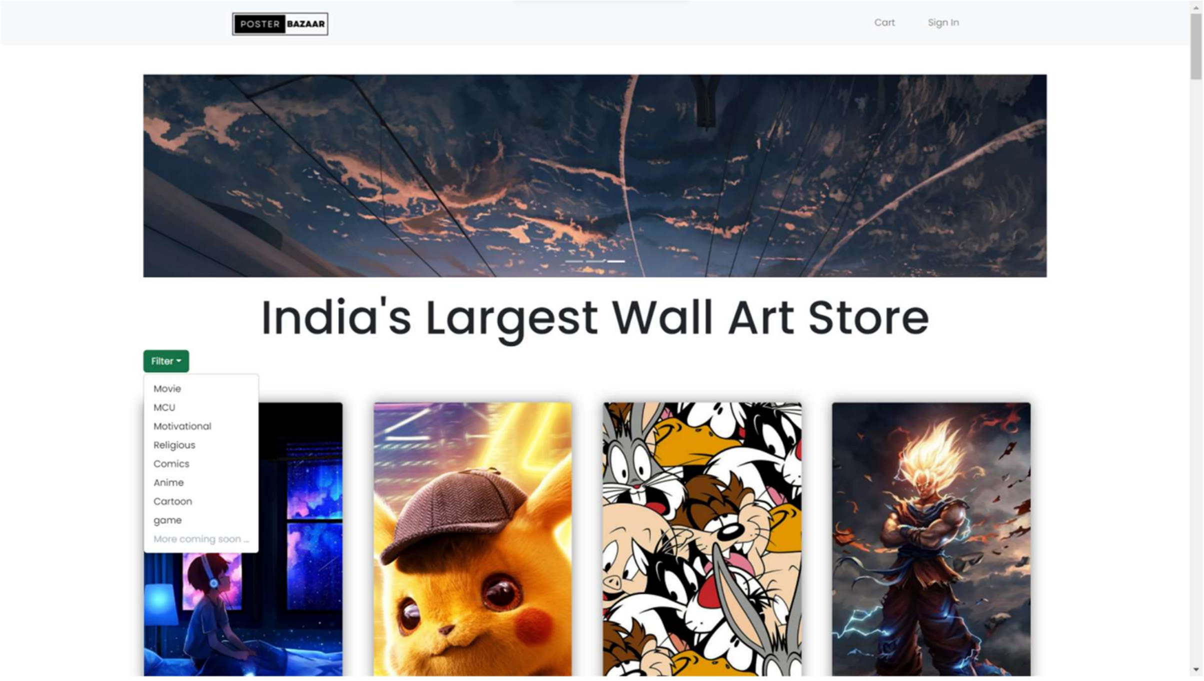
```
function userProfile() {
  var data=localStorage.getItem('userInfo')
  data=JSON.parse(data)
  return (
    <div className='userprofile'>


<div class="card middle  epros">
  <div class="top-section">
    <img src='https://images.unsplash.com/photo-1490585639275-bd19bcafe620?ixlib=rb-1.2.1&ixid=eyJhcHBfaWQiOjEyMDd9&auto=format&fit=crop&w=1350&q=80'
alt="picture"/>
    <div class="menu-icon">
      <span class="s1"></span>
      <span class="s2"></span>
    </div>
    <div class="name">
      {data.name} <br/>
      <span> prime user</span>
    </div>
  </div>

  <div class="info-section">
    <h2>History
      <div class="border"></div>
    </h2>
    <p>
     <b>MAIL : </b> {data.email} <br />
     <b>Merchant : </b>{data.isAdmin?"Yes":"No"}
    </p>
    <h2>contact
      <div class="border"></div>
    </h2>
    <div class="s-m">
      <a href="" class="fab fa-facebook-f"></a>
      <a href="" class="fab fa-twitter"></a>
      <a href="" class="fab fa-instagram"></a>
      <a href="" class="fab fa-youtube"></a>
      <a href="" class="fab fa-whatsapp"></a>
    </div>
  </div>
</div>


    </div>
  )
}


export default userProfile;
```

BACKEND CODE : (SIGNUP , LOGIN)- □

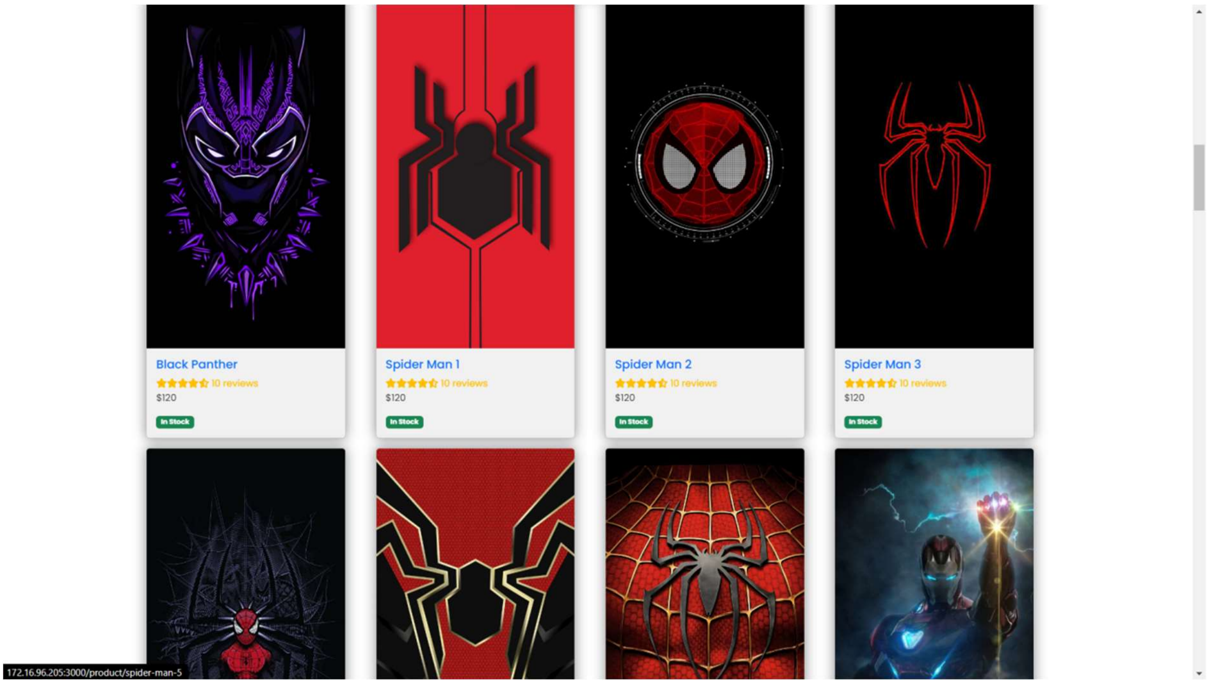
```
 const express = require('express')
const router = express.Router()
const bcrypt = require("bcryptjs");
const expressAsyncHandler = require("express-async-handler");
const User = require('../schemas/userSchema')
const { generateToken } = require('../utils')
router.post('/signin',
    expressAsyncHandler(async(req, res) => {
        const user = await User.findOne({ email: req.body.email })

        if (user) {
            if (bcrypt.compareSync(req.body.password, user.password)) {
                res.send({
                    _id: user._id,
                    name: user.name,
                    email: user.email,
                    isAdmin: user.isAdmin,
                    token: generateToken(user)
                })
                return;
            }
        }
        res.status(401).send({ message: 'Invalid email or password' })
    }))

router.post('/signup',
    expressAsyncHandler(async(req, res) => {
        const newUser = new User({
            name: req.body.name,
            email: req.body.email,
            password: bcrypt.hashSync(req.body.password),
        });
        const user = await newUser.save()
        res.send({
            _id: user._id,
            name: user.name,
            email: user.email,
            isAdmin: user.isAdmin,
            token: generateToken(user)
        })

    })

)
module.exports = router;
```

WORKING

1. Customer will open the site and will see the home page.
2. On home page we have two options signup and login.
3. Here a new customer can sign up and can create an account, Existed customer can login by entering the desired details.
4. After login, the customer will be directed to the second page of the website, where we have all the Posters with a filter button where user can get Poster according to their desired category.
5. On click the ‘Add’ button under Posters, customer can add the poster to the cart.



India's Largest Wall Art Store

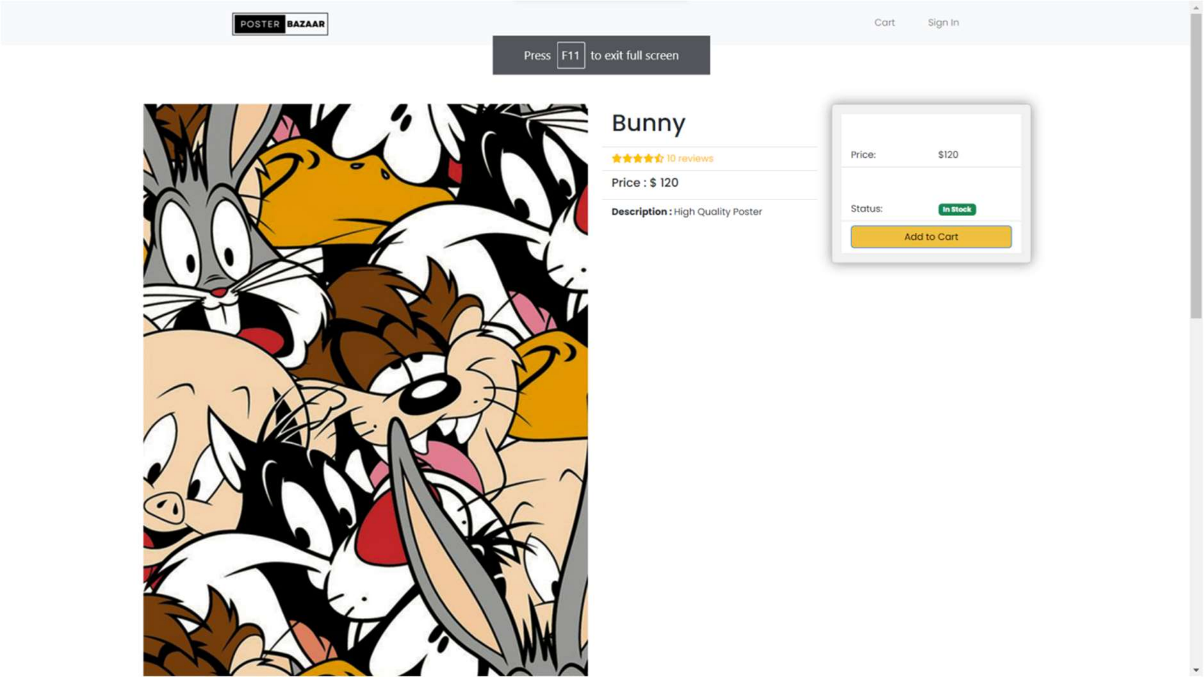

Black Panther
★★★★★ 10 reviews
$120
In Stock

Spider Man 1
★★★★★ 10 reviews
$120
In Stock

Spider Man 2
★★★★★ 10 reviews
$120
In Stock

Spider Man 3
★★★★★ 10 reviews
$120
In Stock

172.16.96.205:3000/product/spider-man-5



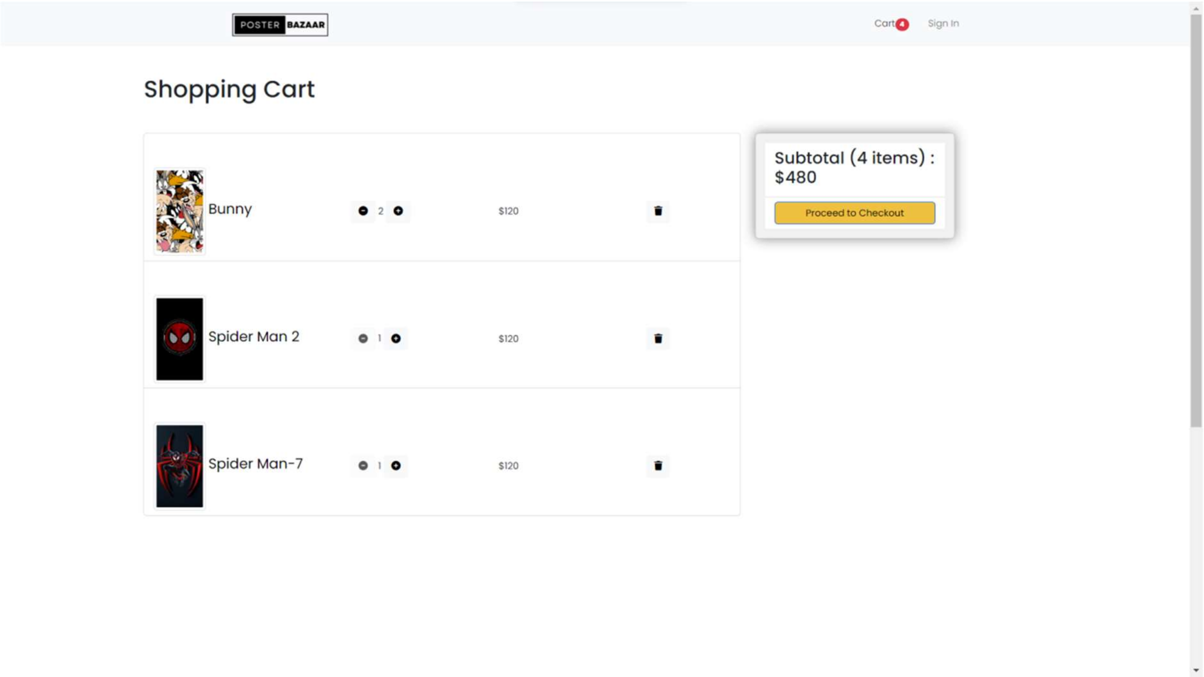
Bunny

★★★★☆ 10 reviews

Price : $ 120

Description : High Quality Poster

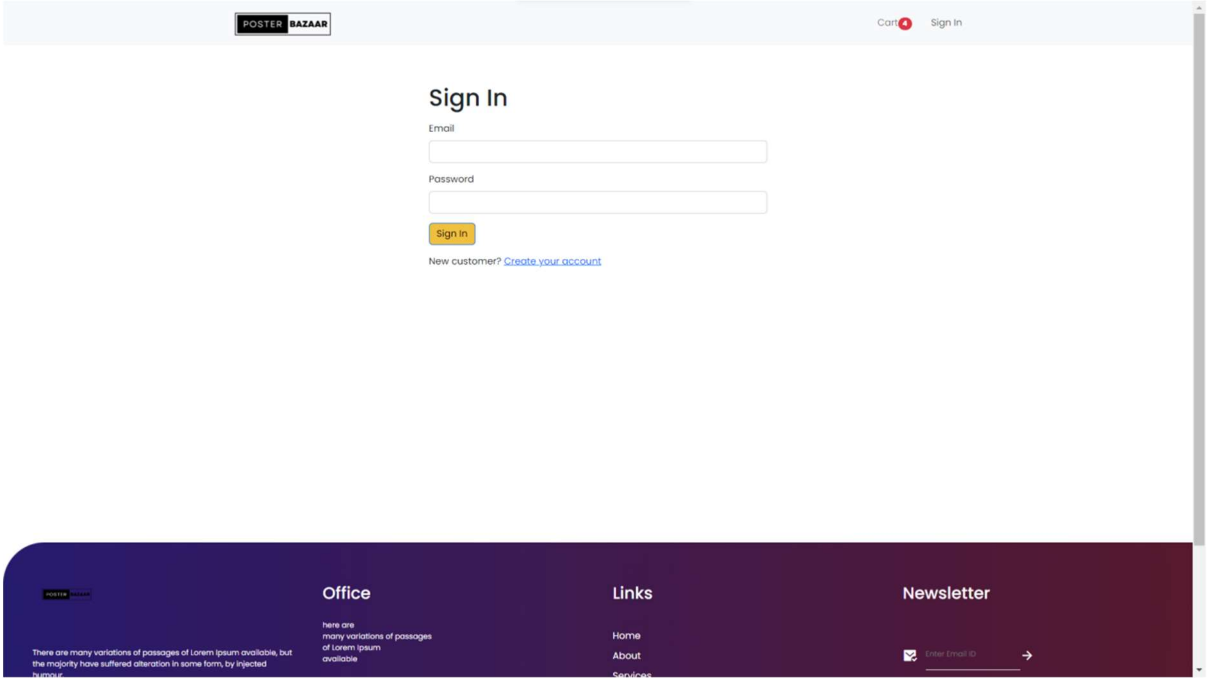
Price:	$120
Status:	In Stock



Shopping Cart

Bunny	2	$120	
Spider Man 2	1	$120	
Spider Man-7	1	$120	

Subtotal (4 items) : $480



Sign In

Email

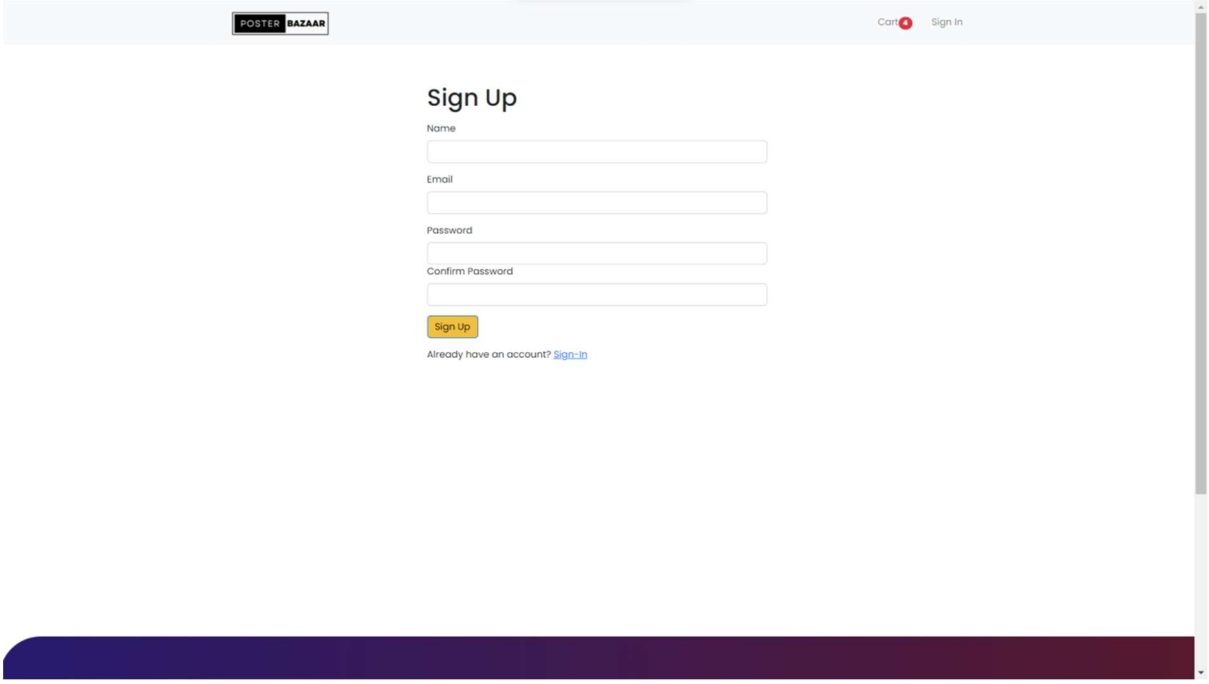
Password

Sign In

New customer? Create your account

There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour

here are many variations of passages of Lorem Ipsum available



Sign Up

Name

Email

Password

Confirm Password

Sign Up

Already have an account? Sign-In

CONCLUSION AND FUTURE WORK

Conclusion:

As per the goal of this project an attempt is made to show how we can take a business online and also enable users to Order Posters online following B2C business model.

Poster Bazar website acts as a platform between user and sellers. Also, it provides support to sellers to take their business online and grow it.

Future work: -

Many different adaptations, tests, and experiments have been left for the future due to lack of time. The present system is just an interactive UI with user database but Future work concerns with developing an Admin panel and a data base to store products detail with user information.